Automatic Closed-Loop Control Applied to TSN Scheduling

Instituto Universitario de Investigación de Ingeniería de Aragón
Universidad Zaragoza

XIV JORNADA DE JÓVENES INVESTIGADORES/AS DEL I3A

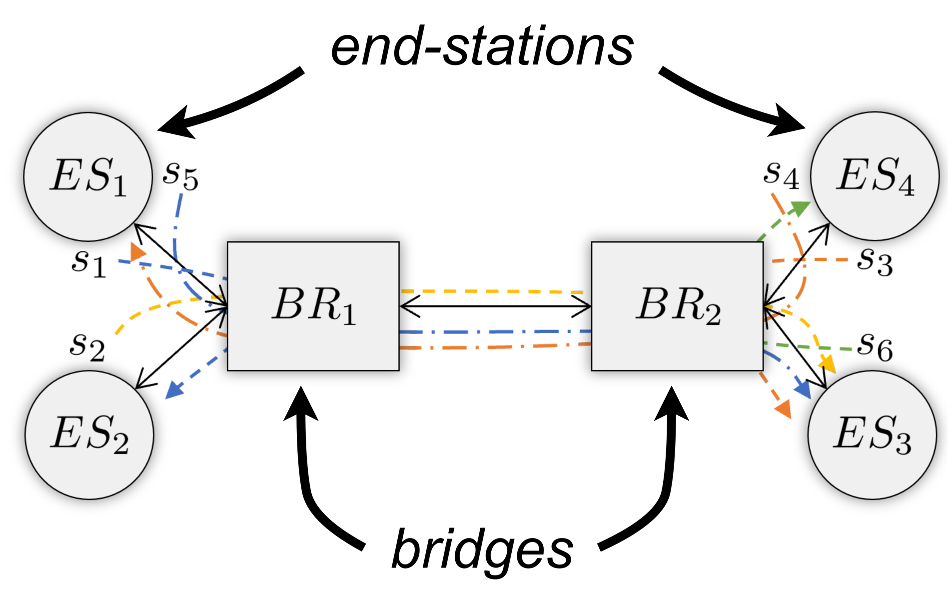
Alitzel Torres[1,2], Antonio Ramírez[2], José Luis Briz[1], Juan Segarra[1], Álex Gracia[1], Héctor Blanco[3]

Grupo de Investigación en Arquitectura de Computadores (gaZ)
Universidad Zaragoza
Cinvestav Guadalajara
intel.

Time-Sensitive Networks (TSN)

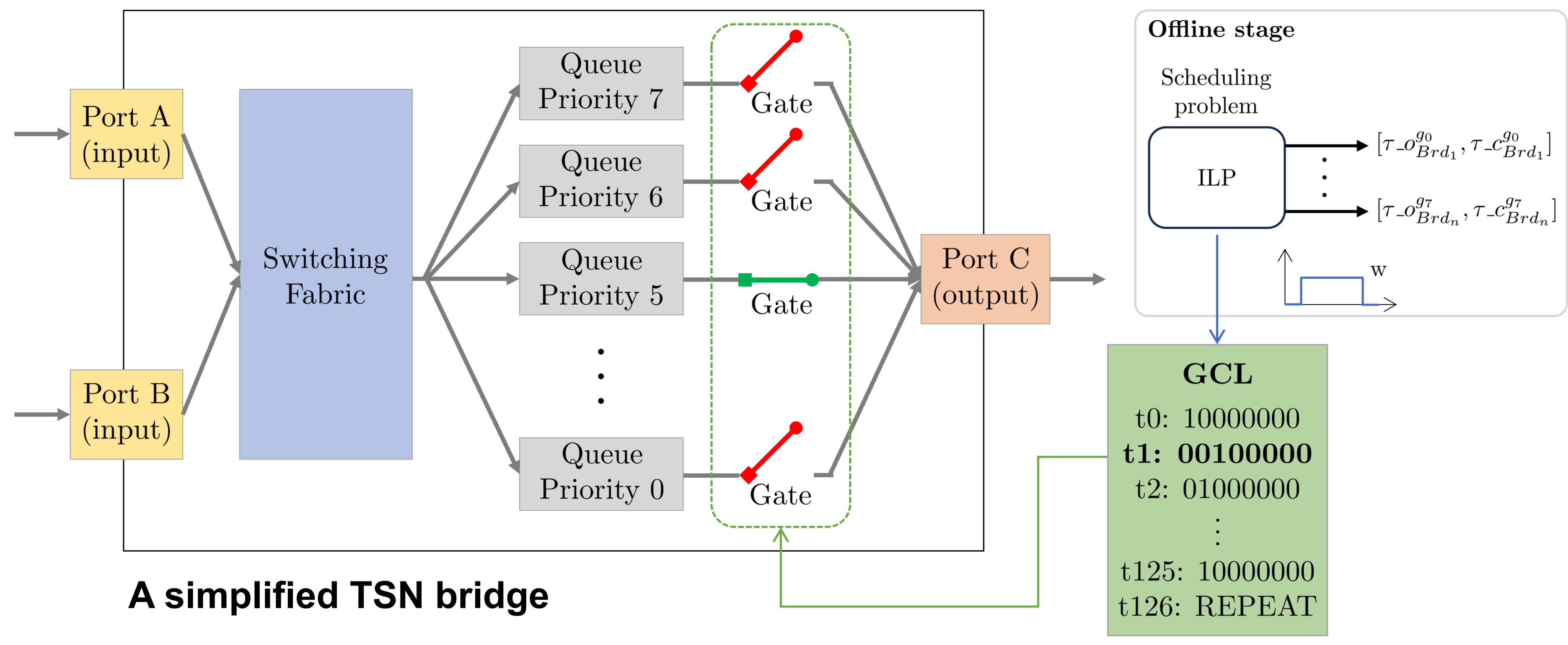
IEEE 802.1Q, a set of standards for:

- Deterministic communication (deadline, jitter).
- Coexistence of time-critical and non-critical traffic.
- Over interoperable Ethernet and wireless technologies.



IEEE 802.1Qbv Time-Aware Shaper (TAS)

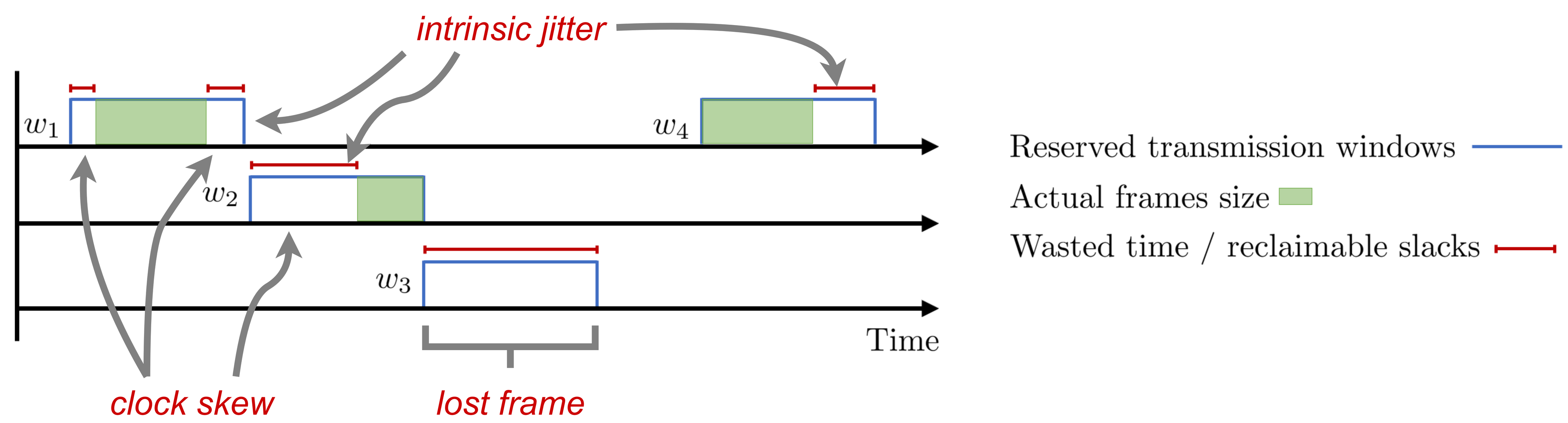
Ensures ultra-low jitter and bounded delays, assigning streams to queues and opening/closing gates through a Gate Control List (GCL).



Problem Motivation

Offline-computed schedules tend to be rather conservative, causing transmission windows to be larger than the actual size of the frames.

