

Original Research Article

End-to-end deterministic performance evaluation of integrated 5G and time-sensitive networking systems for avionics communications

Xiaobin Qi, Pengtao Su, Long Wan

AVIC Xi'an Aeronautics Computing Technique Research Institute, Xi'an, Shaanxi, 710127, China

Abstract: Avionics networks need deterministic Ethernet, which is exactly the gap Time-Sensitive Networking (TSN) is meant to fill. In next-generation aircraft, wired TSN and wireless 5G will run side by side; how well they interwork decides whether end-to-end latency can stay bounded. We built a hybrid testbed in which scheduled TSN traffic, set up per IEEE 802.1Qbv, is carried across the 5G air interface. Commercial off-the-shelf TSN and 5G kit from Xi'an Xiangteng and ZTE—Near product grade—Make up the platform. Our measurements point to the same conclusion from both sides: The 802.1Qbv schedule and the 5G configuration together govern end-to-end determinism, and neither can be ignored.

Keywords: IEEE 802.1Qbv; TSN; 5G

1. Introduction

Backed by the IEEE 802.1 Working Group, TSN tightens what standard Ethernet can promise: Bounded, low-latency delivery with negligible loss, and the ability to carry time-critical and best-effort traffic on the same wire. On board aircraft, legacy point-to-point buses—ARINC 429, MIL-STD-1553—Are being pushed aside in favor of a switched, Ethernet-based fabric, and TSN is the glue that makes that feasible. Industry forecasts put worldwide TSN revenue near USD 1.7 billion by 2028, with demand spilling out of aerospace into automotive and factory floors.

Wired does not mean wireless disappears. On aircraft, 5G is already taking over an increasing share of communications, and its Ultra-Reliable Low-Latency Communication (uRLLC) mode is precisely the kind of bearer that time-sensitive flows have been waiting for.

The problem is that a deterministic promise on the wired side is only as good as the last wireless hop. 3GPP has defined how 5G should interwork with TSN—5G acts as a black-box tunnel, absorbing service-quality demands inside its own stack—But looking TSN-like from the outside is not the same as being TSN-like on the inside. Resource allocation for scheduled TSN flows over the air remains an open question.

2. IEEE 802.1Qbv standard

Four families of standards make up the TSN toolkit: Synchronization (802.1AS), reliability via frame replication and elimination (802.1CB), configuration and management (802.1Qcc), and—Most relevant here—Bounded latency through scheduled traffic (802.1Qbv).

At the heart of 802.1Qbv is the Time-Aware Shaper (TAS). Rather than letting traffic queues fight for bandwidth, TAS opens and closes transmission gates on a global clock, so priority classes are separated in time rather than in priority alone. Flight-control frames and sensor data leave during their own windows; best-effort traffic waits outside. The Gate Control List (GCL) encodes, nanosecond by nanosecond, which queue is open. Guard bands wrap each scheduled window so a late frame cannot overrun the next slot. The price is high: Every node must be tightly synchronized, and a central controller has to compute gate times from the known propagation delays of each link. Looser, asynchronous variants exist, but they trade away some of that precision.

On our bench, TSN switches and gateways from Xi'an Xiangteng Microelectronics—802.1Qbv and TAS implemented—Sit next to a near-commercial 5G network from ZTE. Scheduled TSN frames are pumped over the 5G uRLLC radio, and the two systems are joined through the 3GPP black-box bridge.

3. Testbed setup

At the TSN edge sits the XT-TSN-GW gateway (**Figure 1**), also from Xiangteng. Its job is to let a garden-variety Ethernet device talk TSN without any TSN firmware on the host. Inside is Xiangteng's programmable REM three-port switch chip, which handles the TSN functions in hardware. The software stack covers 802.1AS, 802.1Qbv, 802.1Qcc and stream transformation; 802.1Qcc is driven from an embedded web page.

