Asynchronous Time-Sensitive Networking for Industrial Networks

Jonathan Prados-Garzon, Lorena Chinchilla-Romero, Pablo Ameigeiras, Pablo Muñoz, and Juan M. Lopez-Soler

jpg@ugr.es, lorenachinchilla@ugr.es, pameigeiras@ugr.es, pabloml@ugr.es, juanma@ugr.es

*Department of Signal Theory, Telematics and Communications (DTSTC), University of Granada, Spain.

Abstract—Time-Sensitive Networking (TSN) is expected to be a cornerstone in tomorrow's industrial networks. That is because of its ability to provide deterministic quality-of-service in terms of delay, jitter, and scalability. Moreover, it enables more scalable, more affordable, and easier to manage and operate networks compared to current industrial networks, which are based on Industrial Ethernet. In this article, we evaluate the maximum capacity of the asynchronous TSN networks to accommodate industrial traffic flows. To that end, we formally formulate the flow allocation problem in the mentioned networks as a convex mixed-integer non-linear program. To the best of the authors' knowledge, neither the maximum utilization of the asynchronous TSN networks nor the formulation of the flow allocation problem in those networks have been previously addressed in the literature. The results show that the network topology and the traffic matrix highly impact on the link utilization.

I. INTRODUCTION

Time-Sensitive Networking (TSN) has been specified by IEEE 802 to cover the needs of many sectors (e.g., manufacturing, healthcare, and energy industry) for the transport of critical streams with bounded delay, jitter, and packet loss. Specifically, TSN is a set of standards that define a converged layer 2 (L2) network technology that ensures the deterministic transport of the streams via IEEE 802 networks [1]. TSN harmonizes the principles of both circuit and packet switched networks to convey virtually any kind of traffic in an efficient way.

Today's industrial networks are based on Industrial Ethernet networks to transport the critical traffic and WiFi to provide wireless access to best-effort (BE) traffic. Industrial Ethernet consists of a wide variety of special-purpose technical standards and proprietary solutions (e.g., EtherCAT, Ethernet POWER-LINK)to provide deterministic Quality of Service (QoS), which makes quite difficult the interoperation of the different network segments. Moreover, this type of networks might be prohibitively expensive and hinder scalability. Due to all limitations of the current industrial networks, TSN is expected to replace Industrial Ethernet in tomorrow's industrial domains.

Many bridges implementations defined in TSN standards are synchronous (e.g., IEEE 802.1Qch Cyclic Queuing and Forwarding (CQF) and IEEE 802.1Qbv Time-Aware Shaper (TAS)) [1], i.e., it is needed a precise and common time reference shared among all the TSN devices of the network. The TSN devices employ this time synchronization, for instance, to schedule the transmission of the traffic for the different flows over synchronized time slots (i.e., time-division multiplexing). However, the need for a network-wide coordinated time hampers the network scalability. Also, the use of reserved time slots for each flow or set of flows leads to poor utilization of the link capacities. TSN standards tackle these weaknesses by including

the IEEE 802.1Qcr Asynchronous Traffic Shaper (ATS), based on the Urgency-Based Scheduler (UBS) proposed by Specht and Samii [2], which is asynchronous and offers high link utilization by better leveraging statistical multiplexing.

ATS is the building block for realizing asynchronous TSN network able to support critical services and its coexistence with BE traffic. Traditionally, asynchronous schedulers have relied on traffic prioritization (strict priorities) at the bridge's egress ports for QoS support. The main drawback of this approach is that it results in arbitrarily large worst-case delays as the burstiness of the flows grows at every hop [3]. This issue can be solved by using per-hop shaping. To that end, ATS introduces the interleaved shaping concept, which allows the use of a single queue for regulating a set of flows, each with its own shaping constraints. In this way, ATS enables per-hop traffic regulation in a cost-effective way.

The goal of this paper is to evaluate the capacity the ATS-based networks have to accommodate industrial traffic flows, which has not been addressed by previous works. For this purpose, we formally formulate the flow allocation problem in asynchronous TSN networks. We apply the required transformations (e.g., convexification and linearization techniques) to convert the original flow allocation problem into a convex mixed-integer linear program. In this way, we can solve it using commercial solvers lik Mosek. To the best of the authors' knowledge, this is the first work to formulate the flow allocation problem in asynchronous TSN networks. Please note that we do not pretend to cope with the problem computational complexity but to formally formulate the flow allocation problem as it is. Finally, we consider a realistic industrial scenario to evaluate the attainable utilization of the asynchronous TSN network while the QoS requirements of all the flows are enforced. Specifically, we contemplate a realistic compound traffic model and three typical industrial network topologies (e.g., star, daisy chain, and ring [4]). Results show that the link utilization is severely affected by the network topology and the traffic matrix, i.e., the traffic load generated by any source-destination (S-D) pair.

The remainder of the paper is organized as follows: Section II overviews the existing works that address the performance evaluation of the ATS-based networks. Section III describes the system model and the ATS operation.Section IV states the flow allocation problem in ATS-based networks. Section V formulates the optimization problem for the flow allocation in ATS-based networks as a convex mixed-integer non-linear program. Section VI includes the results showing the performance of an industrial asynchronous TSN network. Last, section VII draws the main conclusions.