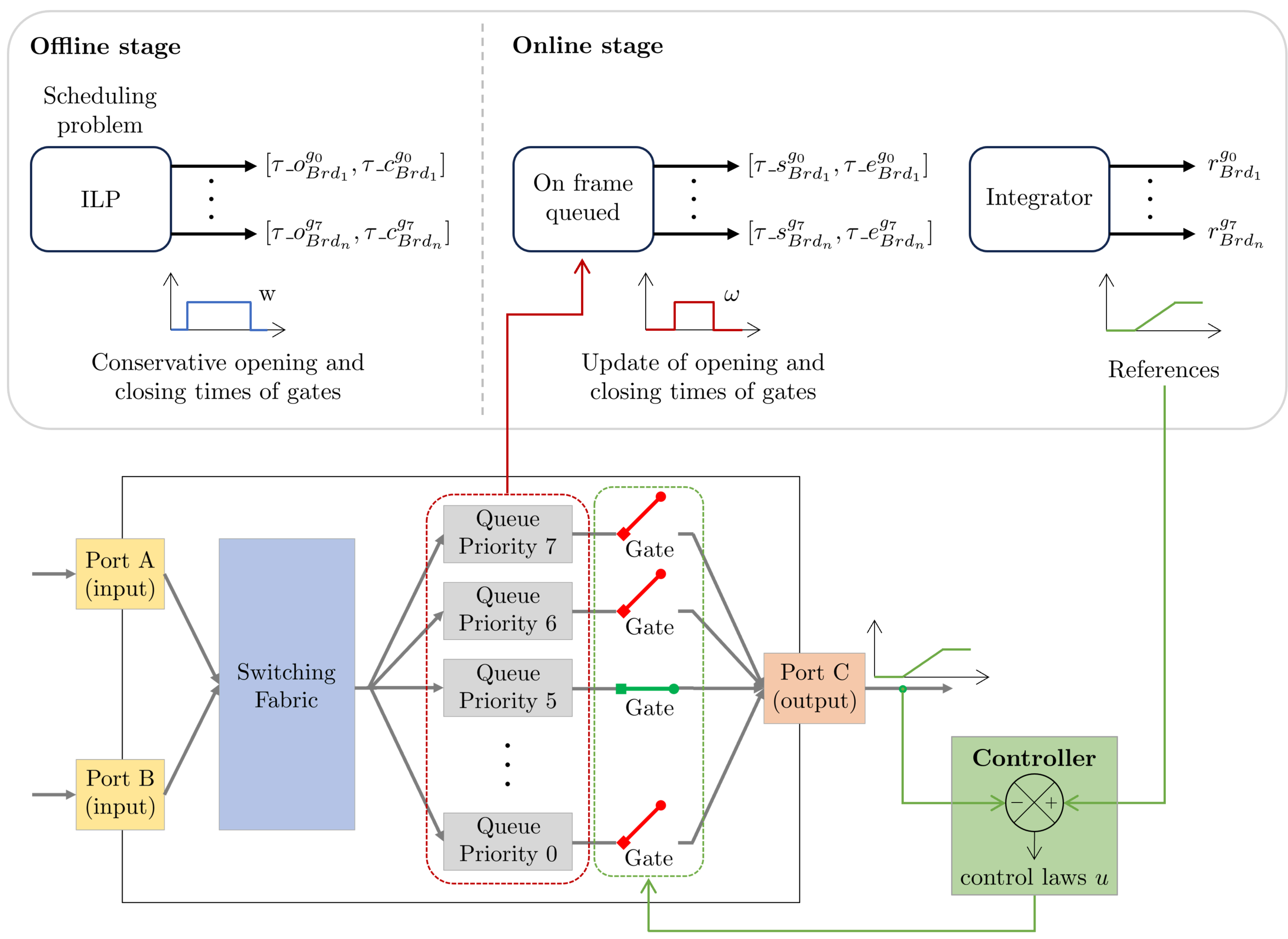
Objective:
Reclaim the slack of windows from the conservative scheduling for non-critical traffic.



Proposed Approach

A novel TAS implementation based on an automatic closed-loop controller:

✔ Replaces the GCL.
✔ Improves network utilization for non-critical traffic.



