Time-Sensitive Networking for 5G Fronthaul Networks

Sushmit Bhattacharjee*, Robert Schmidt†, Kostas Katsalis*, Chia-Yu Chang†, Thomas Bauschert‡, Navid Nikaein†
*Huawei Technologies Duesseldorf Gmbh, Munich, Germany.
†Eurecom, Sophia Antipolis, France.
‡Technische Universität Chemnitz, Germany

***Abstract*—In 5G radio access networks, meeting the performance requirements of the fronthaul network is quite challenging. Recent standardization and research activities are focusing on exploiting the IEEE Time Sensitive Networking (TSN) technology for fronthaul networks. In this work we evaluate the performance of Ethernet TSN networks based on IEEE 802.1Qbv and IEEE 802.1Qbu for carrying real fronthaul traffic and benchmark it against Ethernet with Strict priority and Round Robin scheduling. We demonstrate that both 802.1Qbv and 802.1Qbu can be well used to protect high-priority traffic flows even in overload conditions.**

***Index Terms*—5G, fronthaul networks, IEEE TSN, mobile network.**

I. Introduction

In order to meet the strict requirements imposed by a number of diverse use cases like Ultra Reliable and Low Latency Communications (URLLC), 5G mobile networks are becoming more and more complex. Software Defined Radio (SDR), Software Defined Networking (SDN) and Network Function Virtualization (NFV) concepts are applied in an integrated 5G architecture framework which allows the decoupling and chaining of network functions following a microservices architecture [1].

Regarding the transport layer of 5G networks, academia and industry endeavor to adopt full-fledged solutions built along two strategic axes. The first one is about applying packet switching solutions based on Ethernet and IP, while the second one is about deterministic communications. Indeed technologies like Ethernet over OTN, EoMPLS, and Ethernet over DWDM are well established, while technologies like Flex-E/X-Ethernet [2] or Flex-O intent on achieving not only greater flexibility and efficiency but also deterministic performance. Especially for the fronthaul network, different standardization bodies like 3GPP, IEEE and ITU agree on challenging requirements on high and guaranteed throughput as well as particular upper bounds on delay and jitter. Our focus in this paper is on the IEEE 802.1 Time Sensitive Networking (TSN) technology and we evaluate its ability to provide a guaranteed performance in terms of delay and jitter for the fronthaul network. The IEEE 802.1 TSN TG focuses on time synchronization issues and physical and link layer techniques to achieve a guaranteed delivery of data with bounded low latency, low delay variation and low loss. Time aware shaping (IEEE 802.1Qbv) and frame preemption (IEEE 802.1Qbu) are the two key mechanisms used to limit Ethernet frame propagation latency, while similar mechanisms regarding the IP layer are investigated by the Internet Engineering Task Force (IETF) Detnet Working Group. As the IEEE TSN amendments to 802.1Q have been recently published in IEEE 802.1Q-2018, performance evaluation studies so far only relied on simulations.

The authors in [3], study if Common Public Radio Interface (CPRI) over Ethernet can meet the stringent delay and jitter requirements of the proposed comb-fit scheduling. In [4], a time aware shaper based on the IEEE 802.1Qbv standard is proposed for an Ethernet-based fronthaul network. By applying simulations it is demonstrated that contention of high-priority traffic can be reduced and the frame delay jitter can be minimized. In [5], a different approach is taken by assuming extreme packet delay percentiles contrary to maximum one way end-to-end delays. This comes at an expense of a high frame loss ratio (FLR) but within the limits defined by evolved CPRI (eCPRI) specifications and the IEEE 802.1CM profile. Discrete-event simulations are carried out to confirm the results. In [6], the authors assess the usage of Ethernet with IEEE 802.1Qbu and IEEE 802.1Qbv enhancements for carrying fronthaul traffic (more specifically CPRI traffic). They demonstrate that scheduled Ethernet traffic can meet the CPRI jitter requirements. In comparison to the above works, we provide an in-depth study on the latest technologies for fronthaul networks. For that we consider real fronthaul traffic carried over a TSN prototype. To the best of our knowledge our study is the first one to analyze real fronthaul traffic carried over Ethernet TSN. We also benchmark TSN against the Strict Priority and plain Round Robin scheduling schemes.

The main contributions of this paper are as follows:

- Presentation of the current technology landscape for fronthaul networking.
- Description of the fundamentals of the IEEE TSN technology and the profile for fronthaul networks as standardized in IEEE 802.1CM.
- Discussion of implementation results for traffic aware scheduling and traffic preemption.
- Discussion of observations from scheduling exploiting 802.1Qbv.

In Section II, we outline the evolution of mobile networks towards 5G; and in section III, we describe the fundamentals of the TSN technology and the profile for fronthaul networks.

Then, in section IV, we provide performance evaluation results. In section V, we summarize our findings and outline future research directions.

II. Motivation

As the number of mobile users advances at an unprecedented speed and the users are demanding higher throughput and quality of experience (QoE), legacy LTE technology is not able to meet these requirements. Introduction of Cloud-RAN (C-RAN) might alleviate these problems, by decoupling the radio part and the baseband processing part. In C-RAN the Remote Radio Heads (RRHs) are responsible for the lower layer PHY functions (radio frequency (RF), signal amplification, D/A and A/D conversion), while the basenband processing and the higher layer protocols are performed in a centralized pool of Base Band Units (BBUs). The link between a RRH and a BBU is denoted as fronthaul. The C-RAN concept together with techniques like eICIC (enhanced Inter-Cell Interference Coordination) and CoMP (Coordinated Multipoint) is able to improve radio resource usage efficiency.