II. Related Works

This section overviews the existing works related to asynchronous TSN networks.

The performance analysis of the ATS has been addressed in several works [2], [3], [5]. In [2], Specht and Samii propose the Urgency-Based Scheduler (UBS) as a practical and cost-effectively alternative to the existing asynchronous shapers to enable per-flow traffic regulation. The ATS can be regarded as the UBS realization. Considering the streams are regulated by a leaky bucket, they derive delay bounds for the UBS assuming links with constant capacity. In [3], Le Boudec formally derives the performance bounds of the UBS and introduces the Minimal Interleaved Regulator concept, which expands UBS to a broader set of regulation rules. Remarkably, Le Boudec demonstrates that any minimal interleaved regulator, placed after any arbitrary FIFO queue, does not negatively impact on the worst-case delay of the combination [3]. Mohammadpour *et al.* provide the end-to-end (E2E) worst-case analysis of an ATS-based TSN network. The derived E2E bounds are tighter than those estimated as the sums of per-hop worst-case delay.

The performance evaluation of the ATS is faced through simulation in [6]–[8]. Zhou *et al.* make a review of the operation and bounds of the ATS that were already addressed in [6]. In addition, they evaluate the performance of the ATS. They conclude that ATS accomplishes effective traffic shaping and switching without requiring a global notion of time. Also, they identify the pros and cons of the asynchronous algorithms UBS/Token Bucket Emulation (TBE), UBS/Length Rate Quotient (LRQ), and Paternoster scheduler. The work in [7] compares the performance exhibited by ATS and TAS based networks configured in a ring topology for conveying the traffic of industrial control services. The simulation results suggest that ATS slightly outperforms TAS for sporadic deterministic and non-performance sensitive traffic. Nonetheless, ATS offers higher scheduling delays than TAS [7]. What is more, the heavier the BE traffic is, the more difficult the streams' E2E delay assurance is for the ATS-based network. In [8], Grigorjew *et al.* propose a simulation framework, featuring ATS and IEEE 802.1Qbu frame preemption. Their results suggest a decrease in the jitter of the high priority flows on account of frame preemption, though it unfavorably impacts the latency of the low priority flows. Unlike the works mentioned above, in this article, we evaluate the capacity of an ATS-based network to accommodate streams of industrial services.

Finally, the synthesis of queues and priority assignment for a network that implements UBS is addressed in [9], [10]. In [9], Specht and Samii consider two approaches to get to the bottom of the problem: i) a pure Satisfiability Modulo Theories (STM) solver which always finds a feasible solution, and ii) a heuristic approach dubbed Topology Rank Solver (TRS) to cope with the high complexity exhibited by the pure STM solver. They maximize several variants of the delay surplus. In [10], Prados-Garzon *et al.* propose a deep reinforcement learning-based solution for the flow allocation problem in asynchronous TSN networks. They leverage ATS analytical performance models to check the validity of the actions issued by the agent. In this way, the RL-based solution becomes fully reliable. However, none of
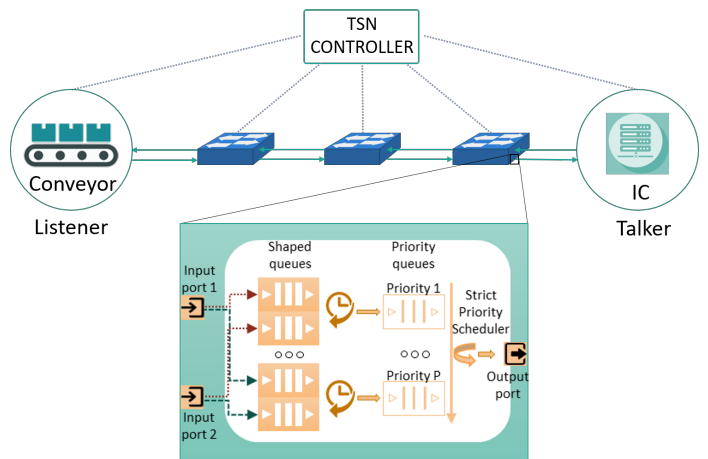

Fig. 1: Fully centralized TSN architecture and ATS architecture and operation.

the above works explicitly include a model of the flow allocation problem in asynchronous TSN networks. In this work, we cover that gap.

III. System Model

A. Network Model

Let us assume an asynchronous TSN network consisting of V ATS bridges interconnected through E simplex links. This network can be modeled as a directed graph (or digraph) $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ and $\mathcal{E}$ denote the vertices (bridges) and the edges (links) of the graph, respectively. The weight $C^{(e)}$ of each edge $e \in \mathcal{E}$ stands for the link capacity. Let $\mathcal{A} \subseteq \mathcal{V}$ be the set of access bridges, i.e., TSN switches that connect directly with the talkers (sources) and listeners (destinations) endpoints. We consider there is a set of predefined paths $\mathcal{P}_{s,d}$ to interconnect every source $s \in \mathcal{A}$ and destination $d \in \mathcal{A}$ pair. A path $p \in \mathcal{P}_{s,d}$ comprises a sequence of adjacent links, i.e., links connected by a TSN bridge.

