On Time Synchronization Issues in Time-Sensitive Networks with Regulators and Nonideal Clocks

Ludovic Thomas
ludovic.thomas@epfl.ch
I&C, EPFL
Lausanne, Switzerland

Jean-Yves Le Boudec
jean-yves.leboudec@epfl.ch
I&C, EPFL
Lausanne, Switzerland

ABSTRACT

Flow reshaping is used in time-sensitive networks (as in the context of IEEE TSN and IETF Detnet) in order to reduce burstiness inside the network and to support the computation of guaranteed latency bounds. This is performed using per-flow regulators (such as the Token Bucket Filter) or interleaved regulators (as with IEEE TSN Asynchronous Traffic Shaping, ATS). Both types of regulators are beneficial as they cancel the increase of burstiness due to multiplexing inside the network. It was demonstrated, by using network calculus, that they do not increase the worst-case latency. However, the properties of regulators were established assuming that time is perfect in all network nodes. In reality, nodes use local, imperfect clocks. Time-sensitive networks exist in two flavours: (1) in non-synchronized networks, local clocks run independently at every node and their deviations are not controlled and (2) in synchronized networks, the deviations of local clocks are kept within very small bounds using for example a synchronization protocol (such as PTP) or a satellite based geo-positioning system (such as GPS). We revisit the properties of regulators in both cases. In non-synchronized networks, we show that ignoring the timing inaccuracies can lead to network instability due to unbounded delay in per-flow or interleaved regulators. We propose and analyze two methods (rate and burst cascade, and asynchronous dual arrival-curve method) for avoiding this problem. In synchronized networks, we show that there is no instability with per-flow regulators but, surprisingly, interleaved regulators can lead to instability. To establish these results, we develop a new framework that captures industrial requirements on clocks in both non-synchronized and synchronized networks, and we develop a toolbox that extends network calculus to account for clock imperfections.

CCS CONCEPTS

• **Networks** → **Network performance analysis**; *Time synchronization protocols*; • **Computer systems organization** → *Real-time systems.*

KEYWORDS

Time Sensitive Networks; Time Synchronization; Per-Flow Regulator; Per-Flow Shaper; Interleaved Regulator; Network Calculus

ACM ISBN 978-1-4503-7985-4/20/06.
https://doi.org/10.1145/3393691.3394206

ACM Reference Format:
Ludovic Thomas and Jean-Yves Le Boudec. 2020. On Time Synchronization Issues in Time-Sensitive Networks with Regulators and Nonideal Clocks. In *ACM SIGMETRICS / International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS '20 Abstracts), June 8–12, 2020, Boston, MA, USA.* ACM, New York, NY, USA, 2 pages.
https://doi.org/10.1145/3393691.3394206

MOTIVATION

Time-sensitive networks support real-time applications in avionics, space, and automobile. The time-sensitive networking (TSN) task group of the Institute of Electrical and Electronics Engineers (IEEE) and the Detnet working group of the Internet Engineering Task Force (IETF) aim to provide deterministic worst-case delay bounds.

Reshaping flows inside the network by means of traffic regulators helps achieve this objective: hardware elements are placed before a multiplexing stage to remove the increased burstiness due to interference with other flows in previous hops. Regulators support higher scalability and efficiency of time-sensitive networks.

When the internal logic of a regulator relies on a perfect clock, it enjoys the "shaping-for-free" property, i.e., a regulator that removes the burstiness increase caused by a first in, first out (FIFO) system does not increase the worst-case delay of flows [3]. In reality, the clock used by a regulator deviates slightly from true time. It can be part of a synchronized or a non-synchronized network: in synchronized networks, the deviations are kept within small bounds, using a time-synchronization protocol.

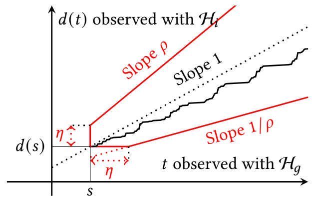
The ongoing standardization of regulators in TSN asynchronous traffic shaping (ATS) has raised discussions on the possible consequences of clock nonidealities when deploying regulators [1]. We provide theoretical foundations to the problem and we determine to what extend delay analyses are affected in non-synchronized and synchronized networks.