However, the need for bandwidth on the fronthaul link (due to the baseband signal transmission) makes the application of C-RAN for 5G mobile networks quite difficult. To lower the bandwidth requirements on the fronthaul, several functional split options between BBU and RRH have been recently proposed. The requirements for the fronthaul link greatly depend on the chosen functional split. Different standardization bodies (3GPP, NGFI or eCPRI) address these split options but unfortunately use a different terminology to denote them.

In Fig. 1 the functional split options 1 to 8 as proposed by 3GPP (see TS 38.401, TS 38.806, TR 38.816) and the eCPRI split options A to E [7] are depicted for the downlink case. Split options 1 to 4 (between MAC and RLC) are called Higher layer Splits (HLS), while the remaining ones are Lower Layer Splits (LLS). Note, that the functional splits for the downlink are not the same as for the uplink direction; for example split 7-3 is not supported for the uplink. In Table I, we summarize the terminologies adopted by 3GPP, CPRI and IEEE. Since this study focuses on the deterministic transmission aspects of the Ethernet-based fronthaul link, we suggest interested readers to refer to the works in [8], [9], [10], [11] for an extensive discussion and analysis of all functional splits and related terminologies.

A. The Fronthaul network for 5G

The midhaul describes the segment between DUs and CUs; the fronthaul the link between BU and RUs and the backhaul the segment between CUs and the core network (Next Generation Core (NGC)) (see also Fig.1). The term *Transport Network* comprises all segments. Table II provides an overview of the interfaces and protocols related to the fronthaul network.

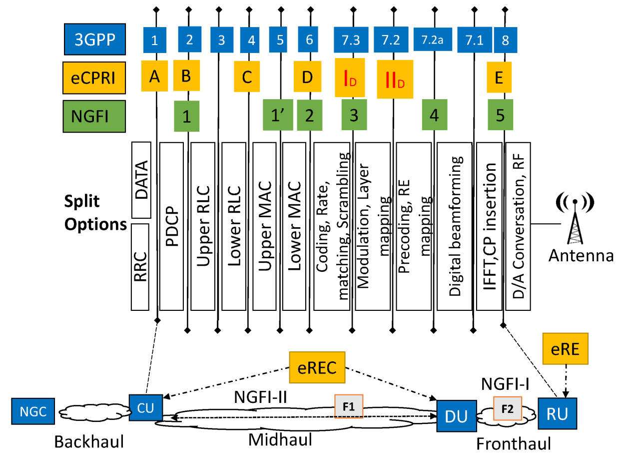


Fig. 1: Functional splits by 3GPP, eCPRI & NGFI (downlink).

TABLE I: Terminology Mappings

3GPP

- Low Layer Split (**LLS**): between radio and central RAN functions
- High Layer Split (**HLS**) between distributed and central RAN funct.
- Radio Unit (**RU**): all RAN functions placed below LLS interface
- Distributed Unit (**DU**): RAN functions between LLS and HLS
- Centralized Unit (**CU**): Contains all RAN functions above HLS and terminates inter-RAN interfaces. Aggregates several DUs.
- 3GPP splits CU function into one control plane function (CU-CP) and one or more user plane functions (CU-UP). *E1* is the interface between CU-UP and CU-CP. Together they form the gNB-CU.
- 3GPP RU plus DU as the evolution of eNodeB (gNB-DU).

CPRI Forum

- uses RE/eRE instead of RU
- uses REC/eREC to cover functionalities of both DU and CU

NGFI

BBU is redefined as Radio Cloud Center (RCC) and RRU becomes Radio Remote System (RRS). Aggregation Unit (RAU) which interfaces with RCC and carries transport for several RRU.

IEEE 1914

- uses RU,DU,CU but is not distinguishing between CU-UP,CU-CP.

TABLE II: 5G Fronthaul network interface and protocols

Interfaces

- **F1** is between gNB-DU and gNB-CU (NGFI-II used in IEEE 1914.1)
- **F2** is between DU and RU (NGFI-I used in IEEE 1914.1)

Protocols

- **CPRI** includes data sample transport, control and management plane, and synchronization mechanism over the point-to-point link. It focuses on layer 1 and 2 with programmable sample size and little overhead.
- **OBSAI** is designed to support several radio standards among different generations. Its RP3-01 interface is used to transport radio samples using fixed sample size with larger overhead for complex network topology.
- **ORI** is specified by ETSI being compliant with CPRI, while it further states applicable topologies and radio standard with extra data sample compression and layer 3 control and management.
- **eCPRI** offers more flexible data transmission than CPRI via packet-based transport network. It lies above the transport network layer and offers different functional split options. However, eCPRI does not include control and management and synchronization.
- **RoE** defines encapsulations of time/frequency radio samples into Ethernet frames in native mode. It also defines structure aware/agnostic modes allowing CPRI to be efficiently remapped/tunneled via RoE.

