Insights on the Performance and Configuration of AVB and TSN in Automotive Ethernet Networks

Jörn Migge, RealTime-at-Work, France
Josetxo Villanueva, Renault Group, France
Nicolas Navet, University of Luxembourg, Luxembourg
Marc Boyer, ONERA, France

Abstract : Switched Ethernet is profoundly reshaping in-car communications. To meet the diverse real-time requirements in automotive communications, Quality-of-Service protocols that go beyond the mere use of priorities are required. In this work, the basic questions that we investigate on a case-study with diverse and demanding communication requirements is what can we expect from the various protocols aimed at providing a better timing Quality of Service on top of Ethernet? And how to use them? Especially how to use them in a combined manner. We will focus on the Credit-Based Shaper of AVB, the Time-Aware Shaper of TSN and the use of priorities as defined in IEEE802.1Q. The performance metrics considered are the distributions of the communication latencies, obtained by simulation, as well as upper bounds on these quantities obtained by worst-case schedulability analysis. If there have been over the last 5 years numerous studies on the performance of AVB CBS, the literature on comparing AVB to TSN and other candidate protocols is still sparse. To the best of our knowledge, this empirical study is the first to consider most protocols currently considered in the automotive domain, with the aim to gain insights into the different technological, design and configurations alternatives. In particular, an objective of this study is to identify key problems that need to be solved in order to further automate network design and configuration.

1 Quality of Service in automotive Ethernet networks

1.1 QoS through priorities, traffic-shaping and time-triggered communication

Quality-of-Service (QoS) in full-duplex Ethernet implies managing the interfering traffic in both the nodes and the communication switches. To achieve this, there are 3 main techniques, each of which being used by one of the 3 protocols under study:

- *Priorities*, as implemented in IEEE802.1Q with 8 priority levels, is a conceptually simple and widely used solution. Priorities, statically assigned to streams, have been used for instance in AFDX networks deployed in planes for over a decade with traffic shaping in emission and traffic policing in the switches. Two limitations of the use of priorities without traffic shaping are 1) it can lead to starvation for the lower-priority streams and 2) it does not offer support for bandwidth reservation.
- One way to overcome these issues is to use *traffic shaping* policies. This is what is done in AVB with the Credit-Based Shaper (AVB/CBS for short, defined in IEEE 801.Qav part of [IEEE802.1Q-2014]). The specification defines 8 classes of traffic: 6 best-effort classes, with priority scheduling among them, and Service Reservation A (SR-A) and B (SR-B) at the two top priority levels. The two latter classes are subject to the CBS to reduce the interferences brought to the other classes. Although the specification defines the standard SR-A and SR-B classes, it is possible to use the two AVB classes configured differently. This is what we refer to as *AVB custom-classes* (see §3.2). If AVB has not been specifically conceived for the automotive industry, its QoS mechanisms and the availability of compliant hardware components makes it a good candidate for in-vehicle communications. The reader is referred to [Que12], [RuBo14], [LiGe16], [CaCuBrLu16] and [Ashjaei17] for descriptions and analyses of the CBS mechanisms.
- A different paradigm to manage the interferences between streams is *time-triggered* (TT) *communication* where time-windows are reserved for the transmission of certain flows. TTEthernet and, earlier, TTP, both from the company TTTech, are prominent representatives of TT networks. Time-Sensitive Networking (TSN, see http://www.ieee802.org/1/pages/tsn.html) is a broadly supported IEEE initiative defining QoS mechanisms at the MAC level, an important one being the Time-Aware Shaper (TSN/TAS for short, defined in IEEE 802.1Qbv-2015) that can be used to implement TT communication for a chosen subset of the traffic. The efficiency of a TT system essentially depends on the ability to build an efficient global schedule, which has to be done by configuration tools, and this holds true for TSN/TAS. TSN/TAS's purpose is to increase the performance and the predictability of the most stringent traffic and, importantly, it can be used in conjunction with AVB CBS. The reader can consult [ThErDi15] and [Ashjaei17] for a description and an analysis of TSN/TAS.

In the rest of the study, we consider the use of priorities assigned statically to the streams, that will be referred to as Static-Priority Ethernet or IEEE802.1Q, as well as AVB/CBS and TSN/TAS. Figure 1 shows a communication traces where best-effort frames get the chance to be transmitted sooner when the higher priority video streams are shaped with AVB/CBS.

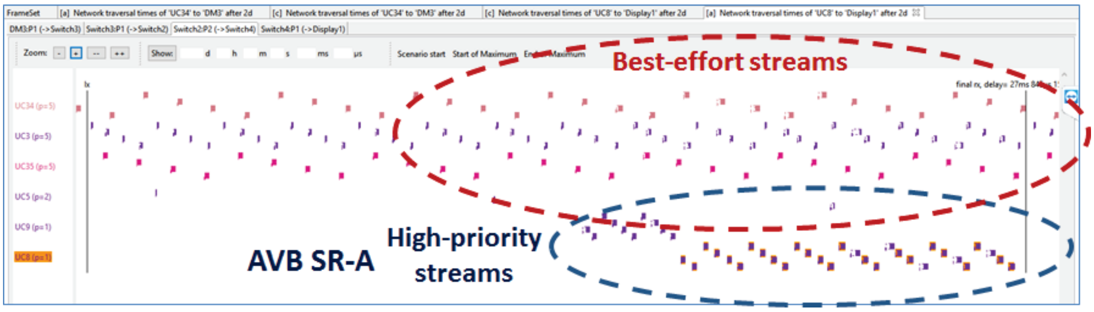

Figure 1: Illustration of traffic shaping under AVB/CBS. The transmission of the frames of the best-effort and higher-priority video streams under AVB are intertwined due to the CBS shaper on the link. This reduces the latencies for the best effort streams. Traces obtained by simulation in RTaW-Pegase.

