



Independent WCRT analysis for individual priority classes in Ethernet AVB

Jingyue Cao¹ · Pieter J. L. Cuijpers¹ · Reinder J. Bril¹ · Johan J. Lukkien¹

Published online: 3 October 2018
© The Author(s) 2018

Abstract

In the high-tech and automotive industry, bandwidth considerations and widely accepted standardization are two important reasons why Ethernet is currently being considered as an alternative solution for real-time communication (compared to traditional fieldbusses). Although Ethernet was originally not intended for this purpose, the development of the Ethernet AVB standard enables its use for transporting high-volume data (e.g. from cameras and entertainment applications) with low-latency guarantees. In complex industrial systems, the network is shared by many applications, developed by different parties. To face this complexity, the development of these applications must be kept as independent as possible. In particular, from a network point of view, progress of all communication streams must be guaranteed, and the performance for individual streams should be predictable using only information regarding the stream under study and the general parameters of the communication standard used by the network. Initial methods to guarantee latency for Ethernet AVB networks rely on the traditional busy-period analysis. Typically, these methods are based on knowledge of the inter-arrival patterns of both the stream under study and the interfering streams that also traverse the network. The desired independence is therefore not achieved. In this paper, we present an independent real-time analysis based on so-called *eligible intervals*, which does not rely on any assumptions on interfering priority classes other than those enforced in the Ethernet AVB standard. We prove this analysis is tight in case there is only a single higher-priority stream, and no additional information on interference is known. In case there are multiple higher-priority streams, we give conditions under which the analysis is still tight. Furthermore, we compare the results of our approach to the two most recent busy-period analyses, point out sources of pessimism in these earlier works, and argue that assuming more information on the sources of interference (e.g. a minimal inter-arrival time between interfering frames) has only limited advantages.

✉ Pieter J. L. Cuijpers
P.J.L.Cuijpers@tue.nl

Extended author information available on the last page of the article

Keywords Ethernet AVB · Ethernet TSN · WCRT analysis · Eligible interval · Busy period analysis · Credit based shaping

1 Introduction

The increasing use of, for example, cameras in industrial systems, is resulting in an increasing demand for high-volume data transports. Especially in the high-tech and automotive industry, there is a need for high bandwidth communication combined with the possibility to give real-time guarantees. Ethernet is currently being considered as a viable solution to this problem, as it provides a high bandwidth as part of a widely accepted standard. Originally, Ethernet was not intended for this purpose, but the development of the Ethernet AVB standard [from the audio/video bridging task group IEEE (2005)], enables the use of Ethernet for transporting high-volume data with latency guarantees. This standard relies on traffic shaping techniques, and in particular on the standardized credit-based shaper [see IEEE 802.1Q-2014 IEEE (2014)] to be able to guarantee required latencies and throughput for audio/video traffic, while preventing starvation of best-effort traffic. In 2012, the task group was renamed to time sensitive networking (TSN) task group, with the aim to support time-sensitive traffic by introducing time-triggered transmission on top of other traffic classes. This latter enhancement of scheduled traffic [see IEEE 802.1Qbv IEEE (2016)] is not considered in this paper as we only focus on the credit-based shaping.

One of the problems faced by, for example, the automotive industry, is that there is a growing number of connected components present in modern cars and other vehicles. Those components are of a heterogeneous nature, in the sense that some pose hard real-time requirements on the communication network, while others only require soft real-time guarantees, or have no specified requirements at all. Furthermore, different components are often developed by different manufacturers, making it difficult to guarantee specific behavior in the traffic load the components put on the network.

To face the consequences of this heterogeneity, the transmission of frames in Ethernet AVB is scheduled in a prioritized fashion (see Fig. 1). Frames from several input ports are distributed over (FIFO) output queues depending on their corresponding priority classes. In Ethernet AVB, several priority classes are possible, but in our previous papers Cao et al. (2016a, b) we restricted ourselves to three: H (high priority), M (medium priority), and L (low priority/best-effort traffic). In this paper, we extend the WCRT analysis of the M class in the presence of multiple high priority streams H_1, H_2, \dots, H_p and multiple low priority streams L_1, L_2, \dots, L_q , see right panel in Fig. 1. Within class H and M , a stream of frames is shaped according to the credit-based shaping algorithm, constraining the transmission for those classes to a fraction of the available bandwidth. This prevents starvation of the low priority classes, and in particular enables us to ensure latency guarantees for the M class, which is the main focus of this paper.

Given the mechanisms of the Ethernet AVB standard, and the characteristics of frames and shapers, the worst-case response time of a frame transmission can be bound using formal analysis techniques. Typically, a busy period analysis is performed to estimate the maximum amount of interference from which a transmission may suffer, see

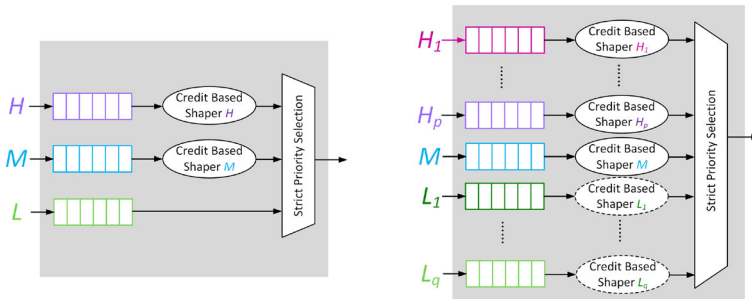


Fig. 1 The output port of an Ethernet AVB switch. Left panel: priority classes H and M undergo credit-based shaping, while class L obtains the remaining bandwidth. Right panel: priority classes H_1, H_2, \dots, H_p and M undergo credit-based shaping. Whether classes L_1, L_2, \dots, L_q are under credit-based shaping (shown as dotted lines) is irrelevant in the WCRT analysis of class M , which will be further explained

Diemer et al. (2012b), Bordoloi et al. (2014), Ashjaei et al. (2017). The problem with using a busy period analysis for the analysis of Ethernet AVB, is that this method was originally developed for non-idling servers. This concern is not explicitly addressed in the works of Diemer et al. (2012b), Axer et al. (2014) and Bordoloi et al. (2014), but it is implicitly solved by simply adding the idling time as additional interference. In Bordoloi et al. (2014), the authors observed that this leads to an overestimation, because some interfering traffic is transmitted during this idling time. This interference should not contribute to the worst-case response time and thus causes pessimism. The authors attempt to compensate for this pessimism, but despite their efforts we show in this paper that their attempt was not entirely successful. Another disadvantage of using busy period analysis is that the analysis requires all detailed traffic models of the interference. This dependence upon traffic model makes the analysis quite complicated, and developers face the problem of an undesired coupling between different applications (often designed by different development teams) when calculating the worst-case performance of their application of interest on the network.

By introducing eligible intervals [see Cao et al. (2016b)] instead of busy periods, we manage to simplify the analysis greatly. As the notion of eligible intervals is tailored to idling servers, we are able to remove the pessimism due to idling present in traditional approaches. Meanwhile, the new analysis is independent of the knowledge of the inter-priority interference other than those assumptions enforced by the Ethernet AVB standard.

The main contribution of this paper is that we extend our previous analysis [see Cao et al. (2016a, b)] to incorporate multiple higher and lower priority streams that individually undergo credit-based shaping. Since the influence of multiple lower priorities is equal to that of a single lower priority, i.e., only one lower priority frame can interfere during an eligible interval, the main challenge lies in the influence of multiple high priorities. For that, we study the variation of the cumulative credit of all higher priority shapers, and find the minimum cumulative credit and the slowest decreasing rate of a set of higher priority shapers to obtain the worst-case relative delay. The difference with our earlier analysis, is that in case of multiple high priority streams it is not always possible to create a worst-case scenario which gives a tight bound.

Still, we are able to give sufficient (and conjectured necessary) conditions under which our analysis yields a tight bound on worst-case relative delay. The complexity of our analysis is factorial in the number of high priority classes. Furthermore, it does not rely on recursions of which the depth is dependent on the chosen system parameters, as it is the case in busy period analysis.

In the remainder of this paper, we briefly introduce the earlier work in Sect. 2, and then describe our system model and our model of behavior of credit-based shaping in Sect. 3. Subsequently, we recapitulate the most important parts of the analysis laid out in Cao et al. (2016b) and add our new analysis of the built-up of cumulative credit of higher priorities under mixed interference in Sect. 4, and present our main analysis of the worst-case response time. Next, in Sect. 5, we discuss the conditions for tightness of our approach. Finally, in Sect. 6, we compare our approach and results to those of Axer et al. (2014) and Bordoloi et al. (2014).

2 Related work

In this section, we first discuss earlier work that focuses on the timing analysis of credit-based shaping in Ethernet AVB. We mainly discuss their methodology, their coverage of single- or multi-hop networks, their assumptions on interfering traffic, and tightness and complexity of their approach. Next, we briefly discuss other works relating to traffic shaping techniques in the broader scope of Ethernet TSN.

Previous studies of Ethernet AVB can be found in e.g. Imtiaz et al. (2009), Diemer et al. (2012a), Diemer et al. (2012b), Reimann et al. (2013), Bordoloi et al. (2014), Axer et al. (2014), Ashjaei et al. (2017) and Li and George (2017). In Imtiaz et al. (2009), a single-hop timing analysis of Ethernet AVB is presented for a scenario in which a single frame is interfered by both high- and low-priority traffic, but no formal study is executed to prove that the resulting formulas match the worst-case in all possible scenarios. In Reimann et al. (2013), a tight timing analysis is proposed without taking a detailed traffic model into account. This approach is based on modular performance analysis [see Wandeler et al. (2006)], which requires a local timing analysis to study the multi-hop network. However, the proposed approach lacks mathematical proofs and the claim of tightness is not validated.

In Diemer et al. (2012a, b) an extensive formal analysis of Ethernet AVB is given based on busy period analysis (Davis et al. 2007; Bril et al. 2009; Lehoczky 1990). It uses the compositional performance analysis approach (Henia et al. 2005) to take an output event stream of one component and turn it into an input event stream of a connected component for multi-hop network analysis. However, as the busy period analysis is not designed for idling servers, pessimism remains a concern in these works. Axer et al. (2014) and Bordoloi et al. (2014) independently improve on the work of Diemer et al. (2012b) to reduce the existing pessimism. Bordoloi et al. (2014) attempts to address two sources of pessimism: one source is that the high priority traffic is also shaped, and the other source is that the high priority traffic which transmits during idling time should not be counted as interference. This improvement is limited to a single-hop setting, while Axer et al. (2014) attempts to address part of that same improvement, but considers a multi-hop analysis.

Ashjaei et al. (2017) presents a schedulability analysis in Ethernet AVB, which extends the work of Bordoloi et al. (2014) with support for scheduled traffic and further proposes a methodology for allocating network bandwidth to credit-based shapers. The improvement done by Bordoloi et al. (2014) is not considered in Ashjaei et al. (2017), as the improved solution is too complicated to determine the network bandwidth reservation. Li and George (2017) present a schedulability analysis to compute the worst-case delay of messages from classes A and B using the trajectory approach (Martin and Minet 2006) in a multi-hop network. This work partially uses busy period analysis to limit the search space of time within which the worst-case delay upper bound can be found and it provides a tighter analysis compared to Diemer et al. (2012b).

Besides the pessimism, another disadvantage of using busy period analysis is that a detailed traffic model is required, e.g., a periodic/sporadic model in Bordoloi et al. (2014) and Li and George (2017) or arrival curves in Diemer et al. (2012b) and Axer et al. (2014). The dependence on a detailed traffic model makes the timing analysis quite complicated in terms of algorithmic complexity, and meanwhile not robust against traffic flow changes.