Each TSN bridge v includes $I^{(v)}$ input ports and $O^{(v)}$ output ports. There is an ATS, whose operation is detailed in the next subsection, at each output port handling the transmission of the frames over a link. Please observe that there is a one-to-one correspondence between the ATSs and the links. Each ATS e has $S^{(e)}$ shaped buffers and $P^{(e)}$ priority queues.

There are Q types or classes of traffic in the network. The E2E delay budget D_q of each traffic class $q \in \mathcal{Q}$ is known in advance. Every traffic flow (or stream) f in the network, regardless of its type of traffic, is regulated and upper constrained by the function $r_f \cdot t + b_f$, where r_f and b_f are the sustainable rate and burstiness (or burst size) of the flow, respectively. These parameters correspond to the committed information rate and burst size if the flow is allocated in the network, respectively. We assume that all the flows with the same traffic type q have the same burstiness b_q. We define r_q as the mean sustainable rate of the flows with traffic type q. Also, we presume all the flows belonging to the same service q have the same lifetime or duration τ_q.

The maximum delay experienced by a flow f when it traverses an ATS is derived in [2] by Specht and Samii. From this bound, the E2E worst-case delay D_f to convey any packet

of the flow f from its source s to its destination d when passes through the path $\mathcal{P}_{s,d} \subseteq \mathcal{E}$ is given by:

$$D_f \leq \sum_{e \in \mathcal{E}_f} d_{f,e,p_e}^{(max)} = \sum_{e \in \mathcal{P}_{s,d}} \left(\frac{\hat{b}_{e,p_e}^{(H)} + \hat{b}_{e,p_e}^{(S)} + \hat{l}_{e,p_e}^{(L)}}{C_e - \hat{r}_{e,p_e}^{(H)}} + \frac{l_f}{C_e} \right)$$

(1)

where $\hat{b}_{e,p}^{(H)} = \sum_{f \in \mathcal{F}_{e,p}^{(H)}} b_f$ and $\hat{r}_{e,p}^{(H)} = \sum_{f \in \mathcal{F}_{e,p}^{(H)}} r_f$ respectively denote the aggregated burstiness and data rate generated by the set of flows $\mathcal{F}_H$ with higher priority level than the flow f at link $e \in \mathcal{P}$, $\hat{b}_{e,p_e}^{(S)} = \sum_{f \in \mathcal{F}_{e,p_e}^{(S)}} b_f$ is the burstiness of the set of flows $\mathcal{F}_{e,p_e}^{(S)}$ with the same priority level as the flow f at link $e \in \mathcal{P}$, $\hat{l}_{e,p_e}^{(L)}$ represent the maximum packet size for the set of flows with lower priority levels than the flow f, l_f is the maximum packet size of the flow f, and C_e is the transmission capacity at a given ATS (link) e.

We consider a fully centralized (Software-Defined Networking (SDN)-like) architecture for the TSN network as defined in IEEE Std 802.1Qcc-2018 (see Fig. 1). Specifically, there is a logically centralized TSN controller in charge of controlling and monitoring the different TSN bridges. The TSN controller is in charge of running an admission control mechanism to decide whether each incoming flow is accepted or rejected. It is also responsible for properly configuring the ATSs if required, e.g., when an incoming flow is accepted, or a long-term configuration is computed for the network like the mapping between traffic classes onto priority levels at every ATS.

B. ATS Model and Operation

The ATS defines an asynchronous method for handling the frames at the egress ports of the TSN bridges [1]. The ATS specified in IEEE P802.1Qcr standard is based on the UBS originally proposed by Specht and Samii in [2]. In this work, we adopt the nomenclature used in [2] for describing the ATS operation.

Figure 1 depicts the ATS queuing model. For the sake of simplicity, Fig. 1 shows only one egress port, but please note that there is an ATS instance per bridge egress port. The ATS consists of two queuing stages: i) a set of shaped queues for interleaved shaping, and ii) a set of priority queues. All these queues follow a First Come, First Served (FCFS) discipline.

The interleaved shaping enables the use of a single queue (shaped queue) for realizing the traffic regulation of a set flows, each with its own constraints. To that end, only the eligibility of the head-of-line (HOL) frame is checked in order to decide its transmission according to the regulation constraints of its flow. If so, the frame is released for transmission to the following queuing level.

ATS supports leaky bucket shaping constraints. It enforces an upper bound on the flows of the form $A_f(t) \leq r_f \cdot t + b_f$ [2], [9]. Where $A_f(t)$ is the accumulated amount of transmitted data until the instant t for the flow f, and r_f and b_f are respectively the enforced sustainable rate and burstiness. The parameters r_f and b_f are also referred to as Committed Information Rate and Committed Burst Size in TSN standards.

The second stage in the queuing hierarchy includes one FCFS queue per priority level in the scheduler. Each queue merges the output of all shaped queues assigned to the same priority

level. The transmission selection algorithm at this stage is strict priorities.

