FactoRing: Asynchronous TSN-compliant Network with low bounded Jitters for Industry 4.0

Ahlem Mifdaoui[1], Jérôme Lacan[1], Arnaud Dion[1], Fabrice Frances[1] and Pierre Leroy[2]
[1]ISAE-Supaéro/ University of Toulouse, [2] Leroy Axseam

Abstract*—Time-Sensitive Networking (TSN) describes a set of features extending the functionalities and Quality of Service (QoS) of standard Ethernet to enable determinism, reliability and reconfigurability, key requirements for Industry 4.0. The TSN profile for industrial automation (IEC/IEEE 60802 standard) defines specific options to favor reliability and reconfigurability. Nevertheless, there are multiple possible options for scheduling to guarantee determinism. Today, the scheduling standards in TSN can be categorized according to the implemented communication paradigm: asynchronous or synchronous. This paradigm is of utmost importance to quantify the synchronization need and the reconfigurability effort. The main contribution of this work is the specification of an asynchronous TSN-compliant network for Industry 4.0, FactoRing, that bridges the gap between both paradigms to guarantee low bounded jitters and latencies, without the need of synchronization and complex network planning. Moreover, FactoRing supports ring-based topologies to significantly reduce installation wiring and costs. In this paper, we first present the main industry 4.0 requirements and assess the ability of recommended TSN mechanisms for industrial automation versus such requirements. Afterwards, we detail the main features of Factoring including QoS, reliability and reconfiguration management. Finally, preliminary results on performance metrics like jitters, latencies and buffer usage are discussed and the first conclusions on the promises of Factoring to meet Industry 4.0 requirements are derived.

***Keywords*-Industry 4.0, TSN, Scheduling, Ethernet, Real-Time**

I. Introduction

Internet of Things (IoT) technologies are reshaping the industrial automation process and driving major system architecture changes in Industry 4.0 [1], known as the fourth industrial revolution. This revolution enables process optimization and new incomes for manufacturers, for instance the market for motion control is expected to reach $16.5B by 2025 [2].

However, these changes in industrial automation process bring new challenges to improve real-time capabilities of automation controllers, that need to process various workloads, like audio and video streams as well as control traffic, and simultaneously communicate with other controllers in a timely and reliable manner. One of the main performance bottleneck in such systems is the communication network.

Historically, the industrial communication systems have two communication domains with different objectives and requirements: Operation Technology (OT) and Information Technology (IT). The former enables the control processes with time-sensitive traffic, such as controlling the position of a mechanical device; whereas the latter deals with computers for manufactory facilities to transmit, process and store data with soft or non real-time constraints. This fact generally leads to heterogeneous and hierarchical communication architecture with legacy real-time buses for OT, such as CAN and Profibus; and high-rate Real-Time Ethernet (RTE) propriety solution for IT, such as PROFINET [3] and EtherCAT[4]. These standards generally share similar requirements but their implementations differ, which necessitate the use of protocol conversion gateways to enable interoperability. This solution has led to limited scalability and flexibility, in addition to increased complexity and costs of industrial automation architectures.

To cope with these emerging challenges, Time-Sensitive Networking (TSN) [5] is considered as an appealing solution, that offers a set of IEEE standards to bring timeliness and reliability to Ethernet technology, and enables the convergence of OT and IT domains. Different traffic classes can be sent on the same communication link, while guaranteeing the requirements of each class. More recently, a draft of IEC/IEEE 60802 standard [6] has been published to define a TSN profile for industrial automation with recommended options, to guarantee the compatibility of existing RTE solutions [7] with TSN, and to achieve a convergent network enabling the interoperability between industrial devices from different manufacturers. The major benefit of TSN is being an open standard, an overwhelming benefit for the industrial market that has struggled for decades with multiple incompatible proprietary solutions. Therefore, TSN Ethernet can be the common communication protocol to connect equipment from different manufacturers, that fulfills the emerging requirements of Industry 4.0 and drives down the development and maintenance costs.

TSN describes a set of features extending the functionalities and Quality of Service (QoS) of standard Ethernet to enable determinism, reliability and reconfigurability, key requirements for Industry 4.0. The TSN profile for industrial automation defines specific options for reliability and configuration management, i.e., IEEE 802.1CB [8] and the fully centralized model of IEEE 802.1Qcc [9]. Nevertheless, there are multiple possible options for scheduling to guarantee determinism. Today, the scheduling standards in TSN can be categorized according to the implemented communication paradigm: asynchronous or synchronous. This paradigm is of utmost importance to quantify the synchronization need and the reconfigurability effort. The asynchronous paradigm simplifies the system reconfigurability, but it needs at the same time further proofs of guaranteed bounded latencies, besides it cannot guarantee low bounded jitter (a key requirement for some industrial traffic). On the other hand, the synchronous paradigm guarantees low bounded latencies and jitters, but needs commonly (complex) global schedule and high synchronization precision.