In Cao et al. (2016b), we have introduced a relative worst-case response time analysis that makes use of eligible intervals. In the follow-up paper Cao et al. (2016a), we have provided a formal analysis of the medium priority traffic with mixed interference, consisting of a single shaped high-priority stream and an unshaped low-priority stream. By further studying the maximum amount of credit that can build up, we produce a bound on the worst-case relative delay and formally prove that this bound is tight. Furthermore, this proposed analysis is independent of the traffic model of the inter-priority interference. As a result, the complexity of the analysis is considerably low compared to the previous analyses. We presume the linear complexity (for a single high priority) allows the analysis to scale in a multi-hop network, however, the contribution is currently limited to single-hop.

Since 2012, the work on credit-based shaping has become part of a larger standardization initiative in the Ethernet TSN task group. In Ethernet TSN, also other shaping technologies are being developed [see e.g. Thangamuthu et al. (2015)]. Several works have presented timing analysis for these new shapers. For instance, Thiele et al. (2015) has proposed an analysis technique based on busy period analysis for the time-aware shaper and peristaltic shaper, and Thiele and Ernst (2016) has addressed the burst-limiting shaper also using busy period analysis. There are also studies addressing the schedulability analysis of credit-based shaping with support for scheduled traffic, e.g. Maxim and Song (2017) and Ashjaei et al. (2017). The new shapers and the combination of shapers in Ethernet TSN are not covered in this paper, as we only focus on credit-based shaping. However, as part of future work it may be interesting to see if the eligible interval method is applicable to those shapers as well.

3 System model

In this section, we first sketch the mechanics of credit-based shaping, relevant for our analysis. Subsequently, we introduce some notation to capture these mechanics.

Finally, we give mathematical relationships that characterize the mechanics of credit-based shaping formally. For this, we adopt an axiomatic approach. This means that, instead of describing exactly how a shaper behaves by presenting a construction of its executions, we only specify relations between events and variables that an implemented Ethernet AVB switch should respect. Subsequently, we use these relations to derive properties of the behavior of a shaper. Naturally, in some places we will give explicit constructions of executions, either as illustrations, or as a witness that a derived bound can be achieved in tightness proofs. As a side-effect, these constructions also serve to convince the reader that the axioms we posed are not self-contradictory.

3.1 Mechanics of credit-based shaping

We are modeling the behavior of a single switch that is part of an Ethernet AVB network. The goal of this switch is to pass on frames from its inputs to its outputs according to some routing table. However, we are interested only in the time it takes to perform this task, and not in the actual routing of frames. In order to be able to calculate this, it is important to know how a switch schedules its transmissions.

As a basis, Ethernet AVB adopts non-preemptive strict priority scheduling. Frames are gathered in one queue for each priority class, and a frame can start its transmission only if there is no higher priority queue that can start (strict priority), if there are no earlier frames of the same priority in the queue (FIFO), and if there is currently no transmission going on (non-preemptive), see Fig. 1.

On top of this, Ethernet AVB adds a credit-based shaping mechanism for the priority class to limit traffic bursts. In order to start the transmission of a frame, the priority class can not have negative *credit*. Whenever credit is above or equal to 0, a frame can start transmission, assuming the other rules also allow this. If credit is lower than 0, frames from lower priority classes are allowed to transmit. Credit for a priority class X starts at 0, and while a frame is in transmission it drops at a rate of α_X^- . If credit is negative, no further frames of that class can be transmitted and credit rises with a rate of α_X^+ . Also, if class X has pending frames waiting but cannot transmit, credit rises at a rate of α_X^+ . Positive credit is reset to 0 if the queue of class X does not contain any frames anymore. One example of credit-based shaping is illustrated in Fig. 2.

In the standard, $0 < \alpha_X^+ \leq BW$ is set to correspond to the desired bandwidth reservation for class X , assuming a total bandwidth of BW . Subsequently, $\alpha_X^- = BW - \alpha_X^+$ is set to the remaining bandwidth. We observe that the maximum utilisation for class X is given by $\frac{\alpha_X^+}{BW}$, see also Fig. 2. When multiple streams are of the same priority class, they share the bandwidth of that class.

When designing a system it is reasonable to expect that the total reservation does not exceed the total available bandwidth, resulting in the requirement:

$$\sum_{X \in \mathbb{P}} \alpha_X^+ \leq BW, \quad (1)$$

where \mathbb{P} denotes the set of all priority classes. Although this requirement is strictly speaking not enforced by the standard, it seems a reasonable design choice in general,

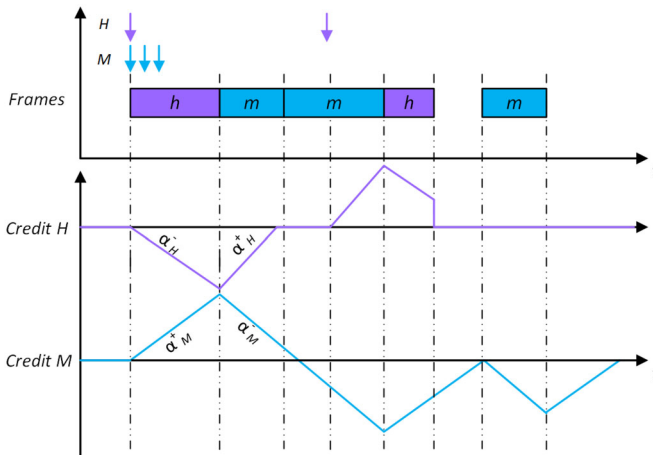


Fig. 2 An example of credit-based shaping with class H and class M traffic. Arrows here indicate frame arrivals while rectangles indicate the transmission of frames. The transmission follows two rules: H is prioritized over M , and there must be non-negative credit at the start of a transmission. The rising slope of credit, α^+ , is set to the reserved bandwidth and the falling slope is set to the remaining bandwidth. For instance, if α_H^+ is set to 60 Mbps while the total available bandwidth is 100 Mbps, then α_H^- is set to 40 Mbps, which leads to a maximum utilization of H of 60%

and we do adopt this assumption later in our analysis. The only exception to this rule, is the lowest priority class, which in practice is not shaped at all, but can be treated as if it is shaped with a reservation equal to the bandwidth ($\alpha_X^+ = BW$).

Another parameter set in the standard, is the maximum size C_X^{\max} that a frame of a given priority class X may have. Size, in this case, refers to the number of bits that need to be transmitted in the standard, but for ease of presentation is converted to the maximum amount of time a transmission takes in this paper.

3.2 Notation

Priority As indicated above, we use the capital letter X , but also Y, M, H , and L , to denote priority classes, and we write \mathbb{P} for the set of all priority classes. This set is considered to be totally ordered, so we may write $X, Y \in \mathbb{P}$ and $X < Y$ to indicate that X has a lower priority than Y . Furthermore, in the remainder we sometimes write $\mathbb{X} \subseteq \mathbb{P}$ if we want to consider a subset of priority classes \mathbb{X} , and write $\alpha_{\mathbb{X}}^+ = \sum_{X \in \mathbb{X}} \alpha_X^+$

and $\alpha_{\mathbb{X}}^- = BW - \alpha_{\mathbb{X}}^+$ to indicate the cumulative reservation of such a set and its remainder.

Frames, streams and sources To indicate that a frame x is of priority class X we write $x \in X$. The set of all frames is denoted by \mathbb{F} , so that formally we can regard a priority class $X \subseteq \mathbb{F}$ as a set of frames, as well as an element of \mathbb{P} . However, when discussing the behavior of credit based shaping, it is more natural to speak of a *stream* X when considering the flow of frames from input to transmission, and to speak of a *priority class* X when considering scheduling decisions on queues. A *source* $\tau \subseteq X$

is a specific set of frames under investigation, taken from the same priority class, e.g., a set of periodically-generated frames.

Dynamics and credit As mentioned before, we follow an axiomatic approach to formally model the dynamics of credit based shaping. In this approach, we consider executions u , and give properties under which u is considered a valid execution of the system, denoted $u \in \mathbb{U}$. These properties are in terms of events and variables defined by the standard. For example, we write $CR_X^u(t)$ to denote the credit of priority class X at time $t \in \mathbb{R}^+$ during an execution u . As an example of a property, we have that $CR_X^u(0) = 0$ for all $u \in \mathbb{U}$, saying that at the start of execution (time 0) the credit of every shaper is 0.

More on executions Typically, one may consider an execution u to be an ordered set of observations, like events or valuations of variables. But for most of our analysis, it is not important how executions are exactly defined, as long as we can define the necessary functions that keep track of variables and events in the system. The standard does not prescribe the exact implementation of the traffic shapers, it merely gives a specification of possible implementations. Therefore, we strive not to define here what an execution is exactly. It is sufficient to know which axioms need to hold for an execution. This keeps our analysis independent of the chosen implementation. Sometimes we do give examples of valid executions by specifying which events and variable valuations are observed in which order. The purpose of these is either to serve as illustrations, or as witnesses to prove that an execution with certain properties actually exists.

Arrival time The behavior of a switch is determined by the arrival time of frames during an execution, and by the order of arrival in case multiple frames arrive at the same point in time. We assume that for any given set of frames we are able to distinguish the first frame that arrived. Technically, let $\mathbb{F}^u \subseteq \mathbb{F}$ denote those frames that arrive during an execution u , and assume (reasonably) that \mathbb{F}^u is a countable set. We denote the arrival time of an arbitrary frame $x \in \mathbb{F}^u$ by $a^u(x) \in \mathbb{R}^+$. Furthermore, because it may be the case that multiple frames arrive at the same point in time, we sometimes write $x <_u x'$ to emphasize the order in which frames arrive. In particular, for some $x, x' \in \mathbb{F}^u$ we may use $x <_u x' \wedge a^u(x) = a^u(x')$ to denote that x arrived before x' at the input queue, but their arrival times cannot be distinguished.

Start, finish, and worst-case response time For a given frame $x \in \tau$, the response time in an execution u is defined as the difference between finish time $f^u(x)$ of that frame and its arrival time $a^u(x)$. The worst-case response time of a frame x given a set of possible executions \mathbb{U} is defined as:

$$WR_{\mathbb{U}}(x) = \sup_{u \in \mathbb{U}} f^u(x) - a^u(x). \quad (2)$$

The worst-case response time of a source τ can be further defined as:

$$WR_{\mathbb{U}}(\tau) = \sup_{x \in \tau} WR_{\mathbb{U}}(x). \quad (3)$$

In this paper, we present a relative response time analysis, meaning that we compare an execution u and a related un-interfered execution u_0 without interference, and then calculate the worst-case relative delay.

Definition 1 (*Un-interfered execution*) Given a priority class X and an execution u , we define the *un-interfered execution of X* to be an execution u_0 (dependent on u) in which only frames of X arrive, and arrive at the same time as in u . Furthermore, they have the same transmission time. So $\mathbb{F}^{u_0} = \mathbb{F}^u \cap X$ and for all $x \in \mathbb{F}^{u_0}$ we have $a^{u_0}(x) = a^u(x)$ and $C^{u_0}(x) = C^u(x)$. Assuming the scheduling of frames is deterministic, this u_0 is uniquely defined.

Given non-preemptive transmissions in the switch, the finish time $f^u(x)$ is determined by the start-time $s^u(x)$, hence we have $f^u(x) = s^u(x) + C^u(x)$. We can then derive a bound on the response time of a frame x in any execution u from that of u_0 by studying the delay in start-times as shown below:

$$\begin{aligned}
 WR_{\mathbb{U}}(x) &= \sup_{u \in \mathbb{U}} (f^u(x) - a^u(x)) \\
 &= \sup_{u \in \mathbb{U}} (s^u(x) + C^u(x) - a^u(x)) \\
 &= \sup_{u \in \mathbb{U}} (s^u(x) + C^{u_0}(x) - a^{u_0}(x)) \\
 &= \sup_{u \in \mathbb{U}} (s^{u_0}(x) + C^{u_0}(x) - a^{u_0}(x) + (s^u(x) - s^{u_0}(x))) \\
 &\leq \sup_{u \in \mathbb{U}} (s^{u_0}(x) + C^{u_0}(x) - a^{u_0}(x)) + \sup_{u \in \mathbb{U}} (s^u(x) - s^{u_0}(x)) \\
 &= WR_{\mathbb{U}_0}(x) + \sup_{u \in \mathbb{U}} (s^u(x) - s^{u_0}(x)),
 \end{aligned} \tag{4}$$

where $WR_{\mathbb{U}_0}(x)$ denotes the worst-case response-time of x in the set $\mathbb{U}_0 = \{u_0 \mid u \in \mathbb{U}\}$ of all un-interfered executions, while the term $\sup_{u \in \mathbb{U}} (s^u(x) - s^{u_0}(x))$ is called the *worst-case relative delay* of a frame x . In Sect. 4, we seek an upper bound to this relative delay, given that $x \in \tau \subseteq X$ is a frame from a given priority class X and source τ . It is assumed that the worst-case response time of a frame x in the un-interfered execution $WR_{\mathbb{U}_0}(x)$ is known.