The performance requirements of the fronthaul network is expressed by the following metrics: (a) capacity, (b) latency, (c) bit error rate, and (d) synchronization accuracy. In general, the fronthaul capacity is ranging from 1 to 10 Gbps for the low-level splits and from 50 to 200 Mbps for the high-level splits. Moreover, the most stringent delay requirement of the fronthaul network occurs for split option 1 (low level split) due to the hybrid automated repeated request (HARQ) process (belonging to the low-level RAN functionality) which guarantees a transport block acknowledgement within a few milliseconds. Thus, the standardization organizations (3GPP, NGFI) recommend 100 to 250 μs as the maximum one-way fronthaul delay for the low-level RAN functional split options. In contrast, the one-way delay requirements for high-level RAN split options are between 2 to 30 ms. Furthermore, the bit error rate shall be kept at a very low level. The specification of the allowed bit error rate and one-way frame loss rate according to CPRI/eCPRI can be found in [7]. Finally, a certain synchronization accuracy of the fronthaul network is required to properly adjust the timing and frequency between transmitter and receiver, e.g., CPRI states $\pm$2ppb frequency accuracy and less than 20 ns for timing error respectively.

III. Time Sensitive Networks for the Fronthaul

In the wireless domain, 3GPP and NGMN considered time-sensitive services under the umbrella of URLLC use cases. For IP networks, the IETF DETNET working group investigates new mechanisms to provide deterministic Quality of Service (QoS), spanning from explicit routes, packet replication and elimination to congestion protection with end-to-end synchronization. For the data link layer, the IEEE 802.1 Audio Video Bridging (AVB) task group focuses on enabling isochronous and deterministic low-latency services over legacy Ethernet. However, this was intended for multimedia streaming applications. In order to widen the area of applications, the IEEE 802.1 TSN Task Group (TG) was founded. The IEEE 802.1 TSN TG focuses mainly on physical and link layer techniques to achieve guaranteed delivery of data with bounded low latency, low delay variation and low loss. A comprehensive survey covering both fixed and wireless ultra low latency communication is presented in [12]. An overview of Ethernet and its evolution to various fields of application is provided in [13].

A. *IEEE Time Sensitive Networking*

A categorization of the relevant IEEE TSN standards is provided in Table III. Resource management aspects of TSN are covered by amendments like 802.1Qcc, 802.1Qdd. TSN synchronization is covered in IEEE 802.1AS, ongoing work is addressed in 802.1AS-Rev. The delay guarantees are supported by techniques like Scheduled Traffic (IEEE 802.1Qbv) and Frame Preemption (IEEE 802.3br, IEEE 802.1Qbu). These standards define how frames belonging to a particular traffic class or having a particular priority are handled by TSN-enabled bridges.

IEEE 802.1Qbv introduces a transmission gate operation for each queue as depicted in Fig. 2. The transmission gates *open/close* according to a known time schedule. The transmissions are controlled by a Gate Control List (GCL) which consists of multiple schedule entries. For instance, in the scenario depicted in Fig. 2, the GCL entry for T2 indicates that the gates for the queues 1 and 7 are open (1), while all other gates are closed (0). Based on these schedules, selected

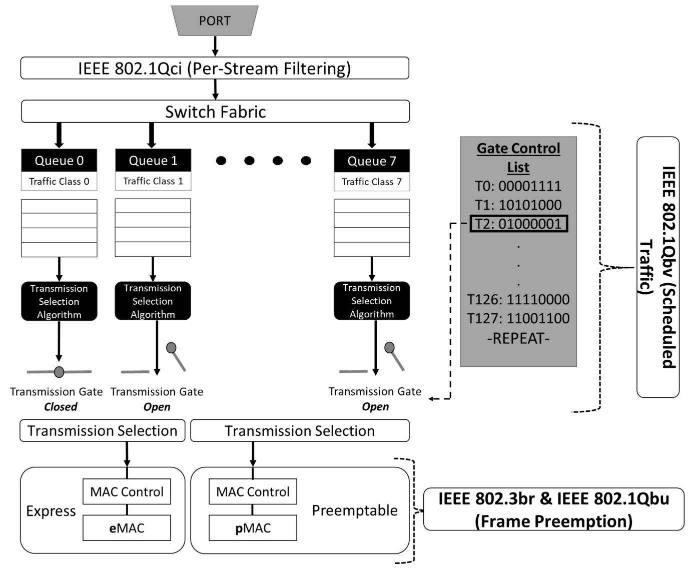


Fig. 2: IEEE TSN Scheduled Traffic & Frame Preemption.

traffic types can be allowed to pass through to the transmission selection block which provides access to the medium. Frame Preemption on the other hand, allows the ongoing transmission of a lower priority frame to be preempted by a higher priority frame and thus ensures lower latency for high priority frames. It maps frames onto two separate MAC service interfaces namely express MAC (eMAC) and preemptable MAC (pMAC) as seen in Fig. 2. Express frames can preempt preemptable frames by either interrupting the frame transmission or by preventing the start of a pMAC frame transmission. In case of an interruption, the pMAC frame transmission resumes after the transmission of the express frame has completed. A live demo presenting preemption for low-Latency mobile X-haul 100G Ethernet was presented by ADVA in ECOC 2018.