Hence, the main contribution of this work is the specification of an asynchronous TSN-compliant network for Industry 4.0, FactoRing, that bridges the gap between both paradigms to guarantee low bounded jitters and latencies, without the need of global synchronization and complex network planning. Moreover, FactoRing supports ring-based topologies to significantly reduce installation wiring and costs.

In the next section, we present the main industry 4.0 requirements and discuss the recommended TSN mechanisms for industrial automation in [6] versus such requirements. Afterwards, the main features of FactoRing including scheduling, QoS, and resource management are detailed in Section 3. In Section 4, some numerical results on performance metrics like jitters, latencies and buffer usage are discussed and the first conclusions on its potential promises are derived.

II. TSN mechanisms vs Industry 4.0 requirements

In this section, we first present the main requirements [10] to fulfill for industrial automation applications in Industry 4.0 era. Then, we discuss the pros and cons of the main TSN mechanisms, recommended in the TSN profile for industrial automation [6], vs the Industry 4.0 requirements.

A. Industry 4.0 and Network Requirements

In [10], the authors have developed the main requirements to design smart factories or extend a traditional factory to make it smart. In this section, we summarize these main requirements from the network point of view.

- Modularity: this implies that the devices can be added on-line using a plug-and-play principle to enable quick reconfiguration and overcome failures;
- Interoperability: a standardized solution is crucial to enable the interoperability between different communication domains from the sensors to the machines and from machines to machines. The choice of TSN Ethernet technology favors this requirement;
- Reconfigurability: this implies a need for supervising mechanisms to enable decision on reconfiguration needs and handle modifications at the hardware level (adding a new component or failure) as well as at the software level (new application or data flow);
- Real-Time capability: the exchanged data must not only be correct, but also meet the constraints of deadline and jitter;
- Safety: This includes fault tolerance and resilience to malicious attacks.

Furthermore, the network shall be efficient to meet the design requirements for the least amount of money. Therefore, a minimized configuration effort and reduced implementation costs are among the most important issues to guarantee. These requirements will be considered to analyze the pros and cons of each TSN mechanism recommended in [6].

B. TSN mechanisms vs requirements

The features of TSN are incorporated at the data link layer through extending the IEEE 802.1 [11] on top of the IEEE 802.3 standard [12] (Ethernet MAC layer). TSN addresses mainly the needs of: (i) QoS management of mixed-criticality data, where some traffic needs null jitter and/or bounded latencies; (ii) a high reliability level with zero packet loss due to buffer congestion and low packet loss due to equipment failure; (iii) an easy (re)-configuration process to increase flexibility. These capabilities are achieved through the main TSN components in Table I: synchronization, scheduling, reliability and resource management. Each component consists of one or many TSN standards, that are briefly described and discussed in this section.

TABLE I
IEEE 802.1 TSN Standards

Key component	IEEE 802.1 standard	Features
Synchronization	802.1AS	Synchronization Protocol
Scheduling	802.1Qbu	Frame Preemption
	802.1Qbv	Time-Aware Shaper
	802.1Qch	Cyclic Queueing and Forwarding
	802.1Qcr	Asynchronous Traffic Shaper
Reliability	802.1CB	Frame Replication and Elimination for Reliability
	802.1Qci	Filtering and Policing
	802.1Qca	Path Control and reservation
Resource Management	802.1Qcc	Stream Reservation Protocol
	802.1Qcw	YANG Model for .1Qbv, .1Qbu et .1Qci
	802.1CBcv	YANG Model for .1CB

Synchronization

The IEEE 802.1AS project has created a profile of the IEEE 1588 Precision Time Protocol (PTP) synchronization protocol for TSN [13]. This covers the maintenance of synchronized time during normal operation and following addition, removal, or failure of network components. The extension includes also an improved scalability and support for long-chains and rings. The redundancy of GrandMaster and paths are also possible. The IEEE 802.1AS is also more responsive with faster GrandMaster changes.

The IEEE 802.1AS favors the real-time capability requirement, since it enables the synchronization of the network, and allows a global planning of frames guaranteeing bounded latencies and null jitters for specific traffic classes. However, there are still some open issues in this standard concerning the detection of a faulty GrandMaster and the switch over procedure from one synchronization tree to another. This procedure has a tremendous impact on the correctness of synchronous scheduling mechanisms.

Scheduling

802.1Qbu: this standard [14] defines a class of service for time-critical frames, that request the transmitter in a bridged Local Area Network to suspend the transmission of a non-time critical frame to be transmitted as soon as possible. When the time-critical frames have been transmitted, the transmission of the preempted frame is resumed. A non time-critical frame could be preempted multiple times. This is done while respecting a minimum fragment size and a pre-defined segment structure.

This standard provides interesting features in terms of real-time capability, since it reduces the blocking time of high-priority traffic due to low-priority one. Nevertheless, activating the frame preemption leads to extra overheads, i.e., 24 bytes per preemption point on preempted frames, which decreases the bandwidth usage efficiency. Moreover, it introduces new safety risks for preempted frames, since we need to compute a specific checksum of each fragment.