1.2 Implementation in the switches

The protocols described in the previous paragraph are supported by the communication switches. Although the actual implementation will differ, Figure 2 shows a schematic view of the output port of a switch with

- up to 8 different classes of traffic corresponding to distinct priority levels,
- AVB CBS with two different traffic classes of adjacent priorities,
- TAS which defines at any point in time whether the transmission gate for a queue is open or closed, *i.e.* whether transmission for a certain traffic class is possible or not. In this study, we consider exclusive bus access for the messages of a designated traffic class[1], which means that whenever the gate is open for that class, the gate is closed for the rest of the traffic classes, and vice-versa. Exclusive bus access implies absence of interference[2] from the rest of the traffic whatever the actual priority value assigned to it.

In case several frames are ready for transmission, e.g. an AVB frame (with non-negative credit for the class) and a best-effort frame, highest priority first scheduling decides which frame gains the link access.

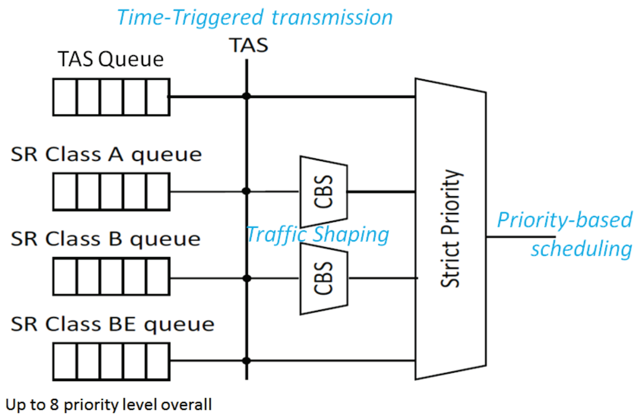

Figure 2: Schematic view of the output port of a switch implementing priority scheduling, AVB/CBS and TSN/TAS (figure from [Ashjaei17] with additions). At any point in time, the TSN gate for a given traffic class is either open or closed, in the latter case no frames of the traffic class can be transmitted.

Nodes should support the QoS protocols too but not necessarily all of them. For instance, some nodes may not support AVB/CBS, and thus will send unshaped traffic streams to the first switch. In that case, only the limit imposed by the parameters of the AVB class in terms of number of frames per CMI and maximum frame size must be satisfied.

[1] This traffic class having exclusive bus-access will be sometimes referred to as "TAS-scheduled".
[2] It should be noted that exclusive bus access under TAS is different from having the highest priority since no lower-priority frame can be sent when the TAS gate is closed for their traffic class.

2 Renault Ethernet prototype network

2.1 Topology and traffic

The case-study is a prototype Ethernet network provided by Renault comprising 5 switches and 14 nodes: 4 cameras, 4 displays, 3 control units and 3 (functional) domain masters, as shown in Figure 2. The data transmission rate is 100Mbit/s on all links except 1Gbit/s on the link between domain master 3 (DM3) and switch 3.

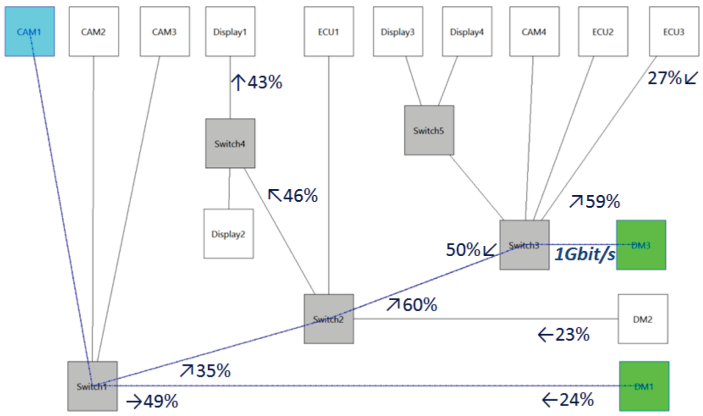

Figure 3: Topology of the prototype network under study. The multicast stream shown here goes from camera 1 to domain masters 1 and 3 (RTaW-Pegase screenshot). The graphic shows the 10 most loaded links, with a maximum of 60% load, and the single 1Gbit/s link.

The traffic is made up of 4 classes for a total of 41 streams whose characteristics are summarized in Table 1.

Audio streams	– 8 streams – 128 and 256 byte frames – up to sub-10ms period and deadline – soft deadline constraints
Video Streams	– 2 ADAS + 6 Vision streams – up to 30*1446byte frame each 16ms (60FPS) or each 33ms (30FPS) – 10ms (ADAS) or 30ms deadline (Vision) – hard and soft deadline constraints
Command & Control (C&C)	–11 streams, 256 to 1024 byte frames – up to sub-10ms periods and deadlines – deadline constraints (hard)
Best-effort: file & data transfer, diagnostics	– 14 streams including TFTP traffic pattern – up to 0.2ms period – both throughput guarantees (up to 20Mbits per stream) and deadline constraints (soft)

Table 1: Characteristics of the 4 types of traffic. The performance requirement is either to meet timing constraints (soft and hard deadlines) or throughput constraints. The frame sizes indicated are data payload only.