The allocation of a flow to a given ATS involves two major decisions, e.g., i) flow to shaping queue, and ii) flow to priority level assignments. These decisions are subject to the following rules: each shaped queue is associated with only one ingress port (QAR1 rule), one priority level in the previous hop (QAR2 rule), and one internal priority level (QAR3 rule). *QAR2* and *QAR3* rules are required to provide deterministic QoS, whereas *QAR1* isolates the flows from different nodes, avoiding the propagation of non-conformant traffic overloads. We refer the interested reader to [2], [9] for further details.

IV. PROBLEM STATEMENT

Let us consider a set of traffic classes $\mathcal{Q}$ in an asynchronous TSN network whose abstract model is described in the previous section. In this context, the flow allocation problem tackled in this work consists in finding the optimal configuration at every ATS $e \in \mathcal{E}$ for each traffic class $q \in \mathcal{Q}$ according to a given optimization goal. Specifically, the configuration of an ATS entails the decision of the traffic classes-to-shaped buffers and the traffic classes-to-priority level assignments. In this work, we consider the minimization of the global network's flow rejection probability as the optimization objective. The problem is subject to the following constraints:

- The E2E delay D_q of every traffic class q must be met.
- The traffic demand at every link $e \in \mathcal{E}$ must not exceed its capacity.
- Taking into account the QAR1, QAR2, and QAR3 rules of the ATS's interleaved shaping, the available number of shaped buffers at every ATS must not be exceeded.

V. FLOW ALLOCATION PROBLEM FORMULATION

This section details the problem formulation to find the optimal flow allocation in terms of flow acceptance ratio within an asynchronous TSN industrial network. Some transformations (e.g., convexification and linearization techniques) are applied to the original program to convert it into a convex mixed-integer non-linear program.

Let us consider the abstract asynchronous TSN network model described in Section III. We define the following decision variables:

- $M_q^{(\mathcal{P})}$ is the maximum number of flows in average with type of service q that can be allocated to the predefined path p that interconnects a given source with a given destination. Observe that $M_q^{(\mathcal{P})} \cdot r_q$ is the E2E capacity reserved for the traffic class q in the path p.
- $N_q^{(e)}$ denotes the average number of flows with traffic class q that can be simultaneously served by the ATS $e \in \mathcal{E}$. Again, please note that $N_q^{(e)} \cdot r_q$ is the capacity reserved for the traffic class q at the ATS e.
- $N_{q,p}^{(e)}$ denotes the average number of flows with traffic class q that can be simultaneously served by the ATS at priority level p.
- $U_{q,p}^{(e)}$ is a binary variable indicating whether the flows with 5QI q are assigned to the priority level p (=1) or not (=0).
- $D_p^{(e)}$ is the worst-case delay experienced by any flow allocated to the priority level p at the ATS $e \in \mathcal{E}$.

Formally, the flow allocation problem for the last hop can be formulated as follows:

$$minimize \left\{ \sum_{r\in\mathcal{P}} \sum_{q\in\mathcal{Q}} \lambda_q^{(r)} \cdot \alpha_q \cdot \left(\frac{\lambda_q^{(r)}}{M_q^{(r)} \cdot \mu_q + \lambda_q^{(r)}} \right) \right\} \quad (2a)$$

s.t :

$$\sum_{q\in Q} N_q^{(e)} \cdot r_q \leq C^{(e)} \quad \forall e \in \mathcal{E} \quad (2b)$$

$$0 \leq N_q^{(e)} \leq \sum_{p=1}^{P} N_{q,p}^{(e)} \quad \forall e \in \mathcal{E}, q \in \mathcal{Q} \quad (2c)$$

$$0 \leq N_{q,p}^{(e)} \leq N_{U,q}^{(e)} \quad \forall e \in \mathcal{E}, q \in \mathcal{Q}, p \in [1, P^{(e)}] \quad (2d)$$

$$N_{q,p}^{(e)} \cdot N_{q,k}^{(e)} \leq 0 \quad \forall e \in \mathcal{E}, q \in \mathcal{Q}, k \neq p \in [1, P^{(e)}] \quad (2e)$$

$$N_q^{(e)} \leq \sum_{r\in\mathcal{P}^{(e)}} M_q^{(r)} \quad \forall e \in \mathcal{E} \quad (2f)$$

$$\forall e \in \mathcal{E}, p \in [1, P^{(e)}], q \in \mathcal{Q} \quad \sum_{q\in\mathcal{Q}} \sum_{k=1}^{p} N_{q,k}^{(e)} \cdot b_q + l_{L,p} \leq$$
$$\left(D_p^{(e)} - l_p^{(e)} \right) \cdot \left(C^{(e)} - \sum_{q\in\mathcal{Q}} \sum_{k=1}^{p-1} N_{q,k}^{(e)} \cdot r_q \right) \quad (2g)$$

$$\forall \mathcal{P} \subseteq \mathcal{E}, q \in \mathcal{Q} \quad \sum_{e\in\mathcal{P}} \sum_{p=1}^{P^{(e)}} U_{q,p}^{(e)} \cdot D_p^{(e)} \leq D_q \quad (2h)$$