TABLE III: IEEE TSN Standards Overview

Category	Standards
Time Synchronization Providing network wide precise synchronization of the clocks of all entities at Layer 2.	IEEE 802.1AS & IEEE 802.1AS-Rev (Network Timing & Synchronization)
Latency & Jitter Separating traffic into traffic classes and efficiently forwarding & queuing the frames in accordance to these traffic classes.	IEEE 802.1Qav (Credit Based Shaping) IEEE 802.1Qbv (Scheduled Traffic) IEEE 802.3br & IEEE 802.1Qbu (Frame Preemption) IEEE 802.1Qch (Cyclic Queuing) IEEE 802.1Qcr (Asynchronous Traffic Shaping)
Reliability & Redundancy Maintaining network wide integrity by ensuring path redundancy and ingress queue policing.	IEEE 802.1CB (Frame Replication & Elimination) IEEE 802.1Qca (Path Control & Reservation) IEEE 802.1Qci (Per-Stream Filtering)
Resource Management Providing dynamic discovery, configuration and monitoring of network in addition to resource allocation & registration.	IEEE 802.1Qat & IEEE 802.1Qcc (Stream Reservation) IEEE 802.1Qcp (YANG Models) IEEE 802.1CS (Link-Local Reservation)

Regarding theoretical frameworks for TSN scheduling, in [14], methods to compute static schedules via Satisfiability Modulo Theories (SMT) are presented which guarantees bounded latency and delay variation for real-time traffic. The authors in [15], formulate a joint scheduling and routing problem with respect to time-trigerred (TT) and AVB traffic. They propose an ILP formulation to ensure that all frames are schedulable and the worst case end-to-end frame delay is minimized. In [16], the authors consider that the GCL cannot be of unlimited size and hence formulate constraints for a window-based IEEE 802.1Qbv GCL scheduling. The performance evaluation results of this scheduling scheme is outlined in [17] showing the trade-off between the computation time and the maximum number of windows per queue. In [18], network calculus is used to calculate the worst case latency bounds of AVB traffic in TSN-based networks. In [19], a set of formulas are provided for computing per traffic class bounds on end-to-end delay and backlog considering credit based shaping (CBS) and asynchronous traffic shaping (ATS).

B. 802.1CM Profile for the Fronthaul

Based on the area of application, *TSN Profiles* have been specified to explain which standards, protocols, features and options should be applied for a given use-case. The existing TSN *Profiles* are 802.1BA for AVB networks, IEC/IEEE 60802 TSN Profile for industrial automation, P802.1DG for automotive in-vehicle Ethernet communications and IEEE 802.1CM TSN for mobile fronthaul networks.

IEEE 802.1CM resulted from a collaborative effort of CPRI and IEEE 802.1. It describes how to meet the stringent fronthaul requirements in an Ethernet-based bridged network which can support not only fronthaul traffic but also other concurrent traffic types. In 802.1CM both CPRI and eCPRI splits are supported (Class 1 and Class 2 respectively). In both cases the following types of data are considered: a) User Data; b) Control and Management Data and c) Synchronization Data. The relevant requirements (for these types of data) are defined by the CPRI Specification V7.0 and by the eCPRI Transport Network Specification V1.1 respectively. For example, for class 2 (eCPRI), the maximum end-to-end one-way latency is 100us for high priority user plane data traffic between eREC and eRE. The maximum tolerable Frame Loss probability for control plane data is 10^{-6}, and the internal time error requirements for eRE/RE synchronization varies between 15 to 30 ns, depending on the case and category.

Moreover, 802.1CM mentions the components that contribute to the worst-case latency for a single hop from a bridge to a bridge: *Input queuing delay + Interference delay (Queuing delay + Self-queuing delay) + Frame transmission delay + LAN propagation delay + Store-and-forward delay*.

Further, P801.CMde is now investigating several enhancements to the fronthaul profiles. Three types of fronthaul data flows are defined:

- High Priority Fronthaul (HPF) data, which has 100 us maximum end-to-end one-way latency (Class 1 IQ data and Class 2 user plane data);
- Medium Priority Fronthaul (MPF) data, which has 1 ms maximum end-to-end one-way latency, (Class 2 user plane data and Class 2 control plane data);
- Low Priority Fronthaul (LPF) data, which has 100 ms maximum end-to-end one-way latency (control data for both class 1 and Class 2).

Furthermore, two profiles are specified being applicable to class 1 (CPRI) and class 2 (eCPRI):

- Profile A: exploits strict priority queuing to perfrom service differentiation between high priority traffic (User data or/and IQ data) and low priority traffic (control and management data). The maximum frame size for all traffic flows is 2000 octets (MAC Protocol Data Unit (PDU)) on each port.
- Profile B: exploits frame preemption (802.3br and 802.1Qbu) together with strict priority scheduling in order to differentiate between high priority (fronthaul traffic) and low priority preemptable traffic (non-fronthaul traffic). The maximum frame size for fronthaul traffic is still 2000 octets (MAC PDU) on each port, while the maximum frame size for non-fronthaul traffic might vary.

802.1CM also discusses how the time synchronization requirements can be met for precision time protocol (PTP) enabled devices satisfying for example the ITU-T G.8275.1 telecom profile and ITU-T G.8272, ITU-T G.8273 depending on the deployment case.