802.1Qbv: it is also known as Time-Aware Shaper (TAS) [15]. This standard specifies time-aware queue-draining procedures, that enable bridges and end stations to schedule the transmission of frames based on timing derived from IEEE Std 802.1AS. A transmission gate is associated with each queue and the state of the gate determines whether or not queued frames can be selected for transmission. The TAS uses a periodic static scheduling (known as Gate Control List - GCL) to manage the gate states, and it is generally combined with 802.1Qbu to guarantee the start time of critical traffic transmission.

This synchronous shaper guarantees low bounded latencies and jitters and is deterministic a priori, which fulfills the real-time capability requirements. However, its correct behavior is conditioned by the correctness of the synchronization protocol, which introduces a central point of failure. Although, the synchronization protocol provides redundancy of clock GrandMaster, the effect of electing a new one (when failure) on the TAS behavior has to be analyzed. Finally, TAS requires building a global time schedule for each flow within each bridge, which is known as a NP hard problem [16] increasing the configuration effort and the implementation costs.

802.1Qch: it is also known as Cyclic Queueing and Forwarding (CQF) [17]. This standard specifies synchronized cyclic enqueuing and queue draining procedures that enable bridges and end stations to transmit their frames, while guaranteeing zero congestion loss and deterministic latency. The CQF uses a global time derived from IEEE Std 802.1AS and divided in odd and even phases to manage different traffic classes. If a frame arrives in an odd (resp. even) phase, it can not be sent before the start of the next even (resp. odd) phase.

Like TAS, CQF favors real-time capability requirement and has the same limitations because of the need of global synchronization. However, the guaranteed maximum jitter is equivalent to two phase durations, that can be high since the phase duration has to cover the queuing delay of the busiest output port among all the bridges. This fact introduces a complexity to tune the phase duration, but the configuration effort and implementation costs are lower than with TAS.

802.1Qcr: it is also known as Asynchronous Traffic Shaping (ATS) [18]. This standard aims to guarantee bounded latencies through regulating the flows at each output port and reducing the burstiness. ATS uses a single queue for the aggregate flows arriving from the same input and exiting at the same output of the bridge. The ATS examines only the packet at the head of its FIFO queue and releases it as soon as doing so does not violate the eligibility time of this flow. ATS introduces a layer of shaped FIFO queues (at least the number of input ports in the bridge), that are merged into per-class FIFO queues at the output ports.

Unlike TAS and CQF, ATS provides independence from clock synchronization protocol in terms of performance and reliability; in addition the configuration procedure is much simpler since we do not need the information on the end-to-end path of each flow and it is done locally in each output port. Hence, ATS favors real-time capability (bounded latencies), in addition to decreasing the configuration effort. However, ATS does not guarantee null jitter and the implementation of the shaped FIFO queues within the bridges increases the implementation costs. Finally, since this standard is based on eligibility time, we need to analyze the impact of the variation of the local clock within each device, as well as the inter-device clock deviation on its performance. These points have been detailed in the standard [18] in Annex V. The proposed solution is to increase the required burst and rate of the contract of each flow at each hop, to compensate for the deviation with reference to the precedent one. However, this solution increases the configuration effort, since the flow contract depends on the position of the bridge along the flow path.

TABLE II
BENCHMARKING OF TSN SCHEDULING MECHANISMS

	Bounded latency	**Null jitter**	**Configuration Effort**	**Implementation costs**
TAS	yes	yes	high	high
CQF	yes	no	medium	medium
ATS	yes	no	medium	medium

The pros and cons of each TSN scheduling mechanism are summarized in Table II, when taking into account the main real-time capability metrics (latency and jitter) in addition to costs aspects (configuration effort and implementation costs).

Reliability

802.1CB: it is also known as Frame Replication and Elimination for Reliability (FRER) [8]. This standard specifies procedures for bridges and end stations that provide the identification and replication of frames for redundant transmission, as well as the elimination of duplicate frames. The FRER mechanism needs a stream identification function for replicate frames to generate sequence numbers. This sequence number is useful to eliminate duplicate frames afterwards. The FRER mechanism can be implemented in end stations and also intermediate nodes in the network, called relay elements. For bridges that do not implement FRER, the stream identification function is needed to forward the duplicate frames on the associated ports.

This standard enables the increase of reliability level of specific flows that do not tolerate loss, which favors the safety requirement. However, FRER can induce out of order phenomena of duplicate flows, which breaks the FIFO property of these flows. This fact is problematic if there is an ATS scheduler downstream the elimination process of FRER, since ATS requires the FIFO property to behave correctly. Moreover, FRER mechanism can lead to latency increase within the bridges implementing such a mechanism, since the worst-case traffic burst is the sum of duplicate flows bursts. This problem has been detailed in the Annex of the standard [8].

802.1Qci: it is also known as Per-Stream Filtering and Policing [19]. This standard specifies a policing mechanism at bridge input to perform frame counting, filtering and policing of data streams, based on a particular data stream identifier. Policing and filtering functions include the detection and mitigation of disruptive transmissions in a network, improving its robustness.