$$\sum_{i\in\mathcal{I}^{(e)}} \sum_{p=1}^{P^{(e)}} \sum_{q=1}^{|\mathcal{Q}|-1} \sum_{k=q+1}^{|\mathcal{Q}|} U_{q,p}^{(e)} \cdot U_{k,p}^{(e)} \sum_{m=1}^{P^{(i)}} U_{q,m}^{(i)} \left(1 - U_{k,m}^{(i)} \right)$$
$$+ \sum_{p=1}^{P^{(e)}} \bigvee_{q\in\mathcal{Q}} U_{q,p}^{(e)} \cdot I_q^{(e)} \leq S^{(e)} \quad \forall e \in \mathcal{E} \quad (2i)$$

$$\sum_{p=1}^{P} U_{q,p}^{(e)} = 1 \quad (2j)$$

1) Objective: The optimization goal (2a) aims to minimize the overall flow rejection probability in the network. Specifically, the factor $\lambda_q^{(r)} / \left(M_q^{(r)} \cdot \mu_q + \lambda_q^{(r)} \right)$ is an upper bound of the flow rejection probability for the M/G/c/c queuing system that models the capacity of each path $r \in \mathcal{P}$ to allocate flows with traffic type q [11], [12]. The parameters $\lambda_q^{(r)}$ and μ_q stand for the arrival rate to the path r and the inverse of the mean lifetime τ_q of flows belonging to the traffic class q, respectively. The upper bound of the flow rejection probability for each traffic class q and predefined path $r \in \mathcal{P}$ is weighted by the respective mean flow arrival rate λ_q^r in order to take into account the importance of the flow rejection probability of the path r and traffic class q in the global rejection probability of the network.

2) Constraints: Constraint (2b) ensures that the aggregated sustainable rate of the number of flows allocated in average to each link e does not exceed its capacity. Constraints (2c)-(2d) impose upper bounds for $N_{q,p}^{(e)}$ and $N_q^{(e)}$ decision variables. Constraints (2e) and (2j) enforce that each traffic class q is assigned to only one priority level p at each ATS e. Constraint (2f) establishes the relation among the variables $N_q^{(e)}$ and $M_q^{(r)}$ in order to make the problem formulation consistent. Constraints (2g) and (2h) assure that the E2E delay requisite is met for all of the traffic classes. Constraint (2i) warrants that the number of shaped buffers demanded in each ATS e is lower than or equal to the number of available shaped queues.

The formulation of the flow allocation problem in asynchronous TSN networks defined by (2a)-(2j) is a non-convex mixed-integer non-linear program. In order to solve it efficiently with standard convex optimization solvers (e.g., MOSEK), we can apply transformations to convert it into a convex mixed-integer non-linear program. Specifically, to that end, it is required to remove the products between decision variables included in constraints (2e), (2g), (2h), and (2i).

Constraint (2e) can be rewritten as a set of linear constraints using $U_{q,p}^{(e)}$ decision variables as follows:

$$0 \leq N_{q,p}^{(e)} \leq U_{q,p}^{(e)} \cdot N_{U,q}^{(e)} \quad \forall e \in \mathcal{E}, q \in \mathcal{Q}, p \in [1, P^{(e)}] \quad (3a)$$

$$0 \leq N_{q,k} \leq (1 - U_{q,p}) \cdot N_{U,q}^{(e)}$$
$$\forall e \in \mathcal{E}, q \in \mathcal{Q}, k, p \in [1, P^{(e)}] : k \neq p \quad (3b)$$

To remove the cross products of decision variables in (2g), we add the following decision variables:

$$z_p^{(e)} = C^{(e)} - \sum_{q\in\mathcal{Q}} \sum_{k=1}^{p-1} N_{q,k}^{(e)} \cdot r_q \quad \forall e \in \mathcal{E}, p \in [1, P^{(e)}] \quad (4)$$

Then, we might define the decision variables $w_p^{(e)}$ and consider McCormick envelopes for linearizing the cross products:

$$w_p^{(e)} \geq D^L \cdot z_p^{(e)} \quad \forall e \in \mathcal{E}, p \in [1, P^{(e)}] \quad (5)$$

$$w_p^{(e)} \geq D^U \cdot z_p^{(e)} + D_p^{(e)} \cdot C^{(e)} - D^U \cdot C^{(e)} \quad \forall e \in \mathcal{E}, p \in [1, P^{(e)}] \quad (6)$$

$$w_p^{(e)} \leq D^U \cdot z_p^{(e)} \quad \forall e \in \mathcal{E}, p \in [1, P^{(e)}] \quad (7)$$

$$w_p^{(e)} \geq D_p^{(e)} \cdot C^{(e)} + D^L \cdot z_p^{(e)} - D^L \cdot C^{(e)} \quad \forall e \in \mathcal{E}, p \in [1, P^{(e)}] \quad (8)$$