In Sects. 4 and 5, we analyze the relative delay without restricting a^u to any particular arrival pattern. In particular, we do not assume the usual periodic or sporadic arrival. In Sect. 6 we compare our results to other work which does depend on periodic/sporadic arrivals, and calculate $WR_{\mathbb{U}_0}(x)$ based on this assumption. Before going into the analysis, however, we first develop the relations we need to derive this upper bound by further formalizing the mechanics of credit based shaping.

3.3 Formalization of shaping rules

In this section, we rephrase the earlier descriptions of credit based shaping in terms of mathematical relationships between arrival times, order of arrival, credit, transmission times, start times and finish times. In the subsequent sections, we use these

relationships to bound the worst-case response time of frames in credit based shaping. As mentioned before, we adopt an axiomatic approach to defining the behavior of credit-based shaping. The predicates in this section define which executions u are valid elements of the set of all executions \mathbb{U} of a credit-based shaper. Next, in the analysis, we can then use these predicates to prove properties of these executions.

Non-preemptive Let us start by recalling a property mentioned previously. The fact that transmissions are non-preemptive means that for all $x \in \mathbb{F}$ and executions $u \in \mathbb{U}$ we find:

$$f^u(x) = s^u(x) + C^u(x). \quad (5)$$

Furthermore, as we have seen before, the fact that the standard bounds the maximum size of frames $x \in X$, and the fact that transmissions require a positive amount of time, gives us the following bounds on the transmission time:

$$0 < C^u(x) \leq C_X^{\max}. \quad (6)$$

Single transmission Since we assume a single transmission at a time, two frames cannot start their transmission at the same time and any frame that starts earlier should finish its transmission before the next frame starts. For all $x, x' \in \mathbb{F}$ and $u \in \mathbb{U}$:

$$x \neq x' \Rightarrow s^u(x) \neq s^u(x'), \quad (7)$$

and

$$s^u(x) < s^u(x') \Rightarrow f^u(x) \leq s^u(x'). \quad (8)$$

Non-negative credit Next, we study the start time of frames in more detail. In particular, we have the rule that in order to transmit a frame, it must first have arrived and its priority class must not have negative credit. So for any priority class $X \in \mathbb{P}$ and frame $x \in X$ we find in any execution $u \in \mathbb{U}$:

$$a^u(x) \leq s^u(x) \wedge CR_X^u(s^u(x)) \geq 0. \quad (9)$$

FIFO Furthermore, the fact that frames of the same class are queued and transmitted in the order in which they arrive (First-In First-Out), gives us for all executions $u \in \mathbb{U}$, all $X \in \mathbb{P}$, and all $x_1, x_2 \in X$:

$$x_1 <_u x_2 \Rightarrow s^u(x_1) < s^u(x_2). \quad (10)$$

Strict priority Similarly, the general notion that frames of higher priority take precedence if credit allows can be formalized by stating that for all $u \in \mathbb{U}$, all $X, Y \in \mathbb{P}$ with $X < Y$, and all $x \in X$ and $y \in Y$ we have

$$(a^u(y) \leq s^u(x) \wedge CR_Y^u(s^u(x)) \geq 0) \Rightarrow s^u(y) \leq s^u(x). \quad (11)$$

Note that Eq. (11) captures the notion of priority even in the case of multiple simultaneous transmissions. Combining this with Eq. (7) and Eq. (8) we get:

$$(a^u(y) \leq s^u(x) \wedge CR_Y^u(s^u(x)) \geq 0) \Rightarrow f^u(y) \leq s^u(x). \tag{12}$$

From Eq. (12) it follows that, if a higher priority has pending load and credit, a lower priority cannot start its transmission. If it would start, the predicate would be violated.

Eager scheduling If we combine the previous properties (non-preemptive, single transmission, non-negative credit, FIFO and strict priority) with the assumption that frames are scheduled as soon as credit allows, we obtain the infimum over the above properties as start-time of a frame. For all $u \in \mathbb{U}$, $X \in \mathbb{P}$ and $x \in X$ we have:

$$s^u(x) = \inf \left\{ t \left| \begin{array}{l} \forall z \in \mathbb{P} s^u(z) < t \Rightarrow f^u(z) \leq t \\ a^u(x) \leq t \wedge CR_X^u(t) \geq 0 \\ \forall x' \in X x' <_u x \Rightarrow s^u(x') < t \\ \forall y \in Y, Y > X (CR_Y^u(t) \geq 0 \wedge a^u(y) \leq t) \Rightarrow (f^u(y) \leq t) \end{array} \right. \right\}. \tag{13}$$

Credit at start The only unknown in the equation above, is the credit, for which we already mentioned that it starts at 0 in every execution $u \in \mathbb{U}$ and for any priority class $X \in \mathbb{P}$:

$$CR_X^u(0) = 0. \tag{14}$$

Credit during transmission While a frame of priority class $X \in \mathbb{P}$ is in transmission, credit drops at a rate of α_X^- . Therefore, for any two points in time $t \leq t'$ for which there exists a frame $m \in X$ such that $s^u(m) \leq t$ and $t' \leq f^u(m)$ we find:

$$CR_X^u(t') = CR_X^u(t) - \alpha_X^- \cdot (t' - t). \tag{15}$$

Credit recovery While credit of priority class $X \in \mathbb{P}$ is negative and X is not transmitting, credit rises with a rate of α_X^+ . Therefore, for any two points $t \leq t'$ with $t' \leq t - \frac{CR_X^u(t)}{\alpha_X^+}$ and X not transmitting at t (i.e. $\forall x \in X f^u(x) \leq t \vee s^u(x) > t$) we find:

$$CR_X^u(t') = CR_X^u(t) + \alpha_X^+ \cdot (t' - t). \tag{16}$$

Pending load Similarly, if there are frames in the queue of priority class $X \in \mathbb{P}$, but this class cannot transmit, this also leads to a credit rise with a rate of α_X^+ . We say that a priority class X has *pending load* at time $t \in \mathbb{R}^+$, denoted $Pending_X(t)$, if there exists a frame $m \in X$ such that $a^u(m) \leq t$ and $f^u(m) > t$. Furthermore, we say that a priority class Y is *transmitting* at time $t \in \mathbb{R}^+$ if there exists a frame $n \in Y$ such that $s^u(n) \leq t$ and $f^u(n) > t$. Now, for any two points $t \leq t''$ such that X has pending load at t but some other class Y is transmitting at every point t' with $t \leq t' < t''$, we find:

$$CR_X^u(t'') = CR_X^u(t) + \alpha_X^+ \cdot (t'' - t). \tag{17}$$

Credit reset Finally, we know that credit of priority class $X \in \mathbb{P}$ is reset as soon as there is no pending load. Therefore, at any time $t \in \mathbb{R}^+$ we find:

$$\left(\neg \text{Pending}_X(t) \wedge \lim_{t' \uparrow t} CR_X^u(t') \geq 0 \right) \Rightarrow CR_X^u(t) = 0. \quad (18)$$

Definition 2 (*Credit based shaping*) An execution $u \in \mathbb{U}$ is *credit based shaped* if its arrival, start, and finishing times a^u , s^u and f^u satisfy Eq. (5) to Eq. (18).

Eligible interval The relations above completely characterize the behavior of credit based shaping. However, during the analysis, we have found that it is convenient to study so-called *eligible intervals* [see also Cao et al. (2016b)].

Definition 3 (*Eligible interval*) A priority class X is *eligible* for transmission at a time t if it is either transmitting, or it has both credit and pending load. An eligible interval of X is a maximal interval during which X is eligible.

Typically, such intervals are left-closed and right-open. They start with an arrival or with credit becoming zero, and end at the end of a transmission. In other words, an interval $E = [E_s, E_e)$ is an eligible interval of $X \in \mathbb{P}$ during execution $u \in \mathbb{U}$ iff:

1. $CR_X^u(E_s) = 0$ and $\text{Pending}_X(E_s)$;
2. $CR_X^u(E_s - \epsilon) < 0$ or $\neg \text{Pending}_X(E_s - \epsilon)$, for a vanishingly small $\epsilon > 0$;
3. For some frame $x \in X$ we have $f^u(x) = E_e$;
4. $CR_X^u(E_e) < 0$ or $\neg \text{Pending}_X(E_e)$;
5. E_e is the smallest time strictly larger than E_s with these properties.

Note, that we will leave the dependency of E on u and X implicit in our notation, since it is always clear from the context which execution and priority is intended. Given this formalization, we are ready to start our analysis.

4 Analysis

Our analysis largely follows the outline set up in Cao et al. (2016b). We focus on the behavior at the output port of an Ethernet AVB switch, as depicted in Fig. 1. This means we assume that incoming frames have already been routed to the proper output port, and that the switch can freely transmit at its output. At the output port, frames are queued according to their class, resulting in several streams of frames, one for each class. We keep track of when frames of a certain stream arrive, and when they are transmitted, and we calculate how long a frame may remain in the output buffer. In other words, we are interested in the maximum amount of time between a frame entering the buffer and finishing transmission.

We first recap, in Sections 4.1, 4.2 and 4.3, the most important observations from Cao et al. (2016b), and analyze a few important cases in sequence. We start by assuming a single input stream of a single priority class X for which we examine the effect of credit based shaping on the output stream by calculating the start times as the result of shaping. This results in an ordinary FIFO schedule, scaled to the reserved

bandwidth. Next, we recall the analysis using so-called *eligible intervals* in which a stream has pending load available (i.e. there are frames to be sent) and also has non-negative credit (i.e. the frames can be sent unless the output port is otherwise occupied) or is transmitting. The main theorem on eligible intervals is that, regardless of the interference, the start time of an eligible interval will always be smaller than or equal to the start time of its first frame in an un-interfered schedule. As a consequence, we can use eligible intervals to calculate a worst-case response time *relative* to an un-interfered schedule. This we do by calculating an upper bound on the start time of each eligible interval. We show this delay is proportional to the credit at the start of transmission of the first frame in the eligible interval, and as a consequence, we conclude that the maximum attainable credit bounds the maximum delay experienced by any frame compared to the un-interfered FIFO schedule.

Next, in Sect. 4.4, we extend the analysis of Cao et al. (2016a) to multiple interfering streams. We bound the maximum attainable credit in case of multiple higher and lower priority shapers, and find a relatively simple expression describing the relationship between the maximum frame sizes in the interfering traffic and the maximum attainable credit. This leads to a similarly simple expression for the worst-case time a frame may remain in its buffer. It is worth emphasizing, that in our analysis, streams are not restricted in the timing of the arrival of frames. A stream X or interfering stream Y may be periodic, sporadic, bursty, or have any other arrival pattern. In particular, it may represent the combination of several sources of traffic. We only use knowledge of the maximum size of frames that are being transmitted. In Sect. 5, we study the tightness of our approach under this limited set of assumptions on interference, and give conditions under which tightness can be guaranteed. In Sect. 6, we compare our estimates to earlier studies on worst-case response times for Ethernet AVB, paying special attention to the situations where our estimates are not tight, or where additional information on interference might give better results.