Upcoming Ethernet networks, as exemplified in this case-study, will support mixed-criticality traffic which implies a diversity of communication requirements: deadlines (soft and hard), bandwidth, segmented messages, client-server transactions, constraints on memory usage, etc. The overall complexity appears to us more important than in the current CAN-based architectures with dedicated multimedia links. With respect to the specific case-study under consideration in this work, it should be noted that there are several demanding communication requirements such as

– 10ms deadline for ADAS camera frames, this means that the whole camera frame, that is 30 Ethernet frames, must be received within 10ms,
– 10Mbit/s and 20Mbit/s throughput constraints for TFTP streams, which, given the frame size used, leads to transmission periods as low as 0.2ms for some best-effort streams.

2.2 Verification techniques and tools

The first performance metric considered is the frame worst-case communication latency, also called WCTT for *Worst-Case Traversal Times*, that is the time it takes in the worst case to receive an Ethernet frame, or the last frame of a segmented message in the case for instance of a video stream. Bounds on the WCTTs are obtained using a state-of-the-art analysis in Network-Calculus, a formalism for instance employed for AFDX that has proven to scale very well with a good numerical accuracy (see the experiments in [BoNaFu12]). WCTT bounds serve to check whether deadline requirements are met. The second performance metric is the throughput guaranteed to certain streams, which is evaluated by simulation. Precisely we use what we refer to as *timing-accurate simulation*, which means that every operation that takes time is captured by the simulation model.

Both timing-accurate simulation and WCTT analysis are complementary. Indeed, if WCTT is the safest approach, it is inherently pessimistic. In addition, it does not provide statistics such as the distribution of the latencies or, for instance, an accurate evaluation of the throughput that can be achieved for FTP-like streams. The reader can refer to [NaSeMi16] for a discussion on how WCTT analysis and simulation are complementary in the design of automotive communication architectures.

The modelling and timing analysis tool used throughout this study is RTaW-Pegase v2.4.5, a product of RealTime-at-Work developed in partnership with ONERA lab since 2009. The simulation samples were collected over long simulations (2 days of uninterrupted functioning, about 17 million transmissions for a 10ms stream) with the clock drift of each station set to a random value within ±200ppm. WCTT analysis takes a few seconds to execute while simulations require around 3 hours on a single core of a standard desktop workstation (Intel I5 processor).

2.3 Solutions experimented

As discussed in Section 1, several QoS paradigms and protocols are available to the designers with many ways to configure them. In this study, we looked for solutions meeting all communication requirements of the case-study with different design choices in terms of protocols selection and configuration. Following solutions have been evaluated and will be discussed in the rest of the paper:

1. Standard AVB classes with C&C traffic as best-effort,
2. AVB “Custom-Classes” with C&C traffic as best-effort,
3. IEEE802.1Q with and without “pre-shaping” in transmission,
4. AVB “Custom-Classes” with C&C under TSN/TAS.

Ultimately, the goal is to get insights into the scope of applicability of each of the protocols, and into how to combine and configure them.

3 Experimental results

3.1 Solution #1 – AVB standard classes

The first solution evaluated is based on AVB/CBS with the use of the standard SR-A and SR-B classes. The important benefit of using the standard classes is that, provided certain load conditions and certain traffic transmission patterns are met, latencies under 2ms over 7 hops are guaranteed for SR-A streams and under 10ms for SR-B streams (see [Avnu15]). In addition, when using the standard classes, the configuration of the AVB/CBS Idles Slope (IS) for each class is straightforward for the designer since the values of the IS should be set to the minimum bandwidth required for the streams of the class.

In this experiment, the priority allocation among the traffic classes is by decreasing priority order as follows:

1. Audio and ADAS video streams under AVB/CBS SR-A,
2. Standard video streams under AVB/CBS SR-B,
3. Command and Control streams are at the third priority level, called here "highest priority best-effort",
4. Finally, file, data transfer and diagnostics streams are at the lowest priority level.

Although this constraint may disappear in the forthcoming TSN IEEE802.1Qcc specifications, the two AVB classes had to be placed above C&C in terms of priority considering the state of today’s technologies.

Let us consider the 30FPS ADAS video stream named UC36 whose native format is 30x1400bytes frames sent every 33ms - with 10ms to receive the last frame of the segmented camera image. The stream is re-shaped by the sending node to meet the AVB SR-A traffic pattern as follows. Let us assume the conditions hold for SR-A to ensure the 2ms latency, there are thus 8ms left to send the entire image with an Ethernet frame every 125us as shown in Figure 4. 125us is the value of an important parameter for AVB SR-A, called the Class Measurement Interval (CMI, see [Avnu15]). In this setup, the data payload of each of the 8ms/125us=64 frames is 703 bytes. This is the optimal configuration in the sense that it minimizes the peak load (i.e., load over the 125us CMI intervals) as defined in the applicability conditions of the latency guarantees, and thus no other configuration will lead to a feasible solution if this one fails.

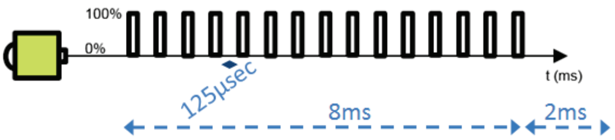

Figure 4: 30-FPS video stream send under AVB SR-A with a 703 bytes Ethernet frame every 125us. 64 frames are sent over 8ms.