METHODOLOGY

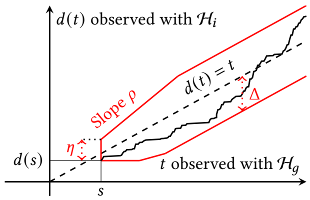
An upper-bounding time-model

We first build a time-model that relies on the model provided in [2]. However, instead of analysing the stochastic properties of clocks, we focus on bounding their relative evolution.

For any pair of clocks $(\mathcal{H}_g, \mathcal{H}_i)$ in the network, we denote by $d_{g\to i}(t)$ the time shown at clock $\mathcal{H}_i$ when clock $\mathcal{H}_g$ shows value t. $d_{g\to i}$ is the relative time-function from $\mathcal{H}_g$ to $\mathcal{H}_i$. For a given network of non-synchronized clocks, we define the *timing-jitter bound* η and the *clock stability bound* ρ, such that any pair of clocks $(\mathcal{H}_g, \mathcal{H}_i)$ in the network verifies, $\forall t \geq s$



(a) In the non-synchronized time-model.



(b) In the synchronized time-model.

Figure 1: Envelope of $d_{g\to i}(t)$ (red) and example of a possible evolution of $d_{g\to i}(t)$ (black).

$$\frac{1}{\rho}(t - s - \eta) \leq d_{g\to i}(t) - d_{g\to i}(s) \leq \rho(t - s) + \eta \quad (1)$$

Figure 1a presents, for a given known starting point $(s, d(s))$, the possible evolution space of $d(t)$ in the non-synchronized model as well as a possible trajectory. We note that the time-error function $d(t) - t$ can be unbounded in this model. η and ρ are network-wide parameters and do not depend on the clock pair. We show that for a TSN network, we can take η =4ns and ρ =1+2·10^{-4}.

If, in addition, clocks in the network are synchronized using a time-synchronization protocol, we further define the *time-error bound* Δ such that for any pair $(\mathcal{H}_g, \mathcal{H}_i)$, $d_{g\to i}$ meets the constraints of Equation (1), plus:

$$\forall t, |d_{g\to i}(t) - t| \leq \Delta \quad (2)$$

Figure 1b presents, for a given known starting point $(s, d(s))$, the possible evolution space of $d(t)$ in the synchronized model as well as a possible trajectory. Note that the Δ envelope is not centered on the starting point but on the $d(t) = t$ function.

A Network Calculus Toolbox

In time-sensitive networks, delays at network elements have to be bounded in worst case, not in average. To this end, network calculus is often used [3]. This framework uses cumulative functions such as $A(t)$, the total number of bits observed at some observation point up to time t. Traffic flows are assumed to be bounded by arrival curve constraints, of the form: $\forall t \geq s \geq 0, A(t) - A(s) \leq \alpha(t - s)$ (the function α is called "arrival curve"). A frequently used function is $\gamma_{r,b}$ defined by $\gamma_{r,b}(t) = rt + b$ for $t > 0$ and $\gamma_{r,b}(t) = 0$ for $t \leq 0$. It corresponds to a limit of a rate r and a burst b.

The service offered by a network element is also assumed to be lower bounded by a condition of the form $\forall t \geq 0 : D(t) \geq (A \otimes \beta)(t)$ where A [resp. D] is the input [resp. output] cumulative function, the function β is called "service curve" and the symbol $\otimes$ is the min plus convolution [3]. Many network elements can be modeled by "rate-latency" service curves $\lambda_{R,T}(t) = \max(0, R(t - T))$, and a FIFO network element that guarantees a delay upper-bounded by D offers the service curve δ_D defined by $\delta_D(t) = 0$ for $t \leq D$ and $\delta_D(t) = +\infty$ for $t > D$. Classic network calculus results give delay and backlog bounds at a network element, given some arrival-curve and service-curve constraints.