This standard improves the reliability level of the system by avoiding the transmission of non-conformant frames; thus favors the safety requirement. However, the impact of the inter-device clock deviation has to be analyzed since it can lead to discard a conformant frame.

802.1Qca: it specifies Path Control and Reservation (PCR) mechanism [20]. This standard provides some extensions for path control, bandwidth assignment and redundant path computation for data flows. It extends the use of IS-IS protocol to transmit control information about synchronization and scheduling, we talk about ISIS-PCR. It is based on a Path Control Element (PCE) to compute explicit paths. The PCE is an external entity to IS-IS, capable of computing a path through a database containing a representation of the network topology. This database is built based on the received information from the PCA (Path Control Agent) of the network.

This standard improves the reliability level through isolating faults to specific regions in the network when computing the flow paths, and reserving multiple disjoint paths for flows submitted to IEEE 802.1CB; thus it favors the safety requirement. Moreover, it fulfills reconfiguration requirement since it enables the computation of new paths if needed, e.g., failure, modification of the topology, adding a new flow. However, the algorithm to compute explicit paths is not specified in the standard and is an open issue.

Resource Management

802.1Qcc: it specifies Stream Reservation protocol (SRP) Enhancements [9]. This standard defines a configuration model and the recommended one for industrial automation is the fully centralized configuration model. Talkers and listeners (applications) send their requirements to a centralized User Configuration (CUC) entity. Afterwards, the CUC communicates these requirements to a Centralized Network Configuration (CNC) entity. The latter provides a decision on the guarantee of the requirements for each application and how to meet them; thus the CNC is responsible for configuring the network to meet the talkers and listeners requirements.

This standard favors modularity and reconfigurability requirements since it enables the reconfiguration for the system under modifications of topology or flows. However, the implementation of the CNC entity is not specified in the standard and is an open issue to optimize the network performance and reliability.

802.1Qcw & 802.1CBcv: Known as YANG Models, specify an XML-based format to allow configuration and status reporting for bridges and end stations with the capabilities of FRER, TAS, CQF and frame preemption, respectively.

These standards enable the interoperability between different devices and the reconfiguration of some mechanisms. However, they are still under extensions to cover all the TSN mechanisms.

TABLE III
TSN STANDARDS VS REQUIREMENTS

Key component	Pros	Cons
Synchronization	Favors real-time capability	Central point of failure for synchronous schedulers
Scheduling	Favors real-time capability	*Introduce overhead and safety risks (.1qbu) *Instability if synchronization fails and high configuration effort (.1qbv) *Instability if synchronization fails and do not guarantee null jitter (.1qch) *Sensitive to variation of local clocks within devices and instability if no FIFO property for the flows (.1qcr)
Reliability	Favors safety	*Induce out of order and instability of downstream ATS scheduler (.1CB) *Risk of discarding conformant frames due to the inter-device clock deviation (.1qci)
Resource Management	Favors modularity and interoperability	The implementation of CNC entity is an open issue (.1qcc)

The assessment of the TSN components, recommended in [6], vs the main Industry 4.0 requirements as well as the main identified limitations are summarized in Table III. It is worth noticing that each TSN mechanism achieves specific requirements, while inducing some limitations on the whole network. Based on this qualitative analysis, our objective is to specify a TSN-compliant network to take advantages of the recommended TSN mechanisms, while avoiding the identified warning points. Therefore, FactoRing aims to bridge the gap between the aforementioned TSN standards to guarantee the Industry 4.0 requirements, while decreasing the configuration effort and the implementation costs.

III. FACTORING SPECIFICATIONS

In this section, we first present the main features of Factoring and the supported topologies. Afterwards, we detail the scheduling mechanisms, the QoS management and resource management.

A. Main Features

FactoRing is a TSN-compliant network at 1Gbit/s, based on an interface called T-FactoRing or T for short, that allows any Ethernet-compliant equipment to exchange data via the network. FactoRing supports two kinds of topology as shown in Figures 1 and 2. The first one is called the simple mono-ring, where T-s are connected in a daisy-chain mode using the ring ports. The second one is called multiple-ring, where the different rings of T-s are interconnected with redundant TSN switches via redundant Gateways (G) to increase robustness. The Gateway is a particular T relaying the traffic from the ring to the switch and vice versa.