IV. Performance Evaluation

The target of our performance evaluation is to investigate the performance of a TSN-enabled network carrying fronthaul traffic. The performance metrics we are focusing on are the average forwarding latency and average delay-variation (jitter) of high priority traffic that competes with lower priority traffic and background traffic. The forwarding latency is measured as the time interval between the start of sending the *k-th* packet from the RRU and the end of receiving the *k-th* packet at the BBU.

A. Scenario Description

Since eCPRI implementations are still not available, we focus on NGFI split 4.5 in our experiments which is similar to 3GPP option 7-1 (see Fig.1). For this split option a custom defined format for packetization is defined by OpenAirInterface (OAI) and implemented in OAI v2019.w30 [20]. Such a functional split requires placing extra DFT/IDFT software modules collocated with RRHs and can decrease the fronthaul capacity requirement roughly by a factor of 2 [21]. The employed split transports I/Q samples in the frequency domain, i.e. after removal of the cyclic prefix and FFT, and before resource element (de-)mapping, while beamforming is not employed.

Like in NGFI split 5, the capacity of split F4.5 depends only on the number of radio carriers, and thus there is no need for standard-specific processing, e.g., radio resource (de-)mapping, at the cell site. It is thus part of the cell-related fronthaul processing (as opposed to user-specific processing), exhibiting a constant bandwidth requirement since the I/Q samples related to the full cell capacity need to be transmitted, irrespective of the cell load [21]. Since the fronthaul capacity scales with the number of cell sectors and antenna elements and not with the traffic load, we do not have to consider the traffic load. Regarding the packetization of the samples, we refer to [22] for more details.

To reduce the fronthaul bandwidth, the samples are compressed using 8-bit A-law compression (which was originally defined in ITU-T G.711 standard for audio encoding). The packets are encapsulated and transported over UDP/IP, annotated with 802.1Q VLAN tags to mark different flows and QoS

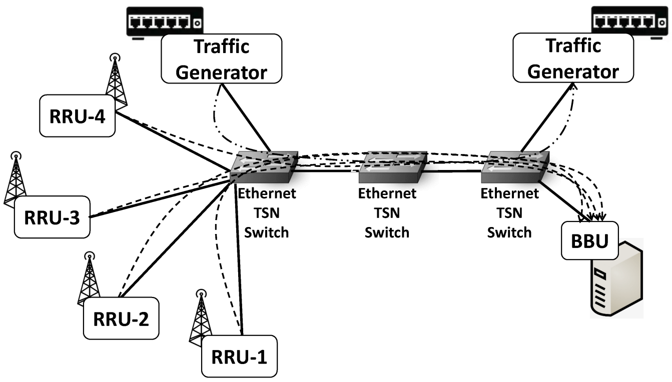


Fig. 3: Considered network topology

TABLE IV: Evaluation Scenarios

Figures	Traffic Types	Policies
4(a): latency/cycle time	Non-Scheduled Priority 7, Non-Scheduled Priority 0	**SP, SRR, TSN-QBV**
4(b): latency/packet size **4(c):** jitter/packet size	Scheduled Priority 7, Scheduled Priority 0, Non-Scheduled Background Traffic (best-effort)	**SP, TSN-QBV**
5(a): latency/background load **5(b):** jitter/background load	Fronthaul Traffic, Non-Scheduled Background Traffic (best-effort)	**SP, TSN-QBV, Preemption**
5(c): jitter-latency/packet size	Fronthaul Traffic, Non-Scheduled Background Traffic (best-effort)	**SP, TSN-QBV**
6(a): jitter-latency/number of RRUs (3 hops) **6(b):** jitter-latency/number of RRUs (4 hops)	Fronthaul Traffic, Non-Scheduled Background Traffic (best-effort)	**SP, TSN-QBV, Preemption**

requirements. In our experimental setup for TSN network, we use prototype TSN switches equipped with eight 1 Gigabit ports each. We use a traffic generator to emulate background traffic. In order to generate IEEE 802.1Qbv scheduled traffic we use Ixia traffic generator from Keysight Technologies, applying IEEE 802.1AS with hardware time-stamping for verifying time-synchronization. In some cases we also use the *iperf* tool to generate background traffic.

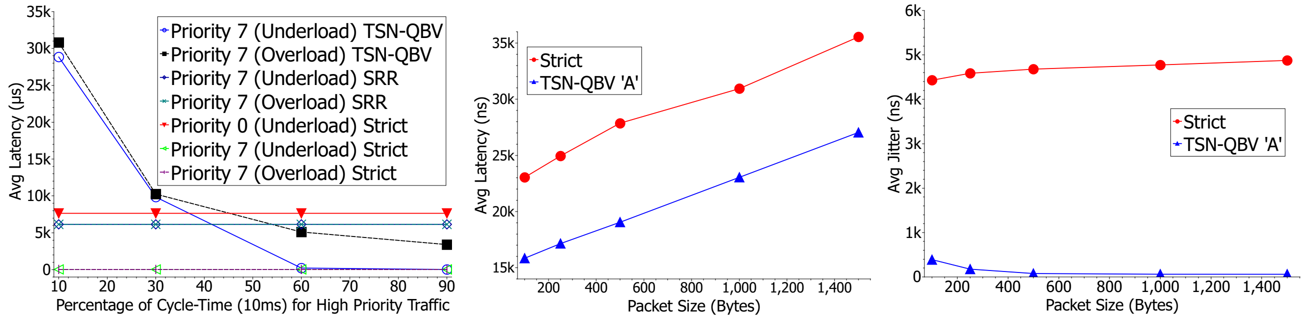
Our performance evaluation is divided into two parts. In the first part (Fig. 4(a) to 4(c)) we compare the performance of TSN 802.1Qbv against the Strict Priority (SP) and Simple Round Robin (SRR) schemes for carrying scheduled and non-scheduled traffic. The results provide a glimpse of the performance of TSN 802.1Qbv in comparison to other scheduling mechanisms. Scheduled traffic is generated by TSN-enabled endpoints in accordance to the TSN-802.1Qbv schedules deployed on the switches. The non-scheduled traffic is generated by endpoints that are not TSN-enabled and is continuous. The network scenario comprises only TSN Ethernet switches, the Ixia traffic generator and the iperf tool. In the second part (Fig. 5(a) to 6(b)), we carry real fronthaul traffic over Ethernet with TSN. The network scenario additionally comprises several RRU(s) and one BBU, see Fig. 3. An overview of all evaluation scenarios is shown in Table IV.