Standard AVB does not provide a solution here for the following reasons. The peak load for stream UC36 alone is 46%, this is partly due to the overhead of using smaller frames than the native format. As, along the path of UC36, there are other SR-A streams including another identical ADAS video stream, the 75% load constraint needed for the 2-ms guarantee is not met. If the 2-ms AVB guarantee cannot be assumed, a solution would have been to show that the latencies for UC36 are smaller than 2ms using the WCTT analysis, more precise and with a broader scope of applicability than the AVB bound. This is however not the case since the load on one link exceeds 100% and this thus rules out that the AVB SR-A solution is feasible.
We then tried to relax the deadline to 15ms, which under SR-A results in 104 frames of 450 bytes sent every 125us over 13ms. The AVB condition could still not be applied but the WCTT analysis shows that the 2ms worst-case latency actually holds. However, a feasible solution could still not be reached, as some deadlines were not met in the C&C traffic transmitted at the priority level below AVB. This experiments shows that in the automotive context when they are stringent deadlines, standard AVB is not always flexible enough to offer an answer, even for video streams for which it has been conceived.

3.2 Solution #2 - AVB custom classes

To schedule the two ADAS videos streams, we now use AVB/CBS with custom classes, that is not using the standard 125/250us CMI and standard Idle Slopes which do not lead to a feasible solution as explained in the previous paragraph. The priority allocation remains identical as for standard AVB.
An advantage of custom classes over standard AVB classes is that the video streams do not need to be “repackaged” into smaller frames, which saves overhead in time and bandwidth. With custom classes, streams can be send in their native format. For instance, for stream UC36, 30 frames are sent each 33ms, with the CMI parameter of AVB set to 33ms. The time between two subsequent frames of a stream over the length of a CMI depends on the value of its Idle Slope, which should be chosen for each output port over the path of the stream so that deadlines are met. We developed for that purpose an algorithm that set the Idle Slopes for an AVB class to the minimal values so that the deadline constraints of all the streams in that class are met. We want to create the least interference to the lower priority traffic, and spread the transmission of AVB streams over time as much as possible, thus the choice of the minimal Idle Slope values. This algorithm implemented in RTaW-Pegase is called *Tight Idle-Slope*.

WCTT analysis and simulation shows that AVB custom-classes configured with Tigh Idle-Slope offer a solution that meets all the performance objectives of the case-study:
- meeting deadlines for ADAS and non-ADAS video streams, C&C traffic and best–effort traffic,
- meeting throughput requirements for best-effort streams.

Figure 5 shows, by comparison with IEEE802.1Q, the worst-case and average communication latencies obtained for the best-effort streams with AVB custom classes configured with the Tight Idle-Slope algorithm. As an illustration of the gain, the TFTP stream UC30 meets its 10Mbit/s bandwidth objective as both the average latencies of the request and response are below 0.4ms.

This was not possible to achieve under IEEE802.1Q without additional pre-shaping in transmission (see §3.3). On Figure 5, AVB *Minimal Idle Slope* is a configuration where Idle Slopes are set to the minimal bandwidth required, as it is done for the standard SR-A and SR-B classes. It does not lead to a feasible system but only serves to show the limited impact on the best-effort stream of the larger Idle Slopes needed to meet the deadlines constraints for ADAS video streams.

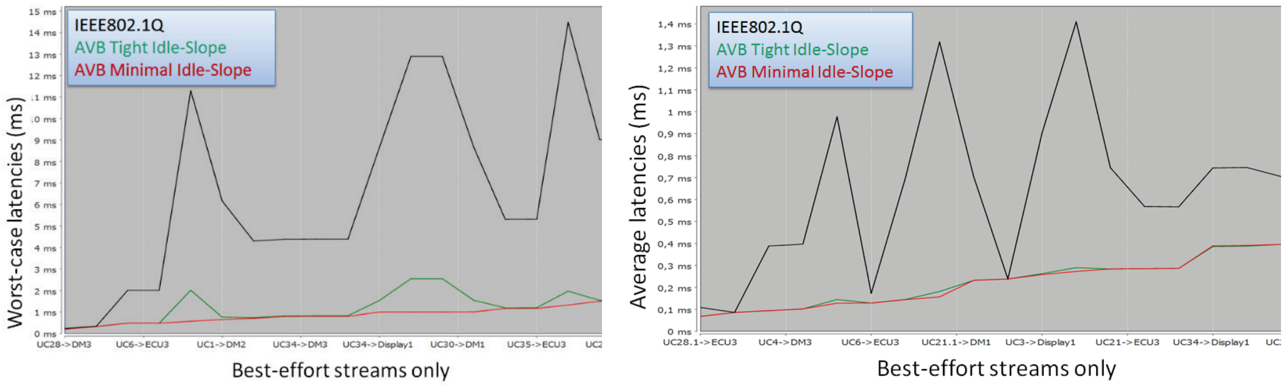

Figure 5: The left-hand graph shows that the worst-case latencies for the best-effort streams under AVB with Tight Idle-Slope is importantly improved over IEEE802.1Q: by 73% on average over all streams, and up to 87%. This is the case also for the average latencies (right-hand graphic) which are improved using AVB with Tight Idle-Slope algorithm over IEEE802.1Q by 54% on average, and up to 86%.

This experiment suggests that we can push the scope of applicability of AVB not using the standard CMI and with application-specific Idle Slopes. This however cannot be done manually, it needs to be automated with configuration tools. In addition to requiring tool support, parameters of custom-classes are more likely to require re-configuration than standard classes if traffic is added to the network.