4.1 Shaping a single un-interfered stream

We start our analysis by considering a single stream X , and at this point assume there is no interference from other streams yet. In other words, for the time being we only consider those executions $u \in \mathbb{U}_0$ for which $\mathbb{F}^u \subseteq X$ and hence $u = u_0$. To emphasize this, we will simply write u_0 for the execution under consideration in the remainder of this section. In the following sections, we will consider how to analyze more complex executions in which there is interference from higher and lower priority streams.

The first frame $x_0 \in \mathbb{F}^{u_0}$ in this stream arrives at $a(x_0) \geq 0$ and the credit is initially 0 [Eq. (14)]. A frame can only start transmission when the credit is non-negative [Eq. (9)]. Since positive credit is possible only when there is interference [only Eqs. (16) and (17) in Sect. 3.3 allow credit to rise], frames in an un-interfered stream start their transmission with a credit of 0. During the transmission, the credit decreases at a rate of α_X^- . The credit has to recover to zero before any next frame can be transmitted. The recovery time is the product of the frame duration and the ratio α_X^-/α_X^+ (see Fig. 3). This gives us the following recurrence for start times.

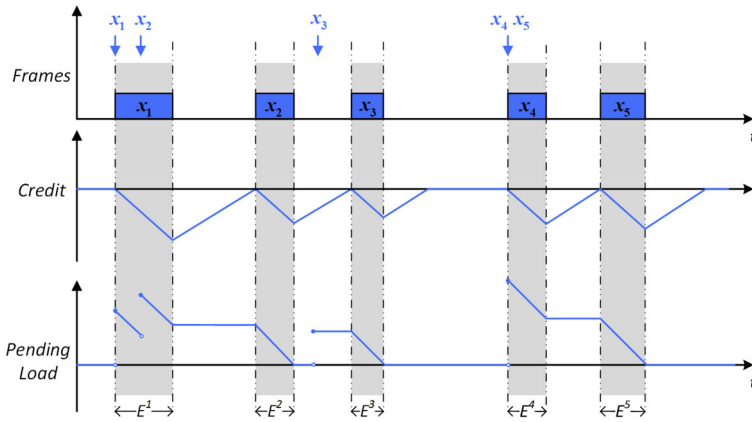


Fig. 3 The transmission of a single stream X undergoing credit-based shaping. Arrows indicate the frame arrivals, and the transmission under credit based shaping is shown along with the evolution of the credit and pending load. The amount of pending load of a class is the total remaining transmission time of pending frames of that class. Each eligible interval is labeled with E , within which there is only one frame

Property 1 Given a single shaped stream X in an un-interfered execution u_0 , and considering the frame arrivals as a sequence x_i , then the start times are determined by

$$s^{u_0}(x_0) = a^{u_0}(x_0),$$

$$s^{u_0}(x_{i+1}) = \left(s^{u_0}(x_i) + C^{u_0}(x_i) \cdot \left(1 + \frac{\alpha_X^-}{\alpha_X^+} \right) \right) \max a^{u_0}(x_{i+1}). \tag{19}$$

Property 1 shows the shaping effect on a single stream to be the same as that of a basic FIFO schedule with enlarged execution times by a factor $(1 + \alpha_X^-/\alpha_X^+)$.

For a single un-interfered stream X , the intervals $E^i = [s^{u_0}(x_i), s^{u_0}(x_i) + C^{u_0}(x_i))$ represent a series of eligible intervals of X . This is illustrated in Fig. 3. In all our figures that display frame transmission, eligible intervals have been marked in gray. Observe, that transmission of frames in X always takes place during some eligible interval, meaning that eligible intervals are always separated by intervals without transmission of X frames. Note also, that there is no idle time for the switch as a whole within an eligible interval, i.e. no time at which no transmissions take place at all. Of course, these transmissions may be from other streams than X . This un-interrupted activity, analogous to the notion of busy-periods in the theory of non-idling servers, makes the behavior within an eligible interval easier to analyse.

Note, that when studying an un-interfered stream, we find that the shape of eligible intervals is rather simple. There is only one frame in each interval E , since the credit becomes negative when a frame is sent and no frame can transmit before the credit recovers to zero. Later, we see that in cases of interference, multiple frames may transmit in an eligible interval as well.

Figure 3 also shows that if arrival times are late there can be some additional slack between two eligible intervals. However, in the un-interfered execution, the minimum distance between $s^{u_0}(x_i)$ and $s^{u_0}(x_{i+1})$ is $C^{u_0}(x_i)(1 + \alpha_X^-/\alpha_X^+)$. This is captured in

the following corollary, which we use in the theorem to determine the start time of an eligible interval in an interfered schedule.

Corollary 1 *Given a single shaped stream X in an un-interfered execution u_0 , and considering the arrivals of frames as a sequence x_i , we find for all $x_0 \dots x_{i+k} \in X$:*

$$s^{u_0}(x_{i+k}) \geq s^{u_0}(x_i) + \sum_{j=i}^{i+k-1} C^{u_0}(x_j) \cdot \left(1 + \frac{\alpha_X^-}{\alpha_X^+}\right)$$

Our remaining analysis focusses on determining how interference may induce a deviation from the FIFO schedule described by Property 1.

4.2 The start of an eligible interval

In this section we study the stream of an arbitrary priority class X during an arbitrary execution u , which may encounter interference from other low- or high-priority streams. I.e. \mathbb{F}^u may also contain frames that are not from X . We show that the start of an eligible interval of X under such interference will always be sooner than or equal to the un-interfered start-time of the first transmitted frame of X of that eligible interval. For this analysis, we can simply consider interference as the union of all higher and all lower priority frames. Later, the individual higher-priority shapers will be studied for their individual effects.

Due to interference, transmissions of frames in the stream X may cluster, and multiple frames can be transmitted in a single eligible interval. But even though the eligible intervals of the interfered execution are different compared to the un-interfered stream, we manage to relate the start-times of the eligible intervals in the interfered execution u to the start-times of frames in the un-interfered execution u_0 .

Obviously, for u_0 the analysis in the previous subsection may be used, which gives us the start-times $s^{u_0}(x)$ of the un-interfered execution.

Now, consider a single eligible interval E of X in the execution u , and let $x_e \in \mathbb{F}^u \cap X$ be the first frame from X transmitted during E , i.e. the first frame such that $s^u(x_e) \in E$. We may then recall the first of the two main theorems from Cao et al. (2016b), which states that the start of an eligible interval in u always lies before or at the transmission of the first frame x_e in u_0 : i.e. $E_s \leq s^{u_0}(x_e)$, see Fig. 4. This allows us further on to relate the delay experienced by any frame back to the un-interfered schedule by just studying the interference within an eligible interval.

Theorem 1 *Let x_e be the first frame of X transmitted in an eligible interval $E = [E_s, E_e]$ of X in execution u , and let u_0 denote the un-interfered execution of X associated with u , then $E_s \leq s^{u_0}(x_e)$.*

Proof The proof proceeds by induction on the sequence of eligible intervals which we denote by E^j for $j \geq 1$.

Clearly, the first eligible interval, E^1 , starts as soon as there is pending load, i.e. at $E_s^1 = a^u(x_1)$ with x_1 the first frame of X in E^1 . Incidentally, this is also the start time of x_1 in the un-interfered schedule. The first frame can start as soon as it arrives,

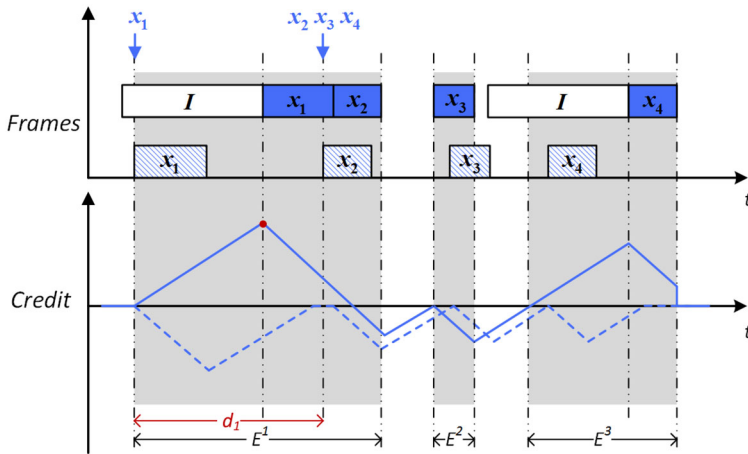


Fig. 4 The transmission of a stream X with the presence of inter-priority interference. Rectangles labeled with I represent the transmission of interference. Solid rectangles represent the transmission of stream X , while shaded ones represent the interference-free transmission of X . Similarly, solid lines represent the credit evolution of X under interference, while dashed lines represent the credit evolution without interference. The eligible interval E^1 starts at the start-time of its first frame x_1 in the un-interfered execution u_0 , i.e. $E_s^1 = s^{u_0}(x_1)$. For E^2 and E^3 , the eligible interval starts before the start-time of its first frame in the un-interfered execution, i.e. $E_s^2 < s^{u_0}(x_3)$ and $E_s^3 < s^{u_0}(x_4)$, see Theorem 1. The relative delay of x_1 , labeled with d_1 , is proportional to the maximum credit achieved, marked using a dot: $s^u(x_1) - s^{u_0}(x_1) \leq \frac{CR_X^u(s^u(x_1))}{\alpha_X^+}$, see Theorem 2

i.e., $a^{u_0}(x_1) = s^{u_0}(x_1)$. By Definition 1, $a^u(x_1) = a^{u_0}(x_1)$, so we have $E_s^1 = s^{u_0}(x_1)$, thus proving the base case.

Next, consider that x_k is the first frame of X in E^j and that frames x_k, \dots, x_l of X are transmitted during E^j and assume that this theorem holds for E^j , i.e., $E_s^j \leq s^{u_0}(x_k)$. Now we need to show $E_s^{j+1} \leq s^{u_0}(x_{l+1})$, with x_{l+1} the first frame of X in E^{j+1} . We know by definition of eligible interval that at the start of the interval E_s^j , the credit equals $CR_X^u(E_s^j) = 0$. Because by definition there is load pending during E^j , credit will fall during the transmission of those frames, and any remaining time in E^j credit will rise, see Fig. 5. At the end of the interval E_e^j we therefore find:

$$\begin{aligned}
 & CR_X^u(E_e^j) \\
 &= \alpha_X^+ \cdot \left(E_e^j - E_s^j - \sum_{i=k}^l C^u(x_i) \right) - \alpha_X^- \cdot \sum_{i=k}^l C^u(x_i) .
 \end{aligned}$$

Now, in case this value is positive, it is reset to zero because there is no pending load at the end of an eligible interval. In that case the start of E^{j+1} lies at the arrival of frame x_{l+1} , i.e. $E_s^{j+1} = a^u(x_{l+1})$. Furthermore, we know that $s^{u_0}(x_{l+1}) \geq a^{u_0}(x_{l+1}) = a^u(x_{l+1}) = E_s^{j+1}$, which concludes this case.

In case the value $CR_X^u(E_e^j)$ is negative, we know that E^{j+1} can only start after credit has recovered to zero, which happens at:

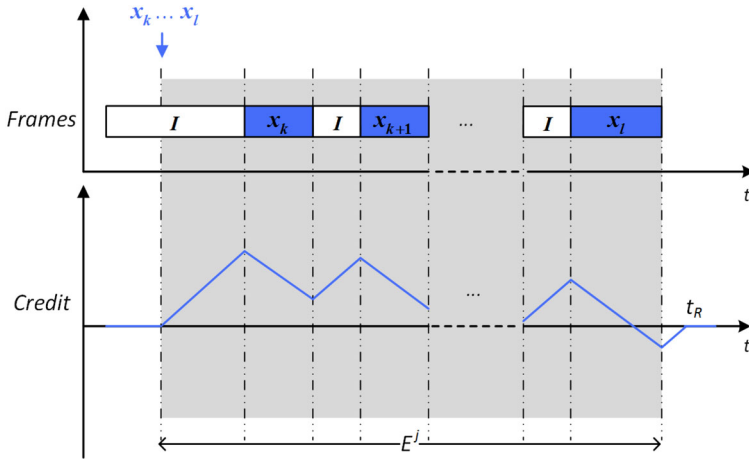


Fig. 5 The transmission of a stream X with the presence of inter-priority interference in an eligible interval E^j . The rectangles labeled with I represent the transmission of interference, while the remaining rectangles represents the transmission of $x_k \dots x_l$. The credit of X decreases while frames from X are transmitting while the credit increases for the remaining time in E . In this plot, the eligible interval ends due to negative credit. This credit will recover to zero at t_R

$$t_R = E_e^j + CR_X^u(E_e^j)/\alpha_X^+ = E_s^j + \sum_{i=k}^l C^u(x_i) \cdot \left(1 + \frac{\alpha_X^-}{\alpha_X^+}\right).$$

Note, that this value is independent of the interference.