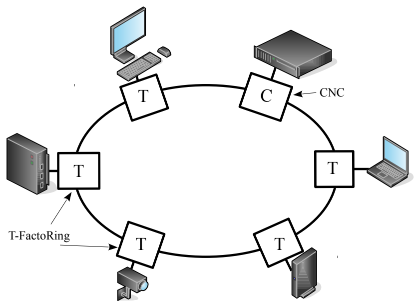

Fig. 1. An example of simple mono-ring topology of FactoRing

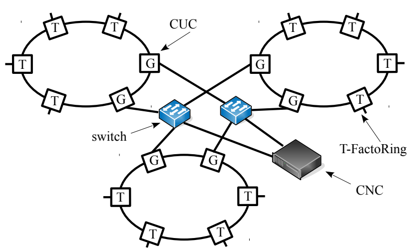

Fig. 2. An example of multiple-ring topology of FactoRing

The T-FactoRing is a TSN-compliant three ports Full Duplex Gigabit Ethernet switch, illustrated in Figure 3, where ports 1 and 2 are the ring ports and port 3 is the equipment port. The T- FactoRing has the following main characteristics:

- Cut-Through forwarding technique: the T-Factoring starts forwarding the packet just after its identification, i.e. only the header of each packet is decoded to determine its destination port. This technique guarantees shorter transmission latency than the Store and Forward technique, which waits until the complete reception of the packet before forwarding it to the destination port. It is worth noting that the erroneous frames will be discarded at the final destination, since there is no CRC check at the intermediate nodes when using Cut-Through technique;
- Non-Preemptive Class-based Strict Priority queuing: The T-Factoring supports 5 priorities (5 is the highest priority) as follows: Control traffic (5), Hard Real-Time (HRT) with strict time constraints (4), HRT with strict jitter constraints (3), Soft Real-Time (2) and Best-effort (1). To avoid the safety risks and extra overheads identified in Section II-B, the TSN preemption mechanism [14] has been discarded;
- Traffic policing and filtering: the T-FactoRing implements traffic policing and filtering mechanisms [19], as recommended in [6], to avoid disruptive transmissions and network saturation. These mechanisms are considered only at the input port 3 connecting the equipment. This restriction is done to avoid the problem of discarding conformant frames due to the inter-device clock deviation explained in Section II-B.
- Traffic shaping and damping: the T-FactoRing implements traffic shaping and damping [21] before the multiplexing stage at each output port. These mechanisms enable guaranteed latencies and null jitter and are detailed in Section III-B;
- FRER mechanism: the T-Factoring implements FRER mechanism [8] only at port 3 (connecting the equipment). This choice has been done to restrict the management of the ”out of order” due to the Elimination function of FRER only in the output port connected to the equipment. This fact will avoid packets ”out of order” in the middle of the network, which can induce instability problems or performance degradation within the shapers. As shown in Figure 3, the replication function is implemented at the input port 3 to identify the duplicate frames and the elimination function is implemented at the output port 3 to discard one of the duplicate;

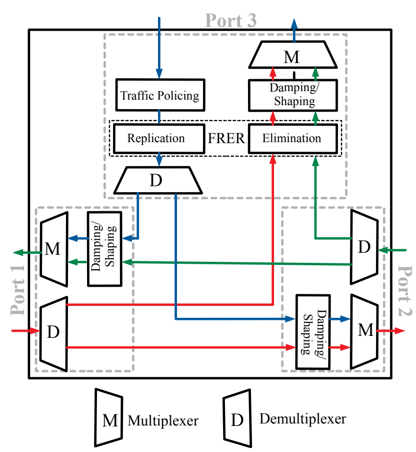

Fig. 3. T-Factoring architecture

B. Scheduling mechanisms

As discussed in Section II-B, the synchronization protocol in TSN is commonly used to guarantee null jitter and to enable global planning for synchronous schedulers. However, it is considered at the same time a central point of failure for synchronous schedulers.

To cope with this limitation, FactoRing is based on asynchronous shaping to guarantee bounded latencies, and damping mechanism [21] to guarantee null jitters. Both mechanisms do not require a synchronization protocol or global planning, which avoids the central point of failure and decreases the configuration effort. The T-Factoring implements at each output port shaping and damping functions, as shown in Figure 3, that are detailed herein.

Per-flow Shaping

To provide bounded latencies, the common solution for asynchronous networks consists in reshaping flows inside the bridges to decrease the impact of interferences with other flows. Reshaping is achieved by means of traffic regulators, which are enabled before the multiplexing stage at the output port. There are two kinds of regulators: the per-flow regulator and the interleaved regulator [22]. The former manages flows individually and a common implementation is the Linux's Token-Bucket Filter; whereas the latter manages the flow aggregates and ATS [18] is an example of such a regulator.

In [23], the authors have proved that in loosely synchronized or non-synchronized networks, these traffic regulators are sensitive to the inter-device clock deviation. This fact can lead to unbounded delays if not addressed. This problem has been pointed out for ATS [18] and the proposed solution is to increase the burst and rate of each flow's contract in a very specific manner along its path. This solution leads to configuring the ATS parameters for each flow depending on its position along the path, which highly complexifies the configuration process.

On the other hand, the per-flow regulator can cope with the inter-device clock deviation issue in a very simple way, through scaling the maximum rate guaranteed to each flow independently from its position along the path. Moreover, the scaling parameter depends only on the clock stability bound. For instance, based on the established bound in [13], the guaranteed maximum rate of each flow has to be increased by less then 1% to guarantee the stability of per-flow regulators and enable computing bounded delays.