3.3 Solution #3 - IEEE802.1Q with pre-shaping

As explained in the previous paragraph, the use of IEEE802.1Q does not lead to a feasible solution because the bandwidth requirements for some best-effort streams cannot be met. Here, we explore the use of "pre-shaping" strategies under IEEE802.1Q for bursty traffic such as video streams, as a simple and efficient alternative to AVB/CBS (see §3.2) and TSN/TAS (see §3.4). The pre-shaping mechanism combines standard priority scheduling with traffic shaping, which is introduced by inserting idle times, pauses, between the times at which the successive frames of a segmented message, typically a camera frame, are enqueued for transmission. All the other characteristics of the traffic remain unchanged. This strategy, applied on the sending nodes on a per flow basis, is conceptually simple and easy to implement in software. Intuitively, pre-shaping allows lower or same priority frames that cross the path of pre-shaped streams to be transmitted sooner, taking advantage of the inserted idle times (see [NaMiViBo18] for a more comprehensive discussion on the pre-shaping mechanism).

Pre-shaping has been applied to the 8 video streams of the case-study. The choice of idle times has been done manually until reaching the configuration shown in Figure 6 that meets all performance constraints. The priority allocation remains unchanged with respect to the IEEE802.1Q configuration without pre-shaping; one has by decreasing priority order: Command & Control, Audio, Video, and best-effort streams at the lowest priority level.

Name	Priority	MinDistance	MaxSize	Sender	Receiver
UC9	2	3 ms / 32 ms	10 x 1246 byte	DM3	Display2
UC8	2	1 ms / 32 ms	30 x 1446 byte	DM3	Display1
UC10	2	1 ms / 32 ms	30 x 1046 byte	DM3	Display3
UC11	2	1 ms / 32 ms	30 x 1046 byte	DM3	Display4
UC26	2	1 ms / 32 ms	30 x 1446 byte	CAM1	DM3
UC32	2	0,5 ms / 16 ms	30 x 1446 byte	CAM4	DM3
UC36	2	0,324 ms / 32 ms	30 x 1446 byte	CAM3	DM1
UC37	2	0,324 ms / 32 ms	30 x 1446 byte	CAM2	DM1

Figure 6: Pre-shaping configuration for the 8 video streams. The first duration in the MinDistance column indicates the idle time between two successive packet transmissions, while the second duration is the time between two successive camera frames. As can be seen, the latter values are set in the tool to be either 16 or 32ms, instead of the native values of 16.66 (60FPS camera) and 33.33ms (30FPS camera). These conservative assumptions enable to speed up the WCTT analysis which involves computations over the LCM of the transmission periods.

Figure 7 shows the worst-case and average communication latencies obtained for the best-effort streams with and without pre-shaping. The use of pre-shaping for audio/video streams in this case-study leads to a drastic reduction of the communication latencies for the best-effort streams while enabling to meet the timing constraints for the rest of the traffic. In addition, as shown in [NaMiViBo18], pre-shaping has virtually no impact on the higher priority traffic. Both pre-shaping and AVB custom classes are feasible solutions here, and they perform almost identically for the average latencies of best effort streams. However, besides not requiring dedicated hardware, pre-shaping has the advantage over AVB that the command and control streams are sent at the highest priority level, which reduces their latencies. This also improves the robustness of the system since the priority levels reflect the actual criticality of the streams.

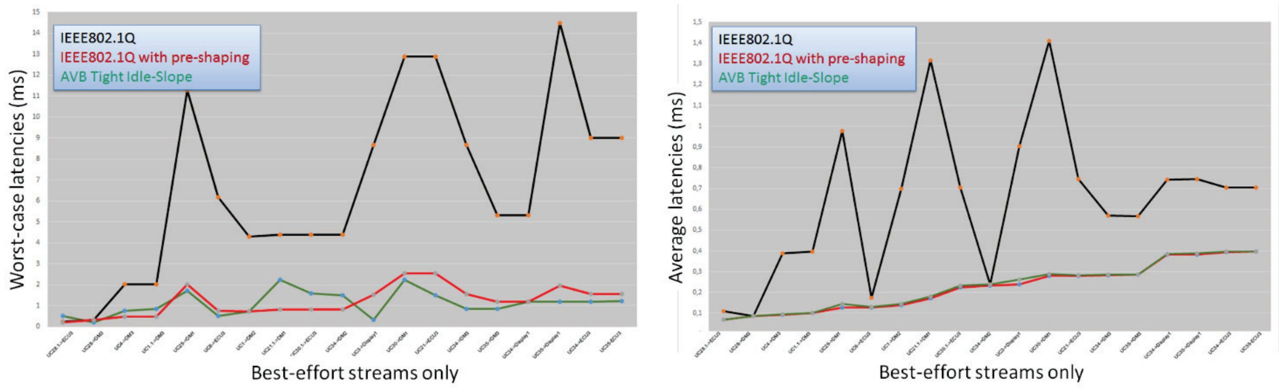

Figure 7: The left-hand graph shows that the worst-case latencies for the best-effort streams under IEEE802.1Q are greatly improved with pre-shaping: by 66% on average and up to 90%. The same observation holds for average communication latencies (right-hand graph) with a latency reduction of 54% on average, and up to 86%. For both worst-case and average latencies, pre-shaping results are comparable to the results obtained with AVB custom classes and the tight Idle-Slope algorithm shown in the green curves.