Using the induction hypothesis $E_s^j \leq s^{u0}(x_k)$ and the definition of un-interfered execution (Definition 1), we can now calculate:

$$\begin{aligned} E_s^{j+1} &= t_R \max a^u(x_{l+1}) \\ &= \left(E_s^j + \sum_{i=k}^l C^u(x_i) \cdot \left(1 + \frac{\alpha_X^-}{\alpha_X^+}\right)\right) \max a^u(x_{l+1}) \\ &\leq \left(s^{u0}(x_k) + \sum_{i=k}^l C^u(x_i) \cdot \left(1 + \frac{\alpha_X^-}{\alpha_X^+}\right)\right) \max a^u(x_{l+1}) \\ &= \left(s^{u0}(x_k) + \sum_{i=k}^l C^{u0}(x_i) \cdot \left(1 + \frac{\alpha_X^-}{\alpha_X^+}\right)\right) \max a^{u0}(x_{l+1}) \end{aligned}$$

and further using Corollary 1

$$E_s^{j+1} \leq s^{u0}(x_{l+1}) \max a^{u0}(x_{l+1}) \leq s^{u0}(x_{l+1}),$$

which concludes the proof. □

4.3 Credit represents delay

Now that we have an upper bound on the start time of each eligible interval, we can study the relative delay of frames transmitted within an eligible interval. As it turns out, for those frames, the delay compared to the un-interfered execution is proportional to the credit of the shaper at the start of transmission in the interfered execution, which gives us the second main theorem from Cao et al. (2016b), see Fig. 4.

Theorem 2 *Given a stream X in some execution u , subject to any interference, and its associated un-interfered execution u_0 , for each frame $x \in \mathbb{F}^u \cap X$ we find $s^u(x) \leq s^{u_0}(x) + \frac{CR_X^u(s^u(x))}{\alpha_X^+}$.*

Proof The proof is with induction on the sequence of frames $x_k \dots x_l$ in an eligible interval E_s . For the first frame x_k we know using the previous theorem that $E_s \leq s^{u_0}(x_k)$. Furthermore, at E_s credit is 0, and in an eligible interval there is pending load, so until $s^u(x_k)$ credit will be rising with rate α_X^+ . As a consequence we find

$$s^u(x_k) = E_s + \frac{CR_X^u(s^u(x_k)) - CR_X^u(E_s)}{\alpha_X^+} \leq s^{u_0}(x_k) + \frac{CR_X^u(s^u(x_k))}{\alpha_X^+}.$$

Next, assume that the above relation holds for some frame x_j with $k \leq j < l$, then we have the following inequality.

$$s^u(x_j) \leq s^{u_0}(x_j) + \frac{CR_X^u(s^u(x_j))}{\alpha_X^+} \tag{20}$$

We find for x_{j+1} that credit falls during the transmission of x_j and then rises until the start of x_{j+1} because in an eligible interval there is always pending load, so credit at the start of x_{j+1} equals (see Fig. 5):

$$CR_X^u(s^u(x_{j+1})) = CR_X^u(s^u(x_j)) - \alpha_X^- \cdot C^u(x_j) + \alpha_X^+ \cdot (s^u(x_{j+1}) - s^u(x_j) - C^u(x_j))$$

Furthermore, we know from Corollary 1 that

$$s^{u_0}(x_{j+1}) \geq s^{u_0}(x_j) + C^{u_0}(x_j) \cdot \left(1 + \frac{\alpha_X^-}{\alpha_X^+}\right)$$

and so we obtain, using the induction hypothesis [Inequality (20)] in the second step,

$$\begin{aligned} s^u(x_{j+1}) &= s^u(x_j) + C^u(x_j) \cdot \left(1 + \frac{\alpha_X^-}{\alpha_X^+}\right) + \frac{CR_X^u(s^u(x_{j+1}))}{\alpha_X^+} - \frac{CR_X^u(s^u(x_j))}{\alpha_X^+} \\ &\leq s^u(x_j) + C^u(x_j) \cdot \left(1 + \frac{\alpha_X^-}{\alpha_X^+}\right) + \frac{CR_X^u(s^u(x_{j+1}))}{\alpha_X^+} - (s^u(x_j) - s^{u_0}(x_j)) \end{aligned}$$

$$\begin{aligned}
 &= s^{u_0}(x_j) + C^u(x_j) \cdot \left(1 + \frac{\alpha_X^-}{\alpha_X^+}\right) + \frac{CR_X^u(s^u(x_{j+1}))}{\alpha_X^+} \\
 &= s^{u_0}(x_j) + C^{u_0}(x_j) \cdot \left(1 + \frac{\alpha_X^-}{\alpha_X^+}\right) + \frac{CR_X^u(s^u(x_{j+1}))}{\alpha_X^+} \\
 &\leq s^{u_0}(x_{j+1}) + \frac{CR_X^u(s^u(x_{j+1}))}{\alpha_X^+},
 \end{aligned}$$

which concludes the proof. □

Now, to prove a bound on the delay in a given scenario, we only need to prove a bound on the maximum credit that can be built up in that scenario.

4.4 Bounding the credit build up by interference

From the previous theorem, it is straightforward to conclude that we can bound the start-time of any frame by bounding the maximal credit that can be build up by a shaper. We define:

$$CR_X^{\max} = \sup_{u \in \mathbb{U}} \sup_{t \in \mathbb{R}^+} CR_X^u(t).$$

To calculate this maximum credit, we need to estimate the amount of interference that can occur, since credit of X only builds up while pending frames of X wait for other transmissions to finish. To estimate the maximum amount of interference, particularly from a set of higher-priority shapers \mathbb{X} , we need to estimate the minimum credit of the set.

We define:

$$CR_{\mathbb{X}}^{\min} = \inf_{u \in \mathbb{U}} \inf_{t \in \mathbb{R}^+} CR_{\mathbb{X}}^u(t) = \inf_{u \in \mathbb{U}} \inf_{t \in \mathbb{R}^+} \sum_{X \in \mathbb{X}} CR_X^u(t)$$

and reason first about the minimum credit of a single priority class as follows.

Theorem 3 *For a single priority class $X \in \mathbb{X}$ under credit based shaping, we find $CR_X^{\min} = -\alpha_X^- \cdot C_X^{\max}$.*

Proof This was claimed but not proven in Cao et al. (2016a). For completeness, we have added the proof here.

Firstly, we have to prove that in any execution u and at any time $t \in \mathbb{R}^+$ we find $CR_X^u(t) \geq -\alpha_X^- \cdot C_X^{\max}$. To see this, assume the opposite, namely that there exists an execution v such that at some point t we have $CR_X^v(t) < -\alpha_X^- \cdot C_X^{\max}$. From the shaping rules we know this negative credit is only possible if a shaper is currently in transmission [Eq. (15)], or is recovering [Eq. (16)]. The latter means that there is an earlier time t' at which the credit is even lower, so without loss of generality we may assume that there exists an execution v and a point t at which the shaper is

in transmission and $CR_X^v(t) < -\alpha_X^- \cdot C_X^{\max}$. Let $x \in X$ be the frame that is under transmission at t , then we may derive that the credit at its start-time would be negative:

$$\begin{aligned} CR_X^v(s^v(x)) &= CR_X^v(t) + \alpha_X^- \cdot (t - s^v(x)) \\ &\leq CR_X^v(t) + \alpha_X^- \cdot C^v(x) \\ &\leq CR_X^v(t) + \alpha_X^- \cdot C_X^{\max} \\ &< 0 \end{aligned}$$

This contradicts the assumption that credit is non-negative at the start of each transmission [Eq. (9)] and proves that such an execution v can not exist.

Secondly, we have to prove that there exists an execution u in which at some time $t \in \mathbb{R}^+$ we actually have $CR_Y^u(t) = -\alpha_Y^- \cdot C_Y^{\max}$. For this, simply consider an execution in which a single frame from Y of maximal size is being transmitted, and take t to be the finish-time of that transmission. At that time the equation is satisfied. \square

Now, if next we would like to consider the total credit $CR_{\mathbb{X}}^u$ of a set of priority classes $\mathbb{X} \subseteq \mathbb{P}$, we could of course simply add up the minimum credits of each of its elements. However, for sets larger than 1 element, there is no execution in which this sum can actually be reached, because as the transmission from one of the elements of \mathbb{X} is proceeding towards the minimum, the credits of other elements in \mathbb{X} will be recovering. In the following theorem, we achieve a tight bound by taking this recovery into account.

Theorem 4 *Given a (finite) set of priority classes \mathbb{X} under credit based shaping, with $\alpha_{\mathbb{X}}^+ \leq BW$, and some execution $u \in \mathbb{U}$. If \mathbb{X} contains only one element, we may use the previous theorem. If \mathbb{X} contains more than one element then we find, iteratively:*

$$CR_{\mathbb{X}}^{\min} = - \sup_{X \in \mathbb{X}} (\alpha_X^- \cdot C_X^{\max} - CR_{\mathbb{X} \setminus X}^{\min}).$$

Proof We prove this by induction on the size of \mathbb{X} . The base case, obviously, is the case where \mathbb{X} only has one element, and was proven in the previous theorem. Subsequently, as induction hypothesis, assume that our theorem holds for all sets of the form $\mathbb{X} \setminus X$, with $X \in \mathbb{X}$.

Firstly, as in the previous theorem, we can prove by contradiction that for all executions u and times t we have $CR_{\mathbb{X}}^u(t) \geq - \sup_{X \in \mathbb{X}} (\alpha_X^- \cdot C_X^{\max} - CR_{\mathbb{X} \setminus X}^{\min})$. Assume there exists an execution v and time t such that $CR_{\mathbb{X}}^v(t) < - \sup_{X \in \mathbb{X}} (\alpha_X^- \cdot C_X^{\max} - CR_{\mathbb{X} \setminus X}^{\min})$. Then, in particular, for each individual $X \in \mathbb{X}$ we have

$$\begin{aligned} CR_X^v(t) &= CR_{\mathbb{X}}^v(t) - CR_{\mathbb{X} \setminus X}^v(t) \\ &< - \sup_{X' \in \mathbb{X}} (\alpha_{X'}^- \cdot C_{X'}^{\max} - CR_{\mathbb{X} \setminus X'}^{\min}) - CR_{\mathbb{X} \setminus X}^v(t) \\ &\leq -\alpha_X^- \cdot C_X^{\max} + CR_{\mathbb{X} \setminus X}^{\min} - CR_{\mathbb{X} \setminus X}^v(t) \\ &\leq -\alpha_X^- \cdot C_X^{\max} \end{aligned}$$

$$\begin{aligned}
 &= -(BW - \alpha_{\mathbb{X}}^+) \cdot C_X^{\max} \\
 &\leq 0
 \end{aligned}$$