To linearize the constraint (2h), we define the auxiliary binary decision variables $y_{q,p}^{(e)} = U_{q,p}^{(e)} \cdot D_p^{(e)}$. Then, (2h) becomes:

$$\forall \mathcal{P} \subseteq \mathcal{E}, q \in \mathcal{Q} \quad \sum_{e\in\mathcal{P}} \sum_{p=1}^{P^{(e)}} y_{q,p}^{(e)} \leq D_q \quad (9)$$

And the following set of linear constraints have to be added:

$$D_p^{(e)} - \left(1 - U_{q,p}^{(e)} \right) \cdot D^U \leq y_{q,p}^{(e)} \leq D_p^{(e)} + \left(1 - U_{q,p}^{(e)} \right) \cdot D^{(U)}$$
$$\forall e \in \mathcal{E}, q \in \mathcal{Q}, p \in [1, P^{(e)}] \quad (10)$$

$$0 \leq y_{q,p}^{(e)} \leq U_{q,p}^{(e)} \cdot D^U \quad \forall q \in \mathcal{Q}, p \in [1, P^{(e)}] \quad (11)$$

TABLE I: Features of typical industrial services traffic flows.

Industrial service	Traffic pattern	Rate (Mbps)	b_q (Bytes)	$l_q^{(max)}$ (Bytes)	$D_q^{(max)}$ (ms)	τ_q (s)
MC	PD	1.55	324	324	1	28800
AR	AND	20	25000	1500	10	3600
PM	PD	0.05	186	186	20	28800
CLPC	PD	2	448	448	1	28800
HMI	AD	4	5000	1500	5	1800

Traffic patterns: Periodic Deterministic (PD), Aperiodic Non-Deterministic (AND), and Aperiodic Deterministic (AD).

Finally, to remove the cross products in (2i), we define the following binary variables: $\delta_{q,k,p}^{(e)} = U_{q,p}^{(e)} \cdot U_{k,p}^{(e)}$ $\forall e \in \mathcal{E}$, $q,k \in \mathcal{Q}$, $q \neq k$, $p \in [1, P^{(e)}]$, $\overline{\delta}_{q,k,p}^{(e)} = U_{q,p}^{(e)} \cdot (1 - U_{k,p}^{(e)})$ $\forall e \in \mathcal{E}$, $q,k \in \mathcal{Q}$, $q \neq k$, $p \in [1, P^{(e)}]$, and $\gamma_{q,k,p,m}^{e,i} = \delta_{q,k,p}^{(e)} \cdot \overline{\delta}_{q,k,m}^{(i)}$ $\forall e \in \mathcal{E}$, $i \in \mathcal{I}^{(e)}$, $q,k \in \mathcal{Q}$, $q \neq k$, $p \in [1, P^{(e)}]$, $m \in [1, P^{(i)}]$. Where $\mathcal{I}^{(e)}$ is the set of the ATSs whose traffic is aggregated in the ATS e. In other words, $\mathcal{I}^{(e)}$ is the set gathering all the previous hops for the ATS e. Then, we can add the following constraints to convexify (2i):

$$\sum_{i \in \mathcal{I}^{(e)}} \sum_{p=1}^{P^{(e)}} \sum_{q=1}^{|\mathcal{Q}|-1} \sum_{k=q+1}^{|\mathcal{Q}|} \sum_{m=1}^{P^{(i)}} \gamma_{q,k,p,m}^{(e,i)} + \sum_{p=1}^{P^{(e)}} \bigvee_{q \in \mathcal{Q}} U_{q,p}^{(e)} \cdot I_q^{(e)} \leq S^{(e)}$$
$$\forall e \in \mathcal{E} \tag{12}$$

$$U_{q,p}^{(e)} - \left(1 - U_{k,p}^{(e)}\right) \leq \delta_{q,k,p}^{(e)} \leq U_{q,p}^{(e)} + \left(1 - U_{k,p}^{(e)}\right),$$
$$\delta_{q,k,p}^{(e)} \leq U_{k,p}^{(e)} \quad \forall e \in \mathcal{E}, \ q,k \in \mathcal{Q}, \ q \neq k, \ p \in [1, P^{(e)}] \tag{13}$$

$$U_{q,p}^{(e)} - U_{k,p}^{(e)} \leq \overline{\delta}_{q,k,p}^{(e)} \leq U_{q,p}^{(e)} + U_{k,p}^{(e)},$$
$$\overline{\delta}_{q,k,p}^{(e)} \leq \left(1 - U_{k,p}^{(e)}\right) \quad \forall e \in \mathcal{E}, \ q,k \in \mathcal{Q}, \ q \neq k, \ p \in [1, P^{(e)}] \tag{14}$$

$$\delta_{q,k,p}^{(e)} - \left(1 - \overline{\delta}_{q,k,m}^{(i)}\right) \leq \gamma_{q,k,p,m}^{(e,i)} \leq \delta_{q,k,p}^{(e)} + \left(1 - \overline{\delta}_{q,k,m}^{(i)}\right),$$
$$\gamma_{q,k,p,m}^{(e,i)} \leq \overline{\delta}_{q,k,m}^{(i)} \quad \forall e \in \mathcal{E}, i \in \mathcal{I}^{(e)}, \ q,k \in \mathcal{Q}, \ q \neq k,$$
$$p \in [1, P^{(e)}], \ m \in [1, P^{(i)}] \tag{15}$$

The optimization problem for the flow allocation in ATS-based networks with the objective (2a) and constraints (2b), (2c), (2d), (3a), (3b), (2f), (4), (5), (6), (7), (8), (9), (10), (11), (12), (13), (14), (15), and (2j) is a convex mixed-integer non-linear program.