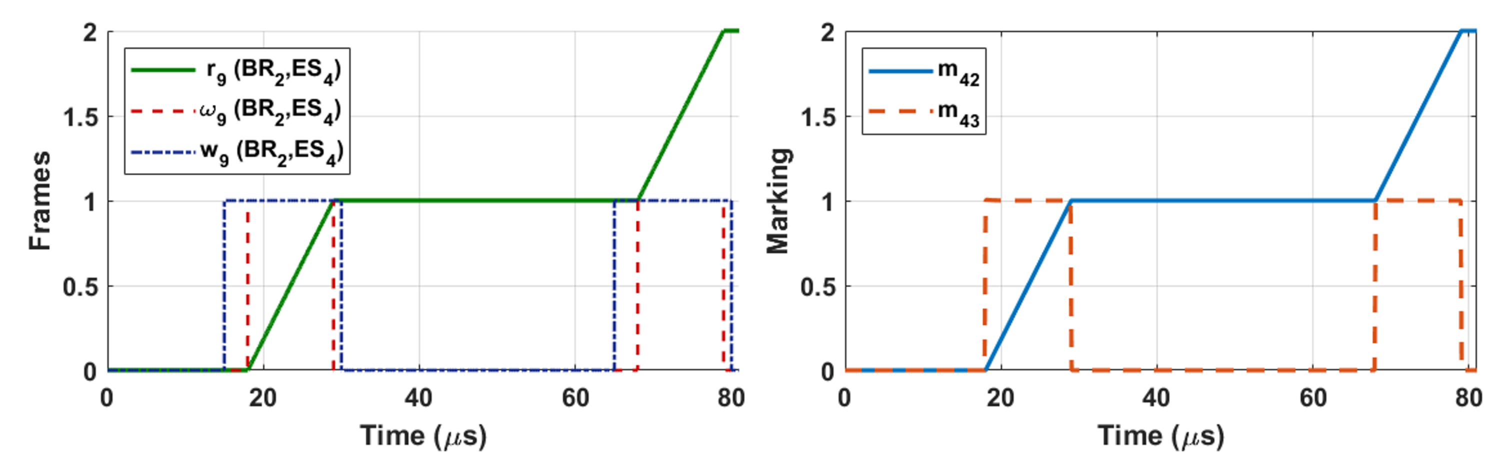
1. TSN model
2. Model analysis
3. Control design

➡

1. Timed Continuous Petri Nets (extension of [1]).
2. Assumed controllability and observability.
3. Backstepping control [2].

Case Study

An in-line network topology consisting of 2 *bridges* with 2 *end-stations* each. Six critical streams are scheduled using the method in [3], considering a maximum clock skew and a processing time of frames in *bridges* of 2 μs.



(a) Stream s_6 (b) Bridge BR_2, port ES_4

(a): updating windows and references generation processes for stream s_6.
(b): marking evolution of the TCPN representing the bridge BR_2 output port to end-station ES_4, traversed by stream s_6.

The reference obtained is tracked correctly, so the system output is the expected one.

Conclusions

Our implementation:

- satisfactorily tracks the error between the expected and the actual critical traffic,
- adjusts the schedule without stopping the TSN system to deploy the new TAS GCLs,
- makes the TAS resilient to parametric variations, and
- improves network utilization by tightening the conservative windows.

References

[1] A.G. Torres-Macías et al. Modeling Time-Sensitive Networking Using Timed Continuous Petri Nets. *IFAC-PapersOnLine*, 58(1):300–305, 2024.
[2] S. Vaidyanathan and A.T. Azar. *Backstepping Control of Nonlinear Dynamical Systems*. Advances in Nonlinear Dynamical Systems and Robotics (ANDC). Elsevier Science, 2020.
[3] A.G. Torres-Macías et al. Optimal and Fast IEEE 802.1Qbv Incremental Scheduling. *Manuscript under review*, 2025.

1 DIIS / I3A, Universidad de Zaragoza, 50018, Zaragoza, Spain. Email: {a.torres, briz, jsegarra, alex.gracia} @unizar.es
2 CINVESTAV Unidad Guadalajara, 45019, Zapopan, Mexico. Email: {alitzel.torres, antonio.ramirezt} @cinvestav.mx
3 Intel Deutschland GmbH, 85579, Neubiberg, Germany. Email: {hector.blanco.alcaine} @intel.com