So for each individual X we have negative credit at time t . From the shaping rules, we know that this is only possible if all shapers are currently either transmitting or in recovery. Like in the previous proof, we reason that credit at an earlier point would only be lower if all shapers are in recovery at t , and therefore we may assume without loss of generality that at time t one of the shapers is transmitting. Let $x \in X$ be the frame that is in transmission at t , and recall that all others are recovering. Furthermore, recall that at the start-time of this transmission we have $CR_X^v(s^v(x)) \geq 0$, so we may derive:

$$\begin{aligned}
 CR_{\mathbb{X} \setminus X}^v(s^v(x)) &= CR_{\mathbb{X}}^v(s^v(x)) - CR_X^v(s^v(x)) \\
 &\leq CR_{\mathbb{X}}^v(s^v(x)) \\
 &= CR_{\mathbb{X}}^v(t) - \alpha_{\mathbb{X} \setminus X}^+ \cdot (t - s^v(x)) + \alpha_X^- \cdot (t - s^v(x)) \\
 &= CR_{\mathbb{X}}^v(t) - \alpha_{\mathbb{X} \setminus X}^+ \cdot (t - s^v(x)) + (BW - \alpha_X^+) \cdot (t - s^v(x)) \\
 &= CR_{\mathbb{X}}^v(t) + (BW - \alpha_{\mathbb{X}}^+) \cdot (t - s^v(x)) \\
 &= CR_{\mathbb{X}}^v(t) + \alpha_{\mathbb{X}}^- \cdot (t - s^v(x)) \\
 &\leq CR_{\mathbb{X}}^v(t) + \alpha_{\mathbb{X}}^- \cdot C_X^{\max} \\
 &\quad \{\text{and by assumption}\} \\
 &< - \sup_{X' \in \mathbb{X}} (\alpha_{\mathbb{X}}^- \cdot C_{X'}^{\max} - CR_{\mathbb{X} \setminus X'}^{\min}) + \alpha_{\mathbb{X}}^- \cdot C_X^{\max} \\
 &\leq -\alpha_{\mathbb{X}}^- \cdot C_X^{\max} + CR_{\mathbb{X} \setminus X}^{\min} + \alpha_{\mathbb{X}}^- \cdot C_X^{\max} \\
 &= CR_{\mathbb{X} \setminus X}^{\min}
 \end{aligned}$$

which contradicts the induction hypothesis, and therefore proves our estimate is sound.

Secondly, to prove our estimate is tight, we need to construct an execution in which this estimate is actually achieved. This is relatively straightforward, using a recursion over the formula. For a set \mathbb{X} containing a single element, the construction is to just transmit a single maximal frame from the one priority class $X \in \mathbb{X}$. For a larger set \mathbb{X} , we may assume that there is an execution $u_{\mathbb{X} \setminus X}$ that achieves the minimum credit of $\mathbb{X} \setminus X$. We observe that in this execution it will not be necessary at any time to transmit a frame from X . Subsequently, we determine which $X \in \mathbb{X}$ achieves the supremum in the formula $-\sup_{X \in \mathbb{X}} (\alpha_{\mathbb{X}}^- \cdot C_X^{\max} - CR_{\mathbb{X} \setminus X}^{\min})$, and we create $u_{\mathbb{X}}$ by appending a single transmission of a maximal frame from X to the execution $u_{\mathbb{X} \setminus X}$, choosing the arrival time of the frame equal to the end of the last transmission in $u_{\mathbb{X} \setminus X}$. By construction, this will lead to the minimal credit of \mathbb{X} . In Sect. 4.6, we give a concrete example of such a construction, and in Sect. 5 we give further illustrations as part of our discussion on the tightness of our entire approach. \square

Iteratively, for a set \mathbb{X} containing N priorities classes, the construction of the last transmissions should be a series of N maximally-sized frames, one from each priority

class, and its order is determined by the calculation of the minimum credit, see the examples in Figs. 8 and 9. The order of the N priority classes plays a critical role to reach the minimum credit. To cover all the possibilities of ordering, it needs $N \cdot (N - 1) \cdot \dots \cdot 2 \cdot 1 = N!$ calculations to ensure the minimum credit. Hence, the complexity of calculation in this theorem is at most factorial.

4.5 Bounding the delay caused by interference

To get from the minimum credit to the maximum time a set of shapers can be submitting is less trivial. Notably, we constructed an execution that achieves the minimum credit, but this execution does not achieve the minimum in the slowest way possible. When estimating interference, it is important that the total credit of interfering (higher-priority) shapers drops as slowly as possible. This is the topic of the next theorem.

Theorem 5 *Given a set of priority classes \mathbb{X} under credit based shaping, if during an execution u at every point t' in an interval $[t, t'']$ there is a frame from \mathbb{X} transmitting, then the total credit of \mathbb{X} decreases with at least a rate $\alpha_{\mathbb{X}}^-$, i.e.*

$$CR_{\mathbb{X}}^u(t') \leq CR_{\mathbb{X}}^u(t) - \alpha_{\mathbb{X}}^- \cdot (t - t').$$

Proof The slowest possible decrease of total credit is achieved, obviously, when credit for all shapers in \mathbb{X} is increasing, except for the shaper that is currently transmitting. Regardless of whether the shapers are increasing because they are recovering or have pending load, assume without loss of generality that shaper $X \in \mathbb{X}$ is transmitting and find:

$$\begin{aligned} CR_{\mathbb{X}}^u(t') &\leq CR_{\mathbb{X}}^u(t) + \alpha_{\mathbb{X} \setminus X}^+ \cdot (t - t') - \alpha_X^- \cdot (t - t') \\ &= CR_{\mathbb{X}}^u(t) + \alpha_{\mathbb{X} \setminus X}^+ \cdot (t - t') - BW + \alpha_X^+ \cdot (t - t') \\ &= CR_{\mathbb{X}}^u(t) + \alpha_{\mathbb{X}}^- \cdot (t - t') \end{aligned}$$

which concludes the proof. \square

Corollary 2 *Given a set of priority classes \mathbb{X} under credit based shaping, if during an execution u at every point t' in an interval $[t, t'']$ all shapers from \mathbb{X} either have pending load or negative credit, then the total credit of \mathbb{X} decreases with exactly a rate $\alpha_{\mathbb{X}}^-$, i.e.*

$$CR_{\mathbb{X}}^u(t') = CR_{\mathbb{X}}^u(t) - \alpha_{\mathbb{X}}^- \cdot (t - t').$$

Finally, combining Theorems 4 and 5, we estimate the maximum credit rise of a priority class X by adding the possible interference from lower and higher priority classes. From lower priority classes, only a single frame can interfere which started non-preemptively ‘just before’ the arrival of a frame from X . In higher priority classes, the same lower priority frame will lead to an increase in the total credit, and the two

previous theorems combined determine a bound on the total time it takes before this total credit is depleted.

The following theorem gives the upper bound of the relative delay, i.e., $\sup_{u \in \mathbb{U}} (s^u(x) - s^{u_0}(x))$ in the relative worst-case response time calculation [see Eq. (4)]. That bound, notably, is not necessarily tight, because the execution that leads to a slowest possible decrease in credit is not always also an execution that leads to minimum credit. This is the topic of Sect. 5.

Theorem 6 *Given a particular stream of interest associated with priority class $M \in \mathbb{P}$. Consider the set $\mathbb{H} = \{X \in \mathbb{P} \mid X > M\}$ of all higher priority classes, and the set $\mathbb{L} = \{X \in \mathbb{P} \mid X < M\}$ of all lower priority classes. Assume that M and \mathbb{H} are under credit based shaping, and $\alpha_{\mathbb{H}}^+ + \alpha_M^+ \leq BW$. Then, for any execution u with associated non-interfered schedule u_0 and any frame $m \in M$ we find:*

$$s^u(m) - s^{u_0}(m) \leq C_{\mathbb{L}}^{\max} \cdot \left(1 + \frac{\alpha_{\mathbb{H}}^+}{\alpha_{\mathbb{H}}^-} \right) - \frac{CR_{\mathbb{H}}^{\min}}{\alpha_{\mathbb{H}}^-}.$$

Proof To prove this theorem, we only need to show that during any execution u , the credit $CR_M^u(t)$ is bounded by

$$\alpha_M^+ \cdot \left(C_{\mathbb{L}}^{\max} \cdot \left(1 + \frac{\alpha_{\mathbb{H}}^+}{\alpha_{\mathbb{H}}^-} \right) - \frac{CR_{\mathbb{H}}^{\min}}{\alpha_{\mathbb{H}}^-} \right),$$

and subsequently use Theorem 2. Next, to study the variation of the credit $CR_M^u(t)$ over time, we divide time in the execution u into phases of three different types depending on what is in transmission: idling phase (nothing is in transmission), L phase (only \mathbb{L} frames transmit) and MH phase (only frames from M or \mathbb{H} transmit). Since these three conditions are mutually exclusive and cover all possibilities, a phase only ends when another type of phase takes over.

Idling Phase

When nothing is in transmission, either the credit of every priority class in \mathbb{H} and the credit of M is negative (transmission is not allowed by the shaper), or there is no pending load (positive credit is reset to 0). Hence, at any time t in an idling phase, the credit of every stream in $X \in \mathbb{H} \cup \{M\}$ is less than or equal to 0: $CR_X^u(t) \leq 0$.

L Phase

An L frame cannot start transmission when either the credit of M or the credit of some $H \in \mathbb{H}$ is strictly positive (because this implies pending load of a higher priority class). Therefore, we find those credits are 0 or negative at the start of any L transmission. During the transmission, credit of $X \in \mathbb{H} \cup \{M\}$ rises with a rate at most α_X^+ . Credit of $\mathbb{H} \cup \{M\}$ only rises above 0 if there are frames pending, so the increase may be less. Once credit of $\mathbb{H} \cup \{M\}$ rises strictly above 0 no new L transmission can start, but the current one can finish. As a consequence, for any $X \in \mathbb{H} \cup \{M\}$ we find credit at a time t in an L phase is bounded by the transmission of a single L frame: $CR_X^u(t) \leq C_L^{\max} \cdot \alpha_X^+$.

MH Phase

The credit variation is more complex in an *MH* phase. Let t_s denote the start time of a given *MH* phase. To bound the credit of M at any time t ($t \geq t_s$) during this phase, we study the total (cumulative) time during which frames from $\mathbb{H} \cup \{M\}$ transmit in $[t_s, t]$. We denote this by writing δ_X for any $X \in \mathbb{H} \cup \{M\}$ and know that $\sum_{X \in \mathbb{H} \cup \{M\}} \delta_X = t - t_s$. Our subsequent reasoning is mainly based on the credit variation rules and the reservation bound $\alpha_{\mathbb{H}}^+ + \alpha_M^+ \leq BW$.