B. Evaluation Results

Fig. 4(a) refers to the evaluation of IEEE 802.1Qbv. Two traffic flows are generated (both continuous and non-scheduled), one with high priority (VLAN priority 7) and one with low priority (VLAN priority 0). We consider two scenarios namely underload and overload. In the underload scenario, both traffic flows have a data rate of 592 Mbps which is approximately 60% of the line-rate (1 Gbps). In the overload scenario, both traffic flows have a data rate of 987 Mbps (approximately 100% of the line-rate). In both scenarios and for all traffic flows, the packet length is set to 1500 Bytes. As mentioned in Section III, with IEEE 802.1Qbv, one can assign time schedules for the opening/closing of the transmission gates. In our experiments we consider a cycle-time of 10 ms and vary the share of the cycle time for which the transmission gates for high-priority traffic remains open. After the cycle-time (10ms) has passed the next cycle starts and the schedules are repeated again, see Fig. 2. In case of SP scheduling, the average latency for the high-priority traffic is the lowest for both the underload and overload scenario with a value of 18.2 us (reason: high-priority traffic has always higher priority than low-priority traffic). The SRR mechanism gives an equal opportunity to both traffic flows irrespective of their priority. We notice that for SRR the average latency values for both the lower and higher priority traffic is the same being close to 6k us.

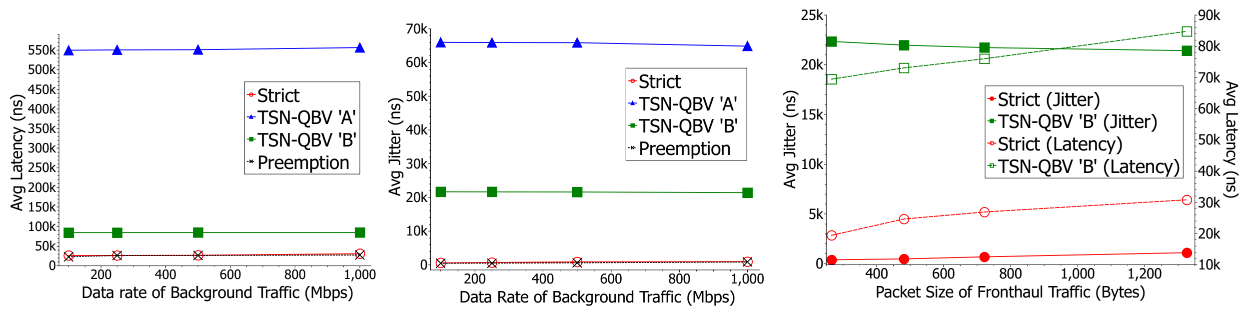
Fig. 4(a) shows that for IEEE 802.1Qbv, the average latency for high-priority traffic decreases as the duration of opening the transmission gates increases. It is as low as that of SP when the transmission gate for high-priority traffic remains open for almost the entire duration of the cycle time. The average latency increases when the transmission gate is open for a lower duration. This is expected because in this scenario, when the high-priotity packet arrives and finds the transmission gate closed, it needs to wait till the gate re-opens in the next cycle. This experiment shows the tunability feature of 802.1Qbv being able to assign specific duration for which a particular traffic flow can be scheduled.

Figures 4(b) and 4(c) show the average latency and jitter for SP and 802.1Qbv for scheduled traffic (i.e. the sender and the switch are using synchronized transmission) in presence of continuous background traffic depending on the packet size. For TSN (*TSN-QBV 'A'*), we fix the schedule such that the high-priority traffic is allowed to pass for 70% of the cycle-time (10 ms) and the low-priority traffic passes for the remaining cycle-time. We ensure that the background traffic is of low-priority i.e. best-effort traffic (VLAN priority 0) and is sent continuously at line-rate (986Mbps) (100% of line-rate). The packet sizes are varied from a minimum of 100 Bytes to a maximum of 1500 Bytes. Since the traffic is scheduled and the packets of the respective priority arrive when the gates for the corresponding priority class are open, TSN has the lowest jitter, the average minimum jitter being 60 ns even in presence of background traffic. The background traffic does not affect the high-priority traffic. This is because the TSN schedule prevents the interference of background traffic when the transmission gates for high-priority traffic are open. For SP, the latency and jitter values are always higher than for TSN. In case of SRR even higher values are observed. Since the latency values are very high, we omit this curve for scaling purposes. The minimum average latency for SRR is 2ms. Thus,