Using our time-model and its parameters η, ρ and Δ (if synchronized), we prove a set of results that can be used to obtain the arrival curve of a flow [resp the service curve of a server], as observed from a clock $\mathcal{H}_g$ (we note it $\alpha^{\mathcal{H}_g}$ [resp $\beta^{\mathcal{H}_g}$]), if we know an arrival curve [resp service curve] observed with a different clock $\mathcal{H}_i$ (and noted $\alpha^{\mathcal{H}_i}$ [resp $\beta^{\mathcal{H}_i}$]). The results are presented in Table 1 for the most common arrival and service curves.

Table 1: Relations between a leaky-bucket arrival curve [resp a rate-latency service curve] as observed with $\mathcal{H}_i$ and an arrival curve [resp service curve] as observed with $\mathcal{H}_g$.

	Leaky-Bucket Arrival Curve	Rate-Latency Service Curve
in $\mathcal{H}_i$	$\gamma_{r,b}$	$\lambda_{R,T}$
in $\mathcal{H}_g$, non-sync	$\gamma_{r\rho,b+r\eta}$	$\lambda_{R/\rho,\rho T+\eta}$
in $\mathcal{H}_g$, sync	$\gamma_{r\rho,b+r\eta} \wedge \gamma_{r,b+2r\Delta}$	$\lambda_{R/\rho,\rho T+\eta} \vee \lambda_{R,T+2\Delta}$

CONSEQUENCES OF CLOCK NONIDEALITIES ON TRAFFIC REGULATORS

Flow shaping (or re-shaping) is performed by regulators, that are either per-flow (PFR) or interleaved (IR). A PFR with an ideal clock, configured with arrival curve σ for flow f, makes sure that its output satisfies the arrival curve constraint σ (also called "shaping curve"). If the input data of flow f arrives too fast, the packets are stored in the PFR buffer (with one FIFO queue per flow), until the earliest time when it is possible to release the packet without violating the arrival curve constraint. An IR is similar to a PFR but all packets of all flows are stored in a single FIFO queue. The packet at the head of the queue is released at the earliest time when it is possible without violating the arrival curve constraint for this flow, and packets of other flows wait until they appear at the head of the queue.

In reality, the true enforced shaping curve deviates slightly from σ. For non-synchronized networks, we show that both types of regulators must be adapted: their configuration must take into account the parameters ρ,η. Otherwise they can lead to unbounded latencies. We detail two methods: rate and burst cascade and asynchronous dual arrival-curve method (ADAM) and we show that both incur a limited delay penalty on the end-to-end delay compared to the ideal situation with perfect clocks.

For synchronized networks, we show a fundamental difference: the penalty of non-adapted PFRs is controlled by the synchronization precision, however, even for tightly-synchronized networks, non-adapted IRs have unbounded delay latencies.

ACKNOWLEDGEMENTS

This work was supported by Huawei Technologies Co., Ltd. in the framework of the project *Large Scale Deterministic Network* (project number EPFL-LCA2-001). The authors thank Bryan Liubingyang for his contribution to the idea of this paper and Ahlem Mifdaoui for many fruitful discussions on an early version.

REFERENCES

[1] IEEE. 2019. Draft Standard for Local and Metropolitan Area Networks—Bridges and Bridged Networks—Amendment: Asynchronous Traffic Shaping. *IEEE P802.1Qcr/D2.0* For access, contact the IEEE802.1 chair. (Dec. 2019). http://www.ieee802.org/1/files/private/cr-drafts/d2/802-1Qcr-d2-0.pdf.
[2] ITU. 1996. Definitions and Terminology for Synchronization Networks. *ITU G.810* (1996). https://www.itu.int/rec/T-REC-G.810-199608-I/en.
[3] Jean-Yves Le Boudec and Patrick Thiran. 2001. *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet.* Springer-Verlag, Berlin Heidelberg. https://www.springer.com/us/book/9783540421849.