Clearly, in an *MH* phase, there is always a transmission of some kind going on. Therefore, the credit of some $X \in \mathbb{H} \cup \{M\}$ will rise as long as it is not transmitting itself, and as long as it is negative or has pending load. If there is no pending load, credit will not rise. For a stream $X \in \mathbb{H} \cup \{M\}$ at time t in an *MH* phase, we therefore know that:

$$CR_X^u(t) \leq CR_X^u(t_s) - \alpha_X^- \cdot \delta_X + \alpha_X^+ \cdot \sum_{H \in \mathbb{H} \cup \{M\} \setminus X} \delta_H$$

In Corollary 2, we found that the total credit of \mathbb{H} rises with a rate $\alpha_{\mathbb{H}}^+$ while M is transmitting and drops with a rate $\alpha_{\mathbb{H}}^-$ when one of the classes in \mathbb{H} is transmitting. Therefore, considering the total credit of the higher priorities, this gives us:

$$CR_{\mathbb{H}}^u(t) \leq CR_{\mathbb{H}}^u(t_s) + \alpha_{\mathbb{H}}^+ \cdot \delta_M - \alpha_{\mathbb{H}}^- \cdot \sum_{H \in \mathbb{H}} \delta_H.$$

Note that this upper bound is reached only if throughout the interval $[t_s, t]$, when one stream is in transmission, all other streams keep increasing their credits. Also, using Theorem 4, we know that the total credit of the higher priorities has a lower bound of $CR_{\mathbb{H}}^{min}$:

$$CR_{\mathbb{H}}^u(t) \geq CR_{\mathbb{H}}^{min}.$$

And so we obtain:

$$CR_{\mathbb{H}}^u(t_s) + \alpha_{\mathbb{H}}^+ \cdot \delta_M - \alpha_{\mathbb{H}}^- \cdot \sum_{H \in \mathbb{H}} \delta_H \geq CR_{\mathbb{H}}^{min}.$$

Next we use the definition $\alpha_X^- = BW - \alpha_X^+$ to derive that:

$$\alpha_{\mathbb{H}}^+ + \alpha_M^+ \leq BW \iff \alpha_{\mathbb{H}}^+ \cdot \alpha_M^+ \leq \alpha_{\mathbb{H}}^- \cdot \alpha_M^-.$$

and we can further refine the upper bound on $CR_M^u(t)$:

$$\begin{aligned} CR_M^u(t) &\leq CR_M^u(t_s) + \alpha_M^+ \cdot \sum_{H \in \mathbb{H}} \delta_H - \alpha_M^- \cdot \delta_M \\ &\leq CR_M^u(t_s) + \frac{\alpha_M^+}{\alpha_{\mathbb{H}}^-} \cdot (CR_{\mathbb{H}}^u(t_s) + \alpha_{\mathbb{H}}^+ \cdot \delta_M - CR_{\mathbb{H}}^{min}) - \alpha_M^- \cdot \delta_M \end{aligned}$$

Table 1 An example of three high priorities levels ($H_1 > H_2 > H_3$)

	α^+	α^-	C^{\max}	CR^{\min}
H_1	10	90	3	- 270
H_2	20	80	2	- 160
H_3	15	85	4	- 340
L			5	

$$\begin{aligned}
 &= CR_M^u(t_s) + \frac{\alpha_M^+}{\alpha_{\mathbb{H}}^-} \cdot (CR_{\mathbb{H}}^u(t_s) - CR_{\mathbb{H}}^{\min}) + \left(\frac{\alpha_M^+ \cdot \alpha_{\mathbb{H}}^+}{\alpha_{\mathbb{H}}^-} - \alpha_M^- \right) \cdot \delta_M \\
 &\leq CR_M^u(t_s) + \frac{\alpha_M^+}{\alpha_{\mathbb{H}}^-} \cdot (CR_{\mathbb{H}}^u(t_s) - CR_{\mathbb{H}}^{\min}).
 \end{aligned}$$

Finally, an MH phase can only follow an idling phase or an L phase. Therefore, the initial credits $CR_X^u(t_s)$ are bounded by the results obtained in those cases, and we derive:

$$\begin{aligned}
 CR_M^u(t) &\leq C_{\mathbb{L}}^{\max} \cdot \alpha_M^+ + \frac{\alpha_M^+}{\alpha_{\mathbb{H}}^-} \cdot (C_{\mathbb{L}}^{\max} \cdot \alpha_{\mathbb{H}}^+ - CR_{\mathbb{H}}^{\min}) \\
 &= \alpha_M^+ \cdot \left(C_{\mathbb{L}}^{\max} \cdot \left(1 + \frac{\alpha_{\mathbb{H}}^+}{\alpha_{\mathbb{H}}^-} \right) - \frac{CR_{\mathbb{H}}^{\min}}{\alpha_{\mathbb{H}}^-} \right),
 \end{aligned}$$

which concludes our proof. □

4.6 An example of relative worst-case response time

Now we use Theorems 4 and 6 to calculate the worst-case relative response time of stream M given a concrete set-up of the inter-priority interference as shown in Table 1. In this example, α_M^+ is set to 10 Mbps given a 100 Mbps total bandwidth. Note that the value of α_M^+ is relevant to the absolute value of credit M , but is not used in the calculation of the worst-case relative delay. Therefore it is not mentioned in the table.

There are three high priorities, H_1 , H_2 and H_3 , in the order of priority. The total reservation of high priorities $\alpha_{\mathbb{H}}^+$ is 45 Mbps and the remaining $\alpha_{\mathbb{H}}^-$ is 55 Mbps. As α_M^+ is set to 10 Mbps, the condition $\alpha_{\mathbb{H}}^+ + \alpha_M^+ \leq BW$ is met, thus guaranteeing that the worst-case response time is finite.

To calculate the minimum credit $CR_{\mathbb{H}}^{\min}$, we mainly use Theorem 4. We start with the minimum credit of a single priority, shown in Table 1, then with the set of two high priorities, and finally with the set of three.

Credit of the set of two priorities

$$\begin{aligned}
 CR_{\{H_1, H_2\}}^{\min} &= - \max \left(\alpha_{\{H_1, H_2\}}^- \cdot C_{H_1}^{\max} - CR_{H_2}^{\min}, \alpha_{\{H_1, H_2\}}^- \cdot C_{H_2}^{\max} - CR_{H_1}^{\min} \right) \\
 &= - \max (370, 410) \\
 &= - 410
 \end{aligned}$$

$$\begin{aligned}
 CR_{\{H_2, H_3\}}^{\min} &= -\max\left(\alpha_{\{H_2, H_3\}}^- \cdot C_{H_2}^{\max} - CR_{H_3}^{\min}, \alpha_{\{H_2, H_3\}}^- \cdot C_{H_3}^{\max} - CR_{H_2}^{\min}\right) \\
 &= -\max(470, 420) \\
 &= -470 \\
 CR_{\{H_1, H_3\}}^{\min} &= -\max\left(\alpha_{\{H_1, H_3\}}^- \cdot C_{H_1}^{\max} - CR_{H_3}^{\min}, \alpha_{\{H_1, H_3\}}^- \cdot C_{H_3}^{\max} - CR_{H_1}^{\min}\right) \\
 &= -\max(565, 570) \\
 &= -570
 \end{aligned}$$

Credit of the set of three priorities

$$\begin{aligned}
 CR_{\mathbb{H}}^{\min} &= -\max\left(\alpha_{\mathbb{H}}^- \cdot C_{H_1}^{\max} - CR_{\{H_2, H_3\}}^{\min}, \alpha_{\mathbb{H}}^- \cdot C_{H_2}^{\max} - CR_{\{H_1, H_3\}}^{\min}, \alpha_{\mathbb{H}}^- \cdot C_{H_3}^{\max} - CR_{\{H_1, H_2\}}^{\min}\right) \\
 &= -\max(635, 680, 630) \\
 &= -680
 \end{aligned}$$

We construct a sequence H_1, H_3, H_2 in this example, so that the minimum credit is obtained through the calculations of $CR_{H_1}^{\min}$, $CR_{\{H_1, H_3\}}^{\min}$ and $CR_{\{H_1, H_3, H_2\}}^{\min}$ as follows:

$$\begin{aligned}
 CR_{H_1}^{\min} &= -\alpha_{H_1}^- \cdot C_{H_1}^{\max} + CR_{\emptyset}^{\min} \\
 CR_{\{H_1, H_3\}}^{\min} &= -\alpha_{\{H_1, H_3\}}^- \cdot C_{H_3}^{\max} + CR_{H_1}^{\min} \\
 CR_{\{H_1, H_3, H_2\}}^{\min} &= -\alpha_{\{H_1, H_3, H_2\}}^- \cdot C_{H_2}^{\max} + CR_{\{H_1, H_3\}}^{\min}
 \end{aligned}$$

As a next step, we use Theorem 6 to obtain the worst-case relative delay.

$$C_{\mathbb{L}}^{\max} \cdot \left(1 + \frac{\alpha_{\mathbb{H}}^+}{\alpha_{\mathbb{H}}^-}\right) - \frac{CR_{\mathbb{H}}^{\min}}{\alpha_{\mathbb{H}}^-} = 5 \cdot \left(1 + \frac{45}{55}\right) + \frac{680}{55} \approx 21.45$$

We show one feasible worst-case scenario in Fig. 6. As the total credit $CR_{\mathbb{H}}$ reaches its minimum at the slowest descent $\alpha_{\mathbb{H}}^- = 55$ Mbps, the calculated worst-case relative delay 21.45 is attained. In this case, observe that the estimate is tight since there exists an execution that achieves the worst-case relative delay. In the next section, tightness is extensively studied by determining sufficient conditions under which such an execution exists in general.

5 Tightness

In this section, we study the conditions under which our relative worst-case response-time analysis is tight. Firstly, we recapitulate our earlier result that it is always tight in case of a single higher priority class. Secondly, we use a number of examples to illustrate why tightness is not guaranteed in case of multiple high-priorities. And thirdly, we construct

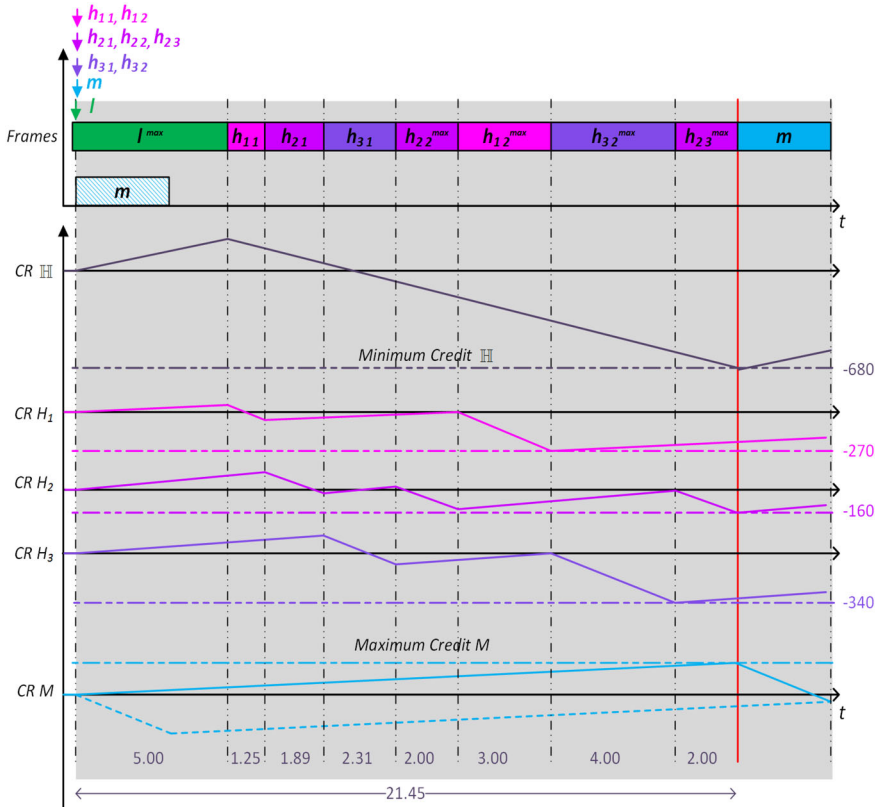


Fig. 6 A stream of M interfered by a set of high priority streams $\mathbb{H} = \{H_1, H_2, H_3\}$ and a low-priority stream L as shown in Table 1. Arrows here indicate frame arrivals. The solid rectangles represent the transmission of streams \mathbb{H} , M and L while the shaded ones represent the interference-free transmission of M . So do the solid and dashed lines representing the credit evolution of M . The eligible interval of stream M has been labeled in grey context. The evolution of the total credit $CR_{\mathbb{H}}$ as well as the individual credit CR_{H_1}, CR_{H_2} and CR_{H_3} has been shown along with the frame transmission. The minimum credit of \mathbb{H} and the maximum credit of M has been marked with dash-dotted lines. It has been shown that the total credit $CR_{\mathbb{H}}$ decreases at the slowest descend to reach the minimum, i.e., when one credit is decreasing and the rest two are always increasing. In this way, the maximum value of credit M is reached when the total credit of \mathbb{H} reaches its minimum and the calculated worst-case relative delay is thus attained. The solid vertical line in red marks the timing when credit M reached the maximum value that we proved

a quite elaborate scenario that reveals a sufficient (and conjectured necessary) condition on the higher priority classes under which our result is tight.

5.1 Tightness for interference by a single higher priority class

To illustrate the worst-case scenario that underlies Theorem 6, we repeat our arguments from Cao et al. (2016a).