VI. RESULTS

A. Experimental Setup

With the focus on assessing the capacity of an asynchronous TSN network to accommodate industrial traffic flows, we carried out a numerical evaluation using Mosek to solve the flow allocation program derived in Section V. To that end, we considered three different network topologies, which are typical in industrial scenarios (e.g., star, daisy chain and ring, as shown in Fig. 2) [4]. The figure also includes the predefined paths considered to interconnect every S-D pair. The capacities and lengths for all the links were set to 1 Gbps and 200 m,

respectively. We consider a speed of light of $2 \cdot 10^8$ m/s in the wire. Five typical industrial services have been considered in our setup, namely, motion control (MC), augmented reality (AR), process monitoring (PM), closed-loop process control (CLPC), and human-machine interface (HMI) [13], [14]. Their characteristics are included in Table I and were mainly extracted from [13] and [14]. We supposed a similar flow arrival rate for all types of industrial services considered. Regarding the ATSs we supposed eight priority queues (priority levels) for all of them as that is the maximum number of priority levels specified in IEEE 802.1Q standards [15]. For the interleaving shaping stage, we assumed ten shaped buffers for every ATS.

B. Performance Evaluation

Figure 3 shows the flow rejection probability as a function of the flow arrival rate per S-D pair for every topology depicted in Fig. 2. We evaluated ten workload (i.e., flow arrival rates) points per S-D pair. These points correspond to a link utilization ranging from 20% to 190%. That is, each S-D pair generates a flow arrival rate such that if all of the incoming flows were accepted, the utilization of the link capacity would vary within the workload interval mentioned above. The results are consistent with the setup. As observed, the star topology presents the highest flow rejection probability due to all of the occupied links aggregate the traffic from two S-D pairs. Ring topology outperforms the daisy chain one because none of the predefined paths include more than two hops. As a consequence, the number of links that aggregate traffic from two S-D pairs is lower than in the daisy chain case.

Figure 4 depicts the experimental delays obtained per traffic type and topology considered. Specifically, given a service and a topology, the height of each bar and error bar stands for the mean and the maximum of the worst-case delays measured for the workload points studied, respectively. As it can be seen in the figure, the E2E delay requirement (refer to sixth column Table I) is fulfilled for every service and topology. Interestingly, the experimental maximum delays are close to the delay budgets. The reason behind lies on the fact that the optimization problem formulated in Section V minimizes the flow rejection probability. Consequently, it consumes the delay budget as much as possible, leveraging the available resources.

Figure 5 shows the utilization of the least used link that belongs to any predefined path. As observed, the topology and setup highly impact on the link utilization. This fact is due to the optimization problem takes into consideration the whole network status to minimize the number of flows rejected. As shown in Table II, the star topology performs the fairest allocation of the number of flows per service followed by the ring topology. Last, the daisy chain topology prioritizes the allocation of flows with the lowest sustainable rate, e.g., PM.

VII. CONCLUSIONS

In this article, we have assessed the maximum attainable utilization offered by asynchronous Time-Sensitive Networking (TSN). We have formally formulated the flow allocation problem for asynchronous TSN networks as a convex mixed-integer non-linear program. Finally, we have numerically evaluated the capacity of the asynchronous TSN networks to accommodate
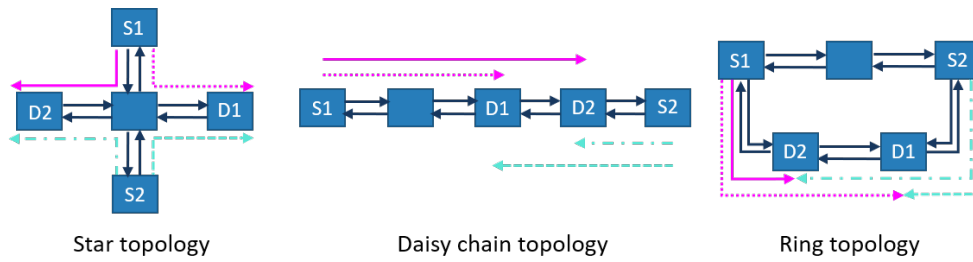
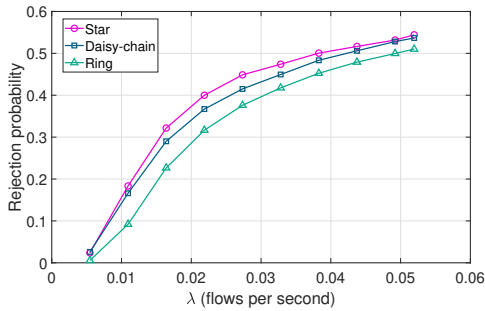Fig. 2: Industrial network topologies considered.