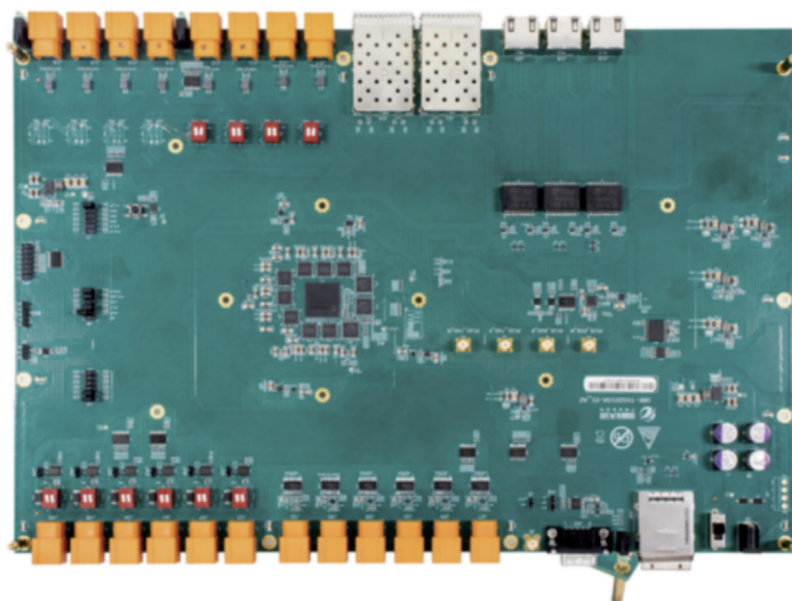


Figure 1. TSN gateway.

Physically, the box exposes one standard Ethernet port and two TSN ports, all 100Base-TX. The TSN ports can be cabled into star, ring, redundant or linear setups, which is what makes it useful for bring-up work: A non-TSN application can exchange deterministic streams with a TSN endpoint without touching the host stack.

The 5G network is a standalone (SA) ZTE deployment aligned with 3GPP Release 16. RAN is split AAU/BBU—BBU runs gNB processing, AAU is the radio front end. A ZTE core and Element Management System (EMS) round out the system; core and RAN software run on standard ZTE servers, and only the AAU hardware is purpose-built. All knobs are turned through ZTE's unified management UI.

TDD in the n78 band (3.3-3.8 GHz), a standard Sub-6 GHz slice in China. Round-trip latency averages about 10 ms—Enough for the bulk of real-time control. At 40 MHz and 100 MHz channels, peak downlink peaks at roughly 240 and 500 Mbps, and the slot format can lean downlink or uplink. Field-trial numbers are given in our earlier paper.

At the user end, a ZTE ZM9000-series module on an M.2-to-USB adapter, with an external antenna through SMA pigtailed, connects to the network. The module is Release 16, SA-mode, across a handful of Sub-6 GHz bands (n78 included), and in our setup it keeps an always-on data connection to the ZTE core.

Configuration happens in a web GUI, reachable from either TSN port. Once the gateway IP and client MAC are in, the operator flags which incoming streams are meant for the TSN fabric. Unflagged traffic is just ordinary best-effort Ethernet; it never touches TAS.

Flagged streams then get queued and scheduled. In 802.1Qbv language, the schedule is a set of transmission windows that repeat every base cycle; timing resolves to nanoseconds. **Figure 2** shows how we assigned streams to queues. The first configuration used a 200 ms cycle with a 25 ms window.

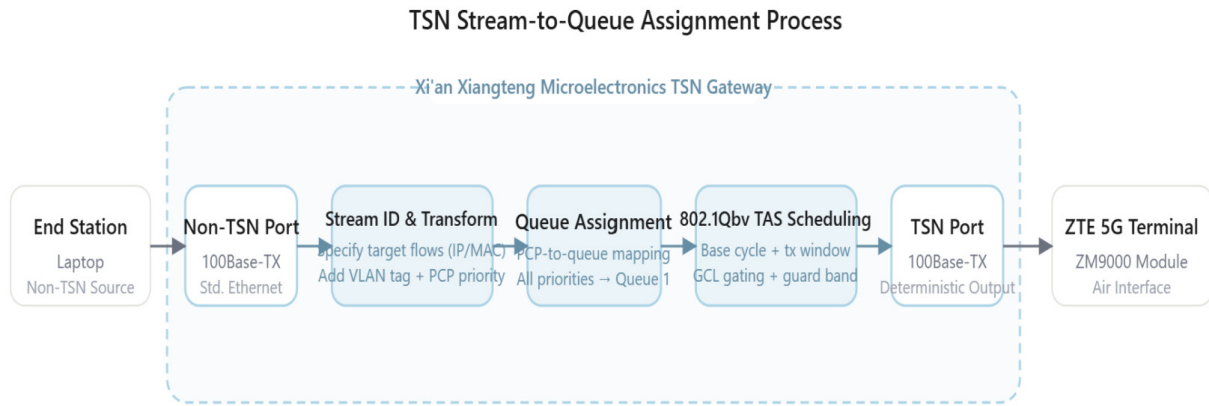


Figure 2. Stream-to-queue assignment.

4. Evaluation and results

Figure 1 sketches the setup. A laptop acts as the end station and pushes ordinary Ethernet into the gateway's non-TSN port. The gateway converts that traffic into scheduled TSN frames, and a ZTE 5G module—Attached to a TSN port through a USB 3.0-to-100M bridge, and on the 5G network by radio—Bounces the frames over the air to the 5G core. iPerf runs between the laptop and the core; Wireshark sniffs every hop. What we want to learn is how the 802.1Qbv schedule changes end-to-end behavior.