Therefore, for FactoRing, the shaping mechanism is based on per-flow regulators that guarantee bounded latencies and decrease the configuration effort, with reference to interleaved regulators like ATS.

Damping mechanism

To guarantee null jitter, T-Factoring implements damping mechanism before the multiplexing stage at the output port. Similarly to shaping, traffic damping consists in preserving the original arrival pattern of packets within each hop. However, the traffic damper will in addition absorb the jitter introduced by the upstream output port by holding each packet until its eligibility time, i.e., the time when the packet will be enqueued into the multiplexing stage. The eligibility time of each packet is computed in such a way to cancel the difference between its maximum delay and its actual delay in the upstream output port. The actual delay of each packet within the multiplexer is written in the packet's header, to be read in the next hop by the damper and to enable the computation of eligibility time. However, the maximum delay is a damper's parameter tuned by the CNC entity of Factoring, as explained in Section III-D.

C. QoS Management

FactoRing guarantees QoS management through the implementation of Non-Preemptive Class-based Strict Priority queuing, which supports five traffic classes as follows:

- Control traffic class: This traffic is generated at the T level and the supervisor level (CNC entity in Figure 2) to enable reconfiguration management. This traffic has the highest priority level (5) to reduce the reconfiguration time in case of failure, thus increasing the availability level of Factoring. This traffic is submitted to shaping mechanism at the output ports;
- HRT traffic with strict latency constraint: This traffic has the second highest priority level (4) and is generated by real-time applications with strict latency constraints. This type of data flow is sent on both ports 1 and 2. It is submitted to shaping at each output port and to FRER mechanism at port 3, i.e., replication at the input port 3 of the source and elimination at the output port 3 of the destination;
- HRT traffic with strict jitter constraint: This traffic has the third highest priority level (3) and is generated by real-time applications with strict jitter constraints. This type of data flow is sent on both ring ports. It is submitted to damping at each output port, which in addition to bounding the traffic as the shaping mechanism it bounds the jitter. This class of traffic is also submitted to FRER mechanism at port 3 similarly to traffic of priority 4;
- SRT traffic class: This traffic is mainly sent by soft real-time applications, such as audio or video transfers, and has the medium priority level (2). This type of data flow is sent on the ring port corresponding to the shortest path and is submitted to shaping at the output port;
- NRT traffic class: This traffic corresponds to non-real-time applications, such as file transfer, and has the lowest priority level (1). This type of data flow is sent on the ring port corresponding to the shortest path and is submitted to shaping at the output port;

It is worth noticing that all the traffic classes generated by the equipment are submitted to traffic policing and filtering at the input port 3. The mechanisms applied to each traffic class are summarized in Table IV.

D. Resource Management

FactoRing implements the fully centralized configuration mechanism, as recommended in [6]. For instance, as shown in Figure 2, the applications send their requirements via the T-FactoRings to the gateways, which are the CUC entities. Afterwards, the gateways communicate these requirements to the central supervisor (CNC entity) connected to both switches.

Finally, the central supervisor provides the different parameters to tune along each flow's path to guarantee its require-

TABLE IV
QOS MANAGEMENT WITHIN T-FACTORING

	Priority	Routing	Damping	Shaping	FRER	Filtering
Control traffic	5	both ports	no	yes	no	no
HRT with strict latency	4	both ports	no	yes	yes	yes
HRT with strict jitter	3	both ports	yes	no	yes	yes
SRT	2	one port	no	yes	no	yes
NRT	1	one port	no	yes	no	yes

ments. These parameters concern for instance the shaping and damping parameters to compute eligibility times.

IV. PERFORMANCE ANALYSIS

In this section, we investigate the main performance metrics of Factoring through a synthetic case study of a single ring. These metrics have been computed with WoPANets tool [24] based on Network Calculus framework [25]. These preliminary results are discussed to highlight the promises of Factoring to fulfill the real-time requirement of Industry 4.0.

A. Test case and metrics

We consider the case study with the following assumptions:

- The network topology is a simple ring;
- The link speed is 1 Gbit/s;
- The network size varies from 5 to 50 end-stations with a step of 5;
- The end-stations are similar and send the same traffic to a supervisor;
- Each T-FactoRing has a technological latency of 500ns;
- Each T-FactoRing has a memory of 1Mbytes, equally partitioned between the 3 ports;
- Each end-station generates three types of traffic as described in Table V: the I/O data, Audio/Video (A/V) data, and monitoring data. For instance, the maximum utilization rate for the network of 50 end-stations is around 40%.

The main idea is to assess the impact of the network size on the main performance metrics: end-to-end latency, jitter and backlog. To highlight the promises of FactoRing, we conduct a comparative analysis with a baseline solution. The baseline solution consists of disabling the damping and shaping mechanisms within the crossed T-Factoring, but keeping the Strict Priority Scheduler within the output ports.