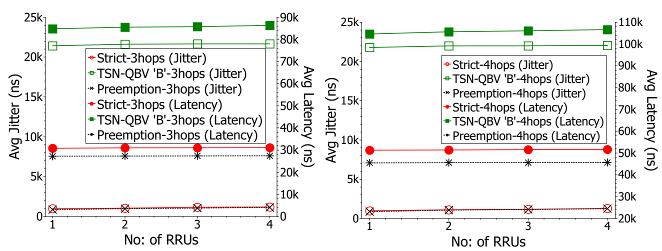
(a) (b) (c)

Fig. 4: Evaluation Results: TSN 8021Qbv compared with Strict Priority and Simple Round Robin (SRR)



(a) (b) (c)

Fig. 5: Evaluation results for fronthaul traffic over (a), (b) varying background traffic and (c) fronthaul packet sizes.



(a) (b)

Fig. 6: Jitter and latency for different number of RRUs and hops. Fig.6(a) depicts for 3 hops and Fig.6(b) for 4 hops.

when the TSN schedule on the switch can be set in accordance to the traffic flow, very low average jitter and latency values can be achieved and the traffic can be well protected even in presence of background traffic.

For the next set of experiments we evaluate a more realistic scenario, where we generate real fronthaul traffic between the RRU and BBU over Ethernet with TSN-802.1Qbv, Frame Preemption (TSN-802.1Qbu) and SP. We also applied the SRR scheme but as the performance values were much higher we omit the results in this paper. Figures 5(a) and 5(b) show the impact of background traffic on the fronthaul traffic performance. Here we consider only one RRU and one BBU which are connected over 3 hops i.e. with two Ethernet switches in between. We vary the data rate of the background traffic from a minimum of 100 Mbps to a maximum of 1Gbps. The packet size for the background traffic is kept constant at 100 Bytes. The fronthaul traffic sent by the RRU has a packet size of 1322 Bytes and is transmitted with an average data rate of 95 Mbps. For Ethernet with TSN-Qbv we use two different schedules. One is the same as the previous experiment namely *TSN-QBV 'A'* and the other one is named *TSN-QBV 'B'* where the high-priority traffic is allowed to pass for 90% of the cycle-time (10 ms) and the low-priority traffic passes in the remaining cycle-time. We also evaluate the performance of 802.1Qbu (Frame Preemption) in this experiment. Note that Frame Preemption is used independently and without the presence of 802.1Qbv. We observe that both in case of SP and Frame Preemption, the difference between the maximum and the minimum average latency is around 375 ns. In case of TSN-Qbv this difference is only 133 ns. This is because the background traffic is completely blocked during the time when the high-priority traffic is allowed to pass. However, the lowest average latency and jitter values are observed in case of Frame Preemption. The time taken to preempt a packet depends on the size of the packet being preempted which in our case is 100 Bytes (background traffic). The minimum average latency in case of Frame Preemption is 23us while the minimum average latency in case of SP is 26us. In case of TSN-Qbv, lower values can be noticed for TSN-QBV 'B' compared to TSN-QBV 'A' which is consistent with the results depicted in Fig. 4(a). The higher latency and jitter values of TSN-Qbv compared to SP and

Frame Preemption is expected because the fronthaul traffic is not scheduled and the packets are transmitted by the switch only according to the time-schedule and not immediately upon arrival. However, this is acceptable because it still falls within the acceptable range mentioned by IEEE 802.1CM for High Priority fronthaul data (100 us).

Fig 5(c) shows the impact of the packet size of the fronthaul traffic on the average latency and jitter. For this experiment, we use the same network scenario as in the previous experiment. However, the background traffic is kept constant at a data rate of 1 Gbps. The packet size of the fronthaul traffic is varied based on the Resource Blocks assigned. We assume fronthaul traffic with 6, 15, 25 and 50 resource blocks which correspond to packet sizes of 266 Bytes, 482 Bytes, 722 Bytes and 1326 Bytes. We observe that the average latency increases with increasing packet size for both TSN-Qbv and SP. However, the average jitter for TSN-Qbv decreases with increasing packet size - this is consistent with the results seen in Fig. 4(c).

Figures 6(a) and 6(b) show the impact of increasing the number of RRUs associated to one single BBU (and thus the impact of multiplexing multiple traffic flows from multiple RRUs) on the average latency and jitter. The results in Fig 6(a) correspond to the network scenario of Fig. 3 with only two switches between the RRUs and the BBU i.e. 3 hops. In Fig. 6(b), the number of hops is increased to 4 to observe the impact on the average latency and jitter. In both the cases, the fronthaul packets are 1322 Bytes long and the average data rate is 95 Mbps. We notice that in both cases, the average latency and average jitter remain more or less constant even as the number of RRUs are increased. Also, the average jitter value remains constant as the number of hops increases while the average latency increases. Note that in both experiments, the fronthaul traffic from all the RRUs is given the same high-priority. Using TSN-Qbv, one can also assign different priorities to the different traffic flows from the different RRUs. By doing so, one can prioritize specific traffic flows and accordingly assign schedules for resource sharing and protection of all traffic flows. However, with SP, this might not be the case. If different priorities are assigned to different traffic flows originating from different RRUs, the SP mechanism would always prioritize only the highest priority traffic and all other traffic flows would observe a significant increase in the latency and jitter. In case of Frame Preemption, the lowest average latency and jitter values are observed.