The tagging step is straightforward: Selected streams get VLAN IDs and priorities, land in their queues, and are pinned to a scheduled window—Frames leave only when the window is open.

Throughout the runs we hold the 5G network at 40 MHz with an uplink-skewed TDD pattern—Five slots and six symbols downlink, four slots and four symbols uplink. Standalone, the link delivers 11.36 ms average, 33.1 ms peak and 6.3 ms minimum round-trip latency. The module attaches at a 23 dB SINR, so radio quality is not the bottleneck. A dedicated network slice, with dedicated user-plane in both RAN and core, isolates the experiment.

Four scenarios, distinguished only by base cycle and window, are run back to back.

Scenario 1—200 ms cycle, 25 ms window. The source traffic is aperiodic Ethernet, but once the gateway stamps it, TSN frames leave on a strict 200 ms cadence. At the core, that cadence survives the 5G hop intact. The cycle is so long that each scheduled burst finishes well before the next gate opens.

Scenario 2—100 ms cycle, still 25 ms window. Cutting the cycle in half does not break the pattern. Both the gateway-side TSN stream and what reaches the core stay periodic at 100 ms; the 5G stack still has room to breathe.

Scenario 3—50 ms cycle, 12.5 ms window. Now the margins shrink. The gateway side still looks clean at 50 ms, and Wireshark flags a few lost frames—But those frames are in the 5G logs, so they did cross the air; Wireshark just misread them at capture. At the core, the 50 ms rhythm is visible but wobbles. No TSN-aware scheduling sits in RAN or core, and at this interval the generic 5G stack cannot keep up.

Scenario 4—40 ms cycle, 10 ms window. Same story, sharper. The gateway stamps frames at 40 ms, but what shows up at the core is only roughly 40 ms, and the losses this time look like genuine delay or drops inside 5G—Not capture noise. The pattern is clear: Off-the-shelf 5G, with no TSN awareness in the scheduler, cannot hold determinism once the cycle tightens past a certain point.

5. Conclusion

What this measurement campaign shows is straightforward. On a hybrid Xiangteng-TSN / ZTE-5G bench, end-to-end determinism is mostly dictated by the 5G stack's native ability to chew through 802.1Qbv scheduled traffic. A vanilla 5G network breaks down once the cycle tightens; without tuning the 5G side—Modeling TAS behavior end to end—The guarantees do not hold.

The same results push toward a single global scheduler. TSN constraints, TSN needs, and 5G resource

state all need to be in the same room, and configuration should be pushed down to both stacks rather than set by hand on each. Our testbed did it the hard way—Locally, through the gateway's web UI; a real deployment would let an edge controller drive the whole thing.

With a global scheduler on top, tight clock alignment between 5G and TSN stops being optional. Our bench still relies on loose sync; we have shown TSN frames can cross the radio, but the quantitative trade-off between sync accuracy and end-to-end determinism is a question for another paper.

Two caveats close the paper. The COTS box works, but software latency stacks up across protocol layers and quietly eats into determinism; purpose-built radios with native 5G/TSN convergence are the real answer. Separately, the ZTE DU does not schedule TSN traffic explicitly, and that alone makes tight cycles hard to guarantee. Window-based scheduling inside the 5G stack looks like the right place to dig in next.

References

- [1] IEEE Standard for Local and Metropolitan Area Networks—Bridges and Bridged Networks, IEEE Std 802.1Q-2018 (Revision of IEEE Std 802.1Q-2014), 2018.
- [2] IEEE Standard for Local and Metropolitan Area Networks—Bridges and Bridged Networks—Amendment 25: Enhancements for Scheduled Traffic, IEEE Std 802.1Qbv-2015 (Amendment to IEEE Std 802.1Q-2014), 2015.
- [3] IEEE Standard for Local and Metropolitan Area Networks—Timing and Synchronization for Time-Sensitive Applications, IEEE Std 802.1AS-2020 (Revision of IEEE Std 802.1AS-2011), 2020.
- [4] 3GPP TR 22.804, "Study on Communication for Automation in Vertical Domains," V15.1.0, 2018.
- [5] M. Gündall, C. Huber, P. Rost, R. Halfmann, and H. D. Schotten, "Integration of 5G with TSN as Prerequisite for a Highly Flexible Future Industrial Automation: Time Synchronization based on IEEE 802.1AS," in *Proc. IECON 2020 – 46th Annual Conference of the IEEE Industrial Electronics Society*, Singapore, Oct. 18–21, 2020.