B. Numerical results

Figure 4 illustrates the maximum end-to-end jitters for I/O data under FactoRing and baseline solutions. There are mainly two interesting observations through this figure. The first one confirms the benefit of damping mechanism to guarantee null jitter under FactoRing, independently from the network size, a key real-time requirement for Industry 4.0. The second observation concerns the network scalability, where the maximum number of interconnected end-stations respecting the jitter constraint (0.2ms) doubles with FactoRing (50), in reference with the baseline solution (only 25). These results show the high ability of FactoRing to guarantee null jitter independently from the network size, which is a promising scalability feature for Industry 4.0.

TABLE V
TRAFFIC CHARACTERISTICS.

	Priority	Payload (byte)	Period (ms)	Deadline (ms)	Jitter (ms)
I/O data	3	16	2	2	0.2
A/V	2	20*1000	20	20	N/A
Monitoring	1	1500	100	infinity	N/A

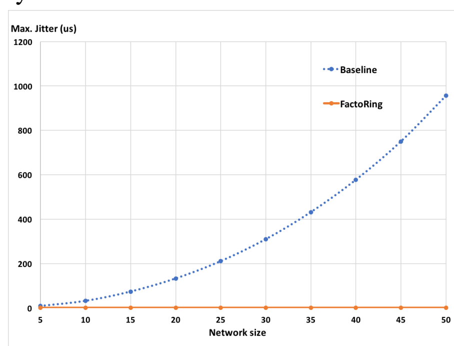

Fig. 4. Maximum jitters for I/O traffic with FactoRing vs the Baseline

To assess the real-time capability of FactoRing in terms of bounded latencies, the maximum end-to-end latencies of A/V traffic are illustrated in Figure 5 when varying the network size. The results show the benefit of using shaping mechanism within each T-Factoring to avoid the burstiness propagation of the flows, and consequently decrease the queueing delays within each crossed T-FactoRing. For instance, the maximum end-to-end latency for a network of 50 end-stations is 14 times lower under FactoRing than under the baseline solution. Moreover, the results confirm the high scalability of FactoRing, where the maximum number of end-stations respecting the deadline constraint of A/V traffic (20ms) is only 25 with the baseline solution and at least 50 with FactoRing.

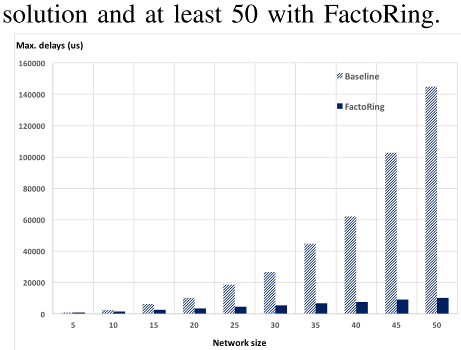

Fig. 5. Maximum delays for A/V traffic with Factoring vs the Baseline

Figure 6 shows the backlog of I/O data within each crossed node along the path of the flow generated by the end station in the middle of a network of 20 end-stations (the 10th position) and received by the supervisor. We focused on the I/O data backlog to verify that there is no risk of loosing data due to buffers overflow, and consequently the availability requirement of this traffic class. Obviously the backlog increases with the number of hops along the path and is always respecting the memory size, i.e., no risk of overflow. As it can be noticed, the backlog is lower with FactoRing than with the baseline. This is mainly due to the non-propagation of the burstiness along the path because of damping and shaping mechanisms in FactoRing. These results show the interest of FactoRing to improve the memory usage, and consequently decreasing the implementation costs, i.e., less memory.

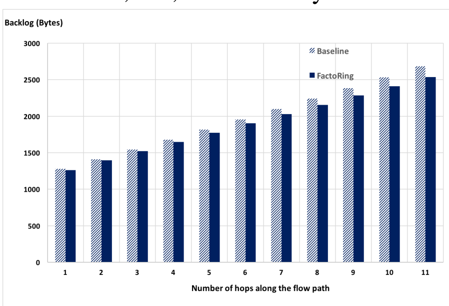

Fig. 6. Maximum backlogs for an I/O flow along its path with Factoring vs the Baseline

This preliminary performance evaluation of FactoRing shows that the results are encouraging to pursue the line through providing deeper analyses for representative industrial use cases.

V. Conclusions

To meet the emerging requirements of industry 4.0 and improve interoperability between devices from different manufacturers, TSN Ethernet has been revealed as an appealing solution in this domain. In this paper, the effectiveness of the TSN mechanisms, recommended in the draft of the TSN profile for industrial automation [6], has been assessed vs the main requirements of Industry 4.0. Afterwards, based on this qualitative analysis, the specifications of FactoRing, an asynchronous TSN-compliant network guaranteeing the main requirements without the need of global synchronization and complex network planning, have been detailed. The preliminary results have shown the ability of FactoRing to guarantee null jitter, low bounded latencies and improved memory usage, key performance metrics for Industry 4.0.

Currently, FactoRing consortium is working on the evaluation of FactoRing on industrial use cases, and on HW implementation of such proposal with open source specifications to facilitate its adoption in the market.