V. Conclusions and Future Work

In this work we evaluate the performance of IEEE TSN 802.1Qbv and IEEE 802.1Qbu for real fronthaul traffic and benchmark these techniques against Strict Priority and Round Robin schemes. We demonstrate that both techniques can be well used for protection of high-priority traffic flows even in overload conditions. In our future work we plan to address the following issues: end-to-end synchronization problems, optimal assignment of priorities for different types of traffic (e.g. user plane traffic, eCPRI messaging) over TSN, investigation of TSN performance under different split options and cross comparison with existing solutions.

Acknowledgments

This work has received funding from the EU H2020 5G-Picture project under grant agreement no 762057.

References

[1] I. Nadareishvili *et al.*, *Microservice architecture: aligning principles, practices, and culture*. O'Reilly Media, Inc., 2016.

[2] K. Katsalis, L. Gatzikis, and K. Samdanis, "Towards Slicing for Transport Networks: The Case of Flex-Ethernet in 5G," in *IEEE CSCN*, 2018.

[3] D. Chitimalla *et al.*, "5g fronthaul-latency and jitter studies of cpri over ethernet," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 9, no. 2, pp. 172–182, 2017.

[4] M. K. Al-Hares, P. Assimakopoulos, D. Muench, and N. J. Gomes, "Modeling time aware shaping in an ethernet fronthaul," in *IEEE GLOBECOM*. IEEE, 2017, pp. 1–6.

[5] G. O. Pérez, D. L. López, and J. A. Hernández, "5g new radio fronthaul network design for ecpri-ieee 802.1 cm and extreme latency percentiles," *IEEE Access*, vol. 7, pp. 82 218–82 230, 2019.

[6] T. Wan and P. Ashwood-Smith, "A performance study of cpri over ethernet with ieee 802.1 qbu and 802.1 qbv enhancements," in *IEEE GLOBECOM)*. IEEE, 2015, pp. 1–6.

[7] I. Specification, "Common public radio interface: ecpri interface specification," *version*, vol. 1.1, p. 62, 2018.

[8] L. M. P. Larsen *et al.*, "A survey of the functional splits proposed for 5g mobile crosshaul networks," *IEEE Communications Surveys Tutorials*, vol. 21, no. 1, pp. 146–172, Firstquarter 2019.

[9] N. Alliance, "Ngmn overview on 5g ran functional decomposition," *Next Generation Mobile Networks Ltd., Feb*, vol. 24, 2018.

[10] Ojaghi *et al.*, "Sliced-ran: Joint slicing and functional split in future 5g radio access networks," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 2019, pp. 1–6.

[11] Kalfas *et al.*, "Next generation fiber-wireless fronthaul for 5g mmwave networks," *IEEE Communications Magazine*, vol. 57, no. 3, pp. 138–144, 2019.

[12] A. Nasrallah *et al.*, "Ultra-low latency (ull) networks: A comprehensive survey covering the ieee tsn standard and related ull research," 03 2018.

[13] J. Sommer *et al.*, "Ethernet–a survey on its fields of application," *IEEE Communications Surveys & Tutorials*, vol. 12, no. 2, pp. 263–284, 2010.

[14] S. S. Craciunas, R. S. Oliver, M. Chmelík, and W. Steiner, "Scheduling real-time communication in ieee 802.1 qbv time sensitive networks," in *International Conference on Real-Time Networks and Systems*. ACM, 2016, pp. 183–192.

[15] P. Pop, M. L. Raagaard, S. S. Craciunas, and W. Steiner, "Design optimisation of cyber-physical distributed systems using ieee time-sensitive networks," *IET Cyber-Physical Systems: Theory & Applications*, vol. 1, no. 1, pp. 86–94, 2016.

[16] S. S. Craciunas *et al.*, "Formal scheduling constraints for time-sensitive networks," *arXiv preprint arXiv:1712.02246*, 2017.

[17] W. Steiner, S. S. Craciunas, and R. S. Oliver, "Traffic planning for time-sensitive communication," *IEEE Communications Standards Magazine*, vol. 2, no. 2, pp. 42–47, 2018.

[18] L. Zhao, P. Pop, Z. Zheng, and Q. Li, "Timing analysis of avb traffic in tsn networks using network calculus," in *Real-Time and Embedded Technology and Applications Symposium*. IEEE, 2018, pp. 25–36.

[19] E. Mohammadpour, E. Stai, M. Mohiuddin, and J.-Y. 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. 2. IEEE, 2018, pp. 1–6.

[20] N. Nikaein *et al.*, "Openairinterface: A flexible platform for 5g research," *ACM SIGCOMM*, vol. 44, no. 5, pp. 33–38, 2014.

[21] C. Chia-Yu *et al.*, "Impact of packetization and functional split on c-ran fronthaul performance," in *Proc. of IEEE ICC*, 2016.

[22] C. Chang *et al.*, "Flexcran: A flexible functional split framework over ethernet fronthaul in cloud-ran," in *2017 IEEE International Conference on Communications (ICC)*, May 2017, pp. 1–7.