The pre-shaping policy with priority scheduling proved to be a simple and effective strategy on this case-study. It however has some limitations:

- It does not protect against a "babbling idiot", that is a node that would send outside its specification, and for instance flood the network. Two solutions may be used: either a per class shaping, like with CBS in AVB or TSN, or a per stream shaping, like in AFDX or in PSFP (IEEE802.1Qci).
- Adding a new function or a new ECU, which results in adding frames to the system, may require a reconfiguration of the pre-shaping parameters for all the flows since the maximal communication latencies will change. This limitation is not specific to pre-shaping and affects most of the QoS protocols except standard AVB which is, within certain limits, more robust in that regard.
- The deadline of a message may impose to assign a high priority to a stream to meet the timing constraints while the stream is not important from a functional point of view. This limitation affects all schemes based on static priority.
- Setting the parameters for the flows subject to the pre-shaping mechanism is a time consuming task when done by trial-and-error. To the best of our knowledge, there are no guidelines, such as optimality results, available to guide the designer in this task. The process of setting parameters should be automated which requires specific tool support.
- As there is no re-shaping along the path of a message, unlike for instance in AVB/CBS or TSN/TAS, the efficiency of the pre-shaping will decrease with the number of hops and thus with the size of the network.
- From the OEM perspective, pre-shaping imposes requesting additional features to ECU suppliers, which has a cost. However, just like transmission offsets in CAN, pre-shaping may be implemented only on a subset of the nodes of a given network. For instance, in our case-study only 5 nodes out of 14 were using pre-shaping in transmission.

3.4 Solution #4 - TSN/TAS to reduce C&C latencies

Finally, the last solution evaluated is the combined use of TSN/TAS with AVB/CBS custom classes. The aim is to estimate the extent to which TAS can help to reduce the latencies for critical streams, here the C&C traffic. In this experiment, all C&C streams are isolated in a dedicated traffic class and TAS gate operations are configured such that "reserved windows" are created that guarantee exclusive bus access to the C&C traffic class., i.e., no other traffic class is allowed to transmit at the same time. Except that the C&C traffic is isolated using TAS, the configuration is identical to the one used in §3.2. In particular the Tight Idle-Slope algorithm is used for the configuration of AVB. Over the use of AVB/CBS alone, this solution possesses the advantage from the dependability point of view that the C&C streams are sent at the highest priority level and fully segregated from the rest of the traffic.

The efficiency of any time-triggered protocol importantly relies on the quality of the transmission schedule. TAS is the most efficient when the gate schedule tables on the sending node and all switches are coordinated, so that the TAS scheduled frames wait as little as possible along their path. For instance, the situation where a frame arrives in a switch and must wait for a long time until the next TAS window should be avoided. On the other hand, TAS windows should not be too frequent and large otherwise the latencies for the rest of the traffic will suffer, and the memory needs will increase in the switches. To build the gate schedule tables, we developed an algorithm, based on mathematical reasoning combined with domain-specific heuristics, which aims to minimize the WCTT for the TAS-scheduled traffic. Because it favours TAS traffic over all other traffic classes, this algorithm implemented in RTaW-Pegase is called the *ASAP* algorithm.
We also assume here that tasks and frames schedules are synchronised: a frame/signal is transmitted in a TAS window immediately after its production by a task. This is the best possible situation for time-triggered communication. In many practical cases however, often by design choice to decouple design activities, both schedules will not be perfectly synchronised and some delays will arise as illustrated in Figure 8. To some extent, we can mitigate this issue by reserving more and/or longer TAS windows but this is going to be at the expense of a less optimized network utilisation and increased latencies for the rest of the traffic.

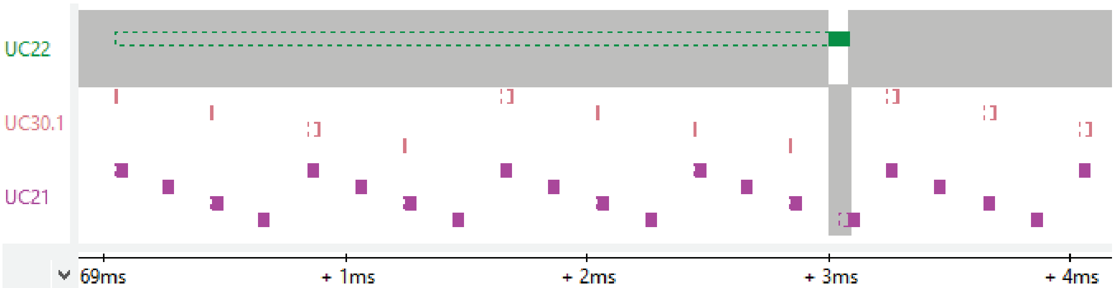

Figure 8: Situation where task and frame scheduling are poorly coordinated. The TAS windows for stream UC22 is scheduled long after the production of the frame and thus the communication latency for that frame increases.

In the case where TAS windows are allocated with a distance equal to the period of the frame(s) to be sent during these windows, like it has been done in this study, not synchronizing task and message schedules means that deadlines less than or equal to the period of the frames cannot be met.

Figure 9 shows the maximum communication latencies for the C&C streams only (left-hand graphic) and for the totality of the traffic. It shows that in this case-study TAS is effective in reducing the maximum latencies for the C&C traffic with a negligible impact on the rest of the traffic. Here, however, all 3 solutions shown in Figure 9 lead to acceptable solutions for the C&C traffic.