Fig. 3: Flow rejection probability for different industrial network topologies.
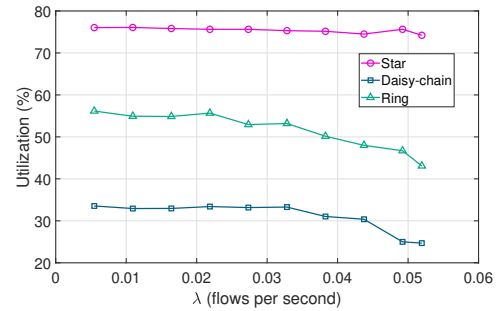

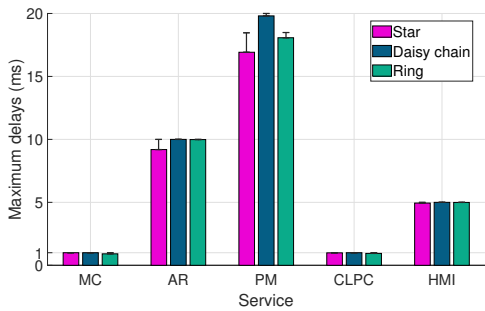Fig. 5: Link utilization for different industrial network topologies.

Fig. 4: Delays per service type and per topology.

streams in a realistic industrial scenario. We have considered three typical topologies for industrial networks, namely, star, daisy chain, and ring. The results suggest that the link utilization heavily depends on the network topology and the traffic matrix.

ACKNOWLEDGMENT

TABLE II: Number of allocated flows per service and per topology when every S-D pair generates a traffic load of 800 Mbps in average.

Topology \ Service	Star	Daisy Chain	Ring
MC	122	46	75
AR	6	4	7
PM	1904	1374	2489
CLPC	88	23	50
HMI	44	17	19

REFERENCES

[1] A. Nasrallah *et al.*, "Ultra-low latency (ull) networks: The ieee tsn and ietf detnet standards and related 5g ull research," *IEEE Communications Surveys Tutorials*, vol. 21, no. 1, pp. 88–145, Firstquarter 2019.

[2] J. Specht and S. Samii, "Urgency-based scheduler for time-sensitive switched ethernet networks," in *2016 28th Euromicro Conference on Real-Time Systems (ECRTS)*, July 2016, pp. 75–85.

[3] J.-Y. Le Boudec, "A theory of traffic regulators for deterministic networks with application to interleaved regulators," *IEEE/ACM Trans. Netw.*, vol. 26, no. 6, pp. 2721–2733, Dec. 2018.

[4] "Integration of industrial ethernet networks with 5g networks," White Paper, 5G ACIA, Nov. 2019.

[5] E. Mohammadpour, E. Stai, M. Mohiuddin, and J. Le Boudec, "Latency and backlog bounds in time-sensitive networking with credit based shapers and asynchronous traffic shaping," in *2018 30th International Teletraffic Congress (ITC 30)*, vol. 02, Sep. 2018, pp. 1–6.

[6] Z. Zhou, M. S. Berger, S. R. Ruepp, and Y. Yan, "Insight into the IEEE 802 .1 Qcr Asynchronous Traffic Shaping in Time Sensitive Network," *Adv. in Sci. Technol. and Eng. Syst. J.*, vol. 4, no. 1, pp. 292–301, 2019.

[7] 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. 44 165–44 181, 2019.

[8] A. Grigorjew, F. Metzger, T. Hoßfeld, and J. Specht, "A simulation of asynchronous traffic shapers in switched ethernet networks," in *2019 Int. Conf. on Networked Syst. (NetSys)*, March 2019, pp. 1–6.

[9] J. Specht and S. Samii, "Synthesis of queue and priority assignment for asynchronous traffic shaping in switched ethernet," in *2017 IEEE Real-Time Systems Symposium (RTSS)*, 2017, pp. 178–187.

[10] J. Prados-Garzon, T. Taleb, and M. Bagaa, "Learnet: Reinforcement learning based flow scheduling for asynchronous deterministic networks," in *ICC 2020 - 2020 IEEE Int. Conf. on Commun. (ICC)*, 2020, pp. 1–6.

[11] M. J. Sobel, "Simple inequalities for multiserver queues," *Management Science*, vol. 26, no. 9, pp. 951–956, 1980.

[12] A. Harel, "Sharp bounds and simple approximations for the erlang delay and loss formulas," *Manage. Sci.*, vol. 34, no. 8, pp. 959–972, 1988.

[13] "A 5g traffic model for industrial use cases," White Paper, 5G ACIA, Nov. 2019.

[14] 3GPP TS22.104 V17.4.0. (2020) Service Requirements for Cyber-Physical Control Applications in Vertical Domains.

[15] "Ieee standard for local and metropolitan area networks–bridges and bridged networks," *IEEE Std 802.1Q-2014 (Revision of IEEE Std 802.1Q-2011)*, pp. 1–1832, 2014.