Acknowledgment

The authors would like to thank Oana Hotescu and Ludovic Thomas for their valuable feedbacks during the project meetings, and Thierry Leydier for his support to implement FactoRing plugin within WoPANets tool.

References

[1] V. Koch, S. Kuge, R. Geissbauer, and S. Schrauf, “Industry 4.0: Opportunities and challenges of the industrial internet,” 2014.

[2] MarketsandMarkets , “Motion Control Market.” [Online]. Available: https://www.marketsandmarkets.com/PressReleases/motion-control.asp

[3] M. Schumacher, J. Jasperneite, and K. Weber, “A new approach for increasing the performance of the industrial Ethernet system PROFINET,” in *WFCS*, 2008.

[4] “EtherCat - the Ethernet Fieldbus, URL:”www.ethercat.org”.”

[5] IEEE TSN Task Group, “TSN Specifications.” [Online]. Available: http://www.ieee802.org/1/pages/tsn.html

[6] IEC/IEEE, “IEC/IEEE 60802 TSN Profile for Industrial Automation.” [Online]. Available: https://1.ieee802.org/tsn/iec-ieee-60802/

[7] *IEC 61784-2, Digital data communications for measurement and control - Part 2: Additional profiles for ISO/IEC 8802-3 based communication networks in real-time applications*, 2010.

[8] IEEE Standard, “IEEE 802.1CB-2017 - IEEE Standard for Local and metropolitan area networks–Frame Replication and Elimination for Reliability,” 2017.

[9] ——, “IEEE 802.1Qcc-2018 - IEEE Standard for Local and Metropolitan Area Networks–Bridges and Bridged Networks – Amendment 31: Stream Reservation Protocol (SRP) Enhancements and Performance Improvements,” 2018.

[10] M. M. Mabkhot, A. M. Al-Ahmari, B. Salah, and H. Alkhalefah, “Requirements of the smart factory system: A survey and perspective,” *Machines*, vol. 6, no. 2, 2018. [Online]. Available: https://www.mdpi.com/2075-1702/6/2/23

[11] IEEE 802.1 Working Group, “IEEE 802.1 Specifications.” [Online]. Available: https://1.ieee802.org/

[12] IEEE Standard, “ IEEE 802.3-2018 - IEEE Standard for Ethernet ,” 2018.

[13] ——, “ IEEE 802.1AS-2020 - IEEE Standard for Local and Metropolitan Area Networks–Timing and Synchronization for Time-Sensitive Applications,” 2020.

[14] ——, “IEEE 802.1Qbu-2016 - IEEE Standard for Local and metropolitan area networks – Bridges and Bridged Networks – Amendment 26: Frame Preemption ,” 2016.

[15] ——, “ IEEE 802.1Qbv-2015 - IEEE Standard for Local and metropolitan area networks – Bridges and Bridged Networks - Amendment 25: Enhancements for Scheduled Traffic,” 2015.

[16] S. S. Craciunas, R. S. Oliver, M. Chmelik, and W. Steiner, “Scheduling real-time communication in IEEE 802.1 Qbv time sensitive networks,” in *RTNS*, 2016.

[17] IEEE Standard, “IEEE 802.1Qch-2017 - IEEE Standard for Local and metropolitan area networks–Bridges and Bridged Networks–Amendment 29: Cyclic Queuing and Forwarding ,” 2017.

[18] ——, “IEEE 802.1Qcr-2020 - IEEE Standard for Local and Metropolitan Area Networks–Bridges and Bridged Networks - Amendment 34:Asynchronous Traffic Shaping ,” 2020.

[19] ——, “IEEE 802.1Qci-2017 - IEEE Standard for Local and metropolitan area networks–Bridges and Bridged Networks–Amendment 28: Per-Stream Filtering and Policing ,” 2017.

[20] ——, “ IEEE 802.1Qca-2015 - IEEE Standard for Local and metropolitan area networks– Bridges and Bridged Networks - Amendment 24: Path Control and Reservation ,” 2015.

[21] D. Verma, H. Zhang, and D. Ferrari, “Delay jitter control for real-time communication in a packet switching network,” in *Proceedings of TRICOMM ‘91: IEEE Conference on Communications Software: Communications for Distributed Applications and Systems*, 1991.

[22] J.-Y. Le Boudec, “A theory of traffic regulators for deterministic networks with application to interleaved regulators,” *IEEE/ACM Transactions on Networking*, vol. 26, no. 6, pp. 2721–2733, 2018.

[23] L. Thomas and J.-Y. Le Boudec, “On time synchronization issues in time-sensitive networks with regulators and nonideal clocks,” *Proc. ACM Meas. Anal. Comput. Syst.*, vol. 4, no. 2, Jun. 2020.

[24] A. Mifdaoui and H. Ayed, “WOPANets : a tool for WOrst case Performance Analysis of embedded Networks,” in *CAMAD 2010*.

[25] J.-Y. Le Boudec and P. Thiran, *Network calculus: a theory of deterministic queuing systems for the internet*. Springer Science & Business Media, 2001.