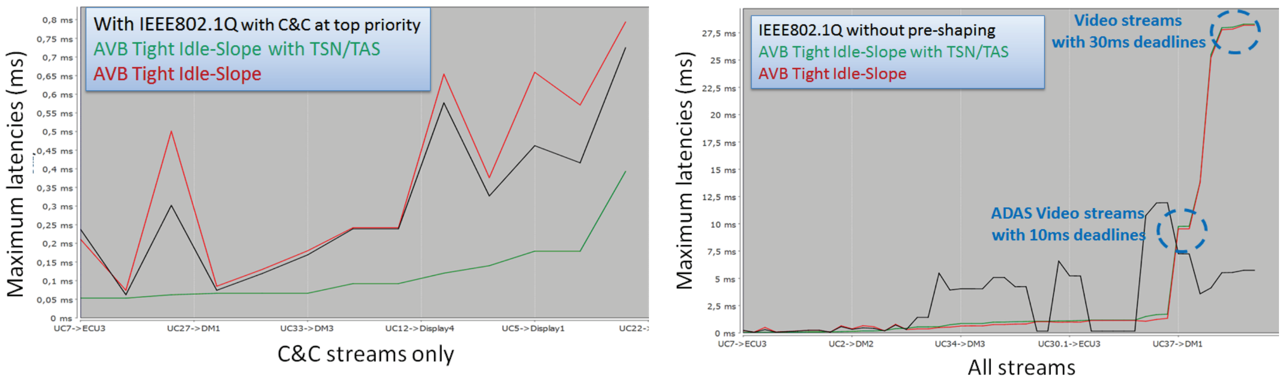

Figure 9: Using TSN/TAS for C&C traffic with ASAP algorithm improves maximum latencies for C&C streams by 54% on average over IEEE802.1Q and by 60% over AVB without TAS (left-hand graphic). The right-hand graphic shows the delay for all the streams with the 3 protocols under study. The maximum latencies of Audio/Video/Best-effort are almost unaffected by TAS: +3% on average when TAS is used.

In our case-study, the use of TSN/TAS in conjunction with AVB/CBS allows to really fine-tune the QoS provided to each class of traffic and led to the overall best results among the solutions experimented. This comes at the price of a certain complexity in terms of the protocols stack and, without proper tool support, important configuration efforts.

3.5 Memory usage in the switches

Figure 10 shows upper bounds on the memory usage in the output ports of the switches obtained by mathematical analysis. AVB standard, which shapes the traffic in a very efficient manner thanks to Idle-Slopes set to the minimum required bandwidth, leads to the lowest memory usage. As shown in §3.2, it however does not offer a feasible solution. On the other end of the spectrum, IEEE802.1Q without pre-shaping creates bursts of frames, which accumulate in the switches. IEEE802.1Q with pre-shaping in transmission improves the memory usage by a factor 2 on average over IEEE802.1Q without pre-shaping. AVB Tight Idle-Slope may insert delays between transmissions on egress ports and thus requires more memory than IEEE802.1Q with pre-shaping (+28% on average). TAS, not presented here, has no significant influence on memory usage in our case-study where it has been used for C&C traffic only.

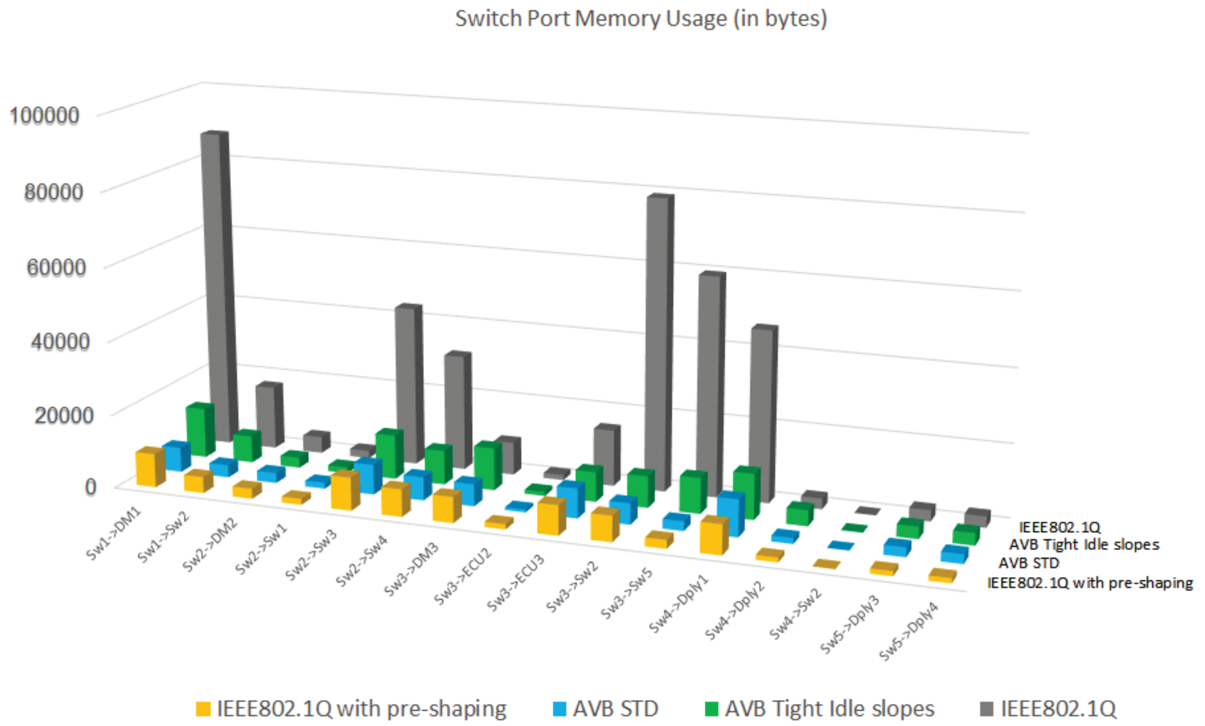

Figure 10: Upper-bounds on the memory usage in the output ports of the switches with IEEE802.1Q (w/wo pre-shaping), AVB standard and AVB Tight-Idle slopes. AVB standard is shown for reference but does not lead to a feasible solution.

4 Conclusion and a look forward

The table below summarizes the results of the experiments presented in this study.

Solutions experimented	Solution meeting all communication requirements?
IEEE802.1Q without pre-shaping	No, throughput requirements for best-effort not met
IEEE802.1Q with pre-shaping	**Yes**
AVB standard classes	No, 10ms deadline for ADAS video not met
AVB custom classes with Tight Idle-Slope algorithm	**Yes**
TSN/TAS under ASAP algorithm with AVB/CBS custom classes – tasks and frame schedules synchronized	**Yes**
TSN/TAS under ASAP algorithm with AVB/CBS custom classes – tasks and messages not synchronized	No, sub-period deadlines for C&C not met

The key takeaways of the experiments in terms of the suitability of the different protocols for our case-study are the following:

- IEEE802.1Q is not suited for bursty traffic/segmented messages when the best-effort traffic at a lower priority level has performance constraints. Pre-shaping the bursty traffic by inserting idle times in transmission leads to important improvements,
- AVB can be an answer to many communication needs but standard classes are not always sufficiently flexible. Their scope of applicability does not cover all automotive needs even for video streams. The use of custom classes allows to get the most out of AVB but tools must be used for configuration and timing verification,
- TSN/TAS is effective at improving the latencies for Command & Control traffic but tools must be used for configuration and timing verification, activities which are both complex under TAS. In addition, the best performance can only be achieved if the scheduling of tasks and message transmissions are coordinated.

Preliminary experiments not shown here suggest that 1Gbit/s Ethernet and frame preemption (IEEE802.3br) may help to simplify protocol stacks for some use-cases.

If three feasible solutions were found for the case-study, it is striking that a fine-grained configuration of the protocol parameters was required to obtain any of these solutions. We are convinced that, given the complexity of the technologies and automotive applications, configuration has become a crucial challenge in the system development. For instance, in this study, the following configuration issues had to be addressed:

- choice of the priorities of the streams,
- allocation of the streams to the AVB classes,
- value of the idle Slopes on each egress port for AVB classes,
- TAS gate schedule table on nodes and switches.

Although out of the scope of this work, in practice the co-scheduling of tasks and messages for TAS nodes and gatewaying strategies are other issues requiring configuration choices. Ultimately, the quality of the configuration will have an impact on the safety, the cost-effectiveness and the capability of the system to evolve over time. This calls for further automation to support protocol selection, their configuration and architectural choices.

Our ongoing work is to develop algorithms that automate the configuration choices we have made manually, and evaluate, based on the same case-study, if the algorithms can be competitive with what engineers can do, or even come up with solutions we have not thought about. From this study, we aim to get insights into what can be automated, what should be best left to the designer and how a tool can interact with the designer to guide him towards an effective solution.

5 References

[Ashjaei17] M. Ashjaei, G. Patti, M. Behnam, T. Nolte, G. Alderisi, L. Lo Bello, “Schedulability analysis of Ethernet Audio Video Bridging networks with scheduled traffic support”, Real-Time Systems, vol. 53, pp526-577, 2017.

[Avnu15] AVnu, “Automotive Ethernet AVB Functional and Interoperability Specification”, R1.4, May 2015.

[BoNaFu12] M. Boyer, N. Navet and M. Fumey, "Experimental assessment of timing verification techniques for AFDX", Embedded Real-Time Software and Systems (ERTS2 2012), Toulouse, France, Feb. 1-3, 2012.

[802.1Qbu] “Standard for Local and Metropolitan Area Networks-Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks - Amendment: Frame Preemption”, 2012.

[CaCuBrLu16] J. Cao, P. Cuijpers, R. Bril, and J. Lukkien, “Independent yet Tight WCRT Analysis for Individual Priority Classes in Ethernet AVB”, Proc. 24th Intl. Conference on Real-Time Networks and Systems (RTNS '16), Brest, Oct. 2016.

[IEEE802.1Q-2014] IEEE, “IEEE Standard for Local and metropolitan area networks--Bridges and Bridged Networks - IEEE Std 802.1Q-2014”, 2014.

[LiGe16] X. Li, L. George, “Deterministic delay analysis of AVB switched Ethernet networks using an extended Trajectory Approach”, Real-Time Systems, Volume 53, Issue 1, pp 121–186, 2016.

[NaSeMi16] N. Navet, J. Seyler, J. Migge, "Timing verification of real-time automotive Ethernet networks: what can we expect from simulation?", Embedded Real-Time Software and Systems (ERTS 2016), Toulouse, France, January 27-29, 2016.

[NaMiViBo18] N. Navet, J. Migge, J. Villanueva, M. Boyer, ONERA “Pre-shaping Bursty Transmissions under IEEE802.1Q as a Simple and Efficient QoS Mechanism”, submitted to the World WCX™: SAE World Congress Experience, Detroit, USA, March 2018.

[Que12] R. Queck, “Analysis of Ethernet AVB for automotive networks using Network Calculus”, IEEE International Conference on Vehicular Electronics and Safety (ICVES), Istanbul, July, 2012.

[RuBo14] J.A. Ruiz De Azua, M. Boyer “Complete modelling of AVB in Network Calculus Framework” Int. Conf. on Real-Time Networks and Systems (RTNS 2014), Versailles, France, October 2014.

[ThErDi15] D. Thiele, R. Ernst, J. Diemer, “Formal worst-case timing analysis of Ethernet TSN's time-aware and peristaltic shapers”, IEEE Vehicular Networking Conference (VNC), Kyoto, December 16-18, 2015.