Consider the left-most eligible interval in Fig. 7. This figure shows how a maximum credit for class M , and hence the maximum delay for a frame from that class, can be

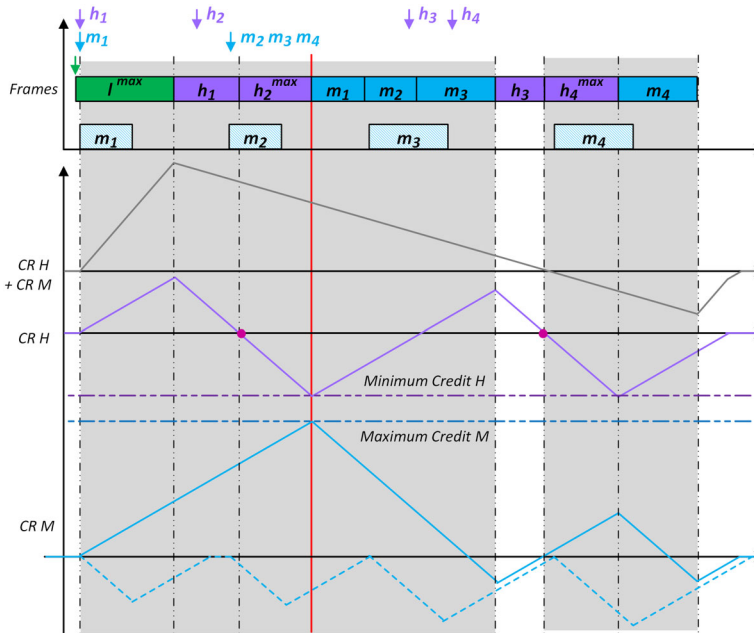


Fig. 7 A stream of M interfered by a stream of H and L , and $\alpha_H^+ + \alpha_M^+ < BW$. The solid rectangles represent the transmission of streams H , M and L while the shaded ones represent the interference-free transmission of M . So do the solid and dashed lines representing the credit evolution. Frames of maximum transmission time have been labeled with superscript max , and the eligible interval has been labeled in grey context. The two dots on the curve of credit H emphasize the moment when credit reaches exactly zero and a frame of maximum size starts its transmission. The solid vertical line in red marks the timing when credit M reached the maximum value that we proved

achieved in case we have *only a single* higher priority class H and one lower priority class L . The example, however, also works for multiple lower priority classes, since only one frame of lower priority can transmit in any case. The downward arrows indicate the arrival of frames, the solid rectangles indicate the execution of the combined H, M and L streams, the shaded rectangles indicate how the M -stream would execute in a scenario without interference. The graphs below this indicate the total credit level of both shapers, and the individual credit levels of H and M during execution.

In the worst-case scenario, the eligible interval starts with an L frame of maximum transmission time. Immediately after the start of L , the frame m_1 (our frame of interest) arrives together with an H -frame. As a consequence, credit builds up for M and H . Subsequently, frames from H start transmitting and new frames arrive until the credit built up by the L transmission is exactly depleted. This may take a careful choice of non-maximal frame-sizes. Lastly, a maximal frame of H arrives before credit is entirely depleted, to ensure that the credit of H reaches its minimal value (which in case of a single priority class equals $CR_H^{\min} = -\alpha_H^- \cdot C_H^{\max}$). Now, frame m_1 is ready to transmit, but at this point the credit of M has reached its maximum value, and therefore frame m_1 has experienced its maximum delay compared to the un-interfered schedule. In the

Table 2 An example of four priorities levels ($H_1 > H_2 > H_3 > H_4$) in which reaching the minimum credit with the slowest decreasing rate is feasible in the last 4 transmissions

	α^+	α^-	C^{\max}	CR^{\min}
H_1	10	90	5	-450
H_2	10	90	6	-540
H_3	15	85	8	-680
H_4	10	90	4	-360

execution that follows in Fig. 7, we see that the total credit keeps on decreasing, and that the maximum credit level of M is never obtained again. Only in case $\alpha_H^+ + \alpha_M^+ = BW$, a stream M may reach its maximum more than once in a single eligible interval, because the total credit does not decrease.

To prove that the formula developed in the previous section gives a tight bound on the worst-case interference when there is only a single higher priority class, it is enough to realize that the example illustrated in Fig. 7 in fact provides a generic construction of the worst case. For any given medium-priority scenario, the arrival of a low-priority frame and immediately subsequent arrival of a number of high-priority frames of a well-chosen size can delay the transmission of any medium-priority frame by the amount specified in the formula. This construction can be carried out for any frame in a particular shaped stream, as long as we have the freedom to assign a worst case arrival time to a single low priority frame and have the freedom to inject a burst of high priority frames of the right size. Therefore, our approach is tight under the assumptions made in our system model, i.e. the estimate cannot be improved without further knowledge of the interference.

5.2 Examples why tightness is not always guaranteed

To understand why tightness is not always guaranteed, recall the way to calculate the minimum credit of a set of N priority classes under credit-based shaping, given in Theorem 4. To reach the minimum credit, the critical part is the transmission of N frames, one of each priority level, in the order determined by the iterative calculation of the minimum credit. Furthermore, recall that according to Theorem 6, this minimum credit should be reached using the slowest possible descent of the cumulative credit, while Corollary 2 shows that this slowest descent is attained if all credits are increasing at all times, except for the one that is transmitting.

In Table 2, we have specified a scenario consisting of four higher priority classes, interfering with our medium priority stream. To reach minimum cumulative credit for those 4 priority classes, one only needs to perform the execution of four maximum frames, one of each class, in the order H_2, H_3, H_1, H_4 , as illustrated in the left panel of Fig. 8. To reach minimum credit using the slowest possible descend, however, one should make sure that credit of each of the shapers is rising whenever it is not transmitting. Since transmissions should start at 0 credit (otherwise, minimum credit is not reached), this means that credit of all shapers must be *recovering* before each transmission. This is

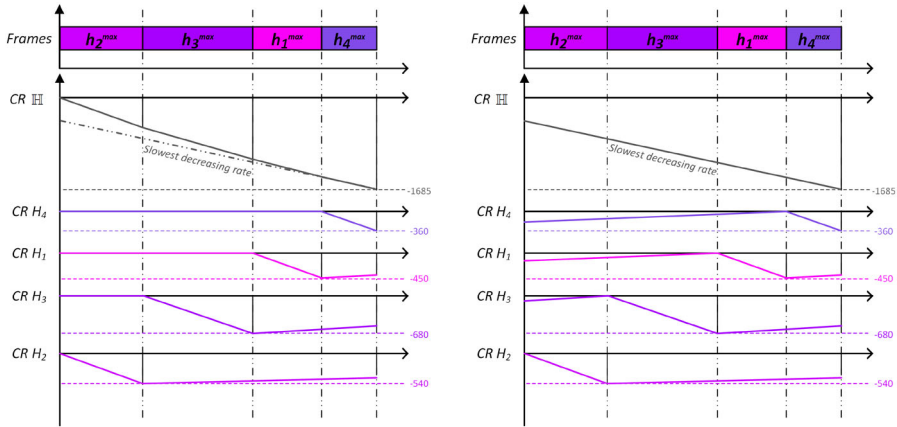


Fig. 8 Example on how to reach minimum cumulative credit, and reach it using slowest possible descent. The last four transmissions are maximally-sized frames from H_2, H_3, H_1, H_4 , determined by the minimum total credit calculation, and the total minimum credit CR_{\min}^{\min} is -1685 . Left panel: the total credit reaches the minimum but not at the slowest descent. Right panel: the total credit reaches the minimum at the slowest descent, i.e., during the transmission of one priority, all other credits are increasing. Note the connection between this figure, and the construction given in the proof of Theorem 6. The transmissions given here are also the last transmissions in that construction

Table 3 An example of two priorities levels ($H_1 > H_2$) in which reaching the minimum credit with the slowest decreasing rate is not feasible in the last two transmissions

	α^+	α^-	C^{\max}	CR^{\min}
H_1	20	80	1	- 80
H_2	40	60	6	- 360

illustrated in the right panel of Fig. 8. Naturally, these recoveries can only be caused by earlier transmissions.

Now, to see why in the case of multiple high priorities it is not always possible to reach the worst-case response time, consider the slightly simpler scenario given in Table 3, where two higher priority streams are interfering with our stream of interest. In this scenario, we need to transmit a maximal frame of H_2 followed by a maximal frame of H_1 , and credit of H_1 needs to be recovering during the transmission of H_2 . Now, the illustration in Fig. 9 clearly shows that, if H_1 has to recover exactly at the start of its final transmission, then this recovery must have started at a value that is lower than what can be achieved by a single transmission of H_1 . A second transmission is not possible, given negative credit, and therefore minimum cumulative credit cannot be attained in the slowest possible way using this sequence of transmissions. Furthermore, we have not been able to find another sequence of transmissions that would have the desired result.

5.3 Tightness for interference of multiple higher priority classes

Now that we have shown tightness for a single interfering higher priority class, and given examples why tightness is not self-evident for multiple higher priority interferences, let us

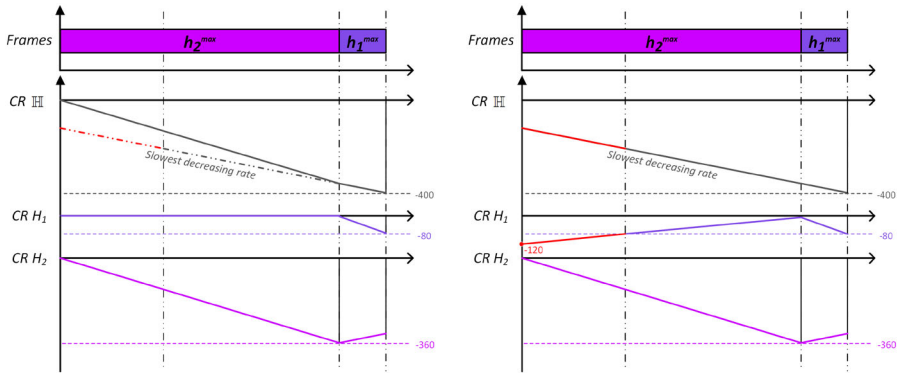


Fig. 9 Example of a case where attaining minimum cumulative credit at a slowest descent is not possible. The last two transmissions are maximally-sized frames from H_2 , H_1 , determined by the minimum total credit calculation, and the total minimum credit $CR_{\mathbb{H}}^{\min}$ is -400 . Left panel: the total credit reaches the minimum but not at the slowest descend. Right panel: the total credit cannot reach the minimum and meanwhile decrease at the slowest descend. At the start of the first transmission, the credit of H_1 violates its own minimum credit bound, resulting in an impossible scenario

study the latter in more detail. In case of a stream M interfered by a set of multiple higher priority classes \mathbb{H} and lower priority classes \mathbb{L} , our claims of tightness are conditional. Of course, still only one lower priority frame can interfere. But as indicated before, if we try to create a scenario that achieves the worst-case, we need to aim for interference by the higher priority classes that at the same time a) reaches the minimum total credit $CR_{\mathbb{H}}^{\min}$ and b) achieves this in the slowest way possible. In other words, we need a single scenario that serves as a witness for both Theorems 4 and 5. This is not always possible, but it is possible to turn the combined construction of both witnesses into a sufficient condition for tightness.

Theorem 7 Given a stream of priority class $M \in \mathbb{P}$, we use $\mathbb{H} = \{X \in \mathbb{P} \mid X > M\}$ to denote all streams of higher priority and $\mathbb{L} = \{X \in \mathbb{P} \mid X < M\}$ to denote all streams of lower priority. Assume that M and \mathbb{H} are under credit based shaping and $\alpha_{\mathbb{H}}^+ + \alpha_M^+ \leq BW$. If there exists a sequence $H_1 \dots H_N \in \mathbb{H}$, with N the number of streams in \mathbb{H} , such that each priority class only occurs once in the sequence (but not necessarily in order of priority), and if for all $1 \leq n \leq N$:

$$CR_{\bigcup_{1 \leq i \leq n} H_i}^{\min} = -\alpha_{\bigcup_{1 \leq i \leq n} H_n}^- \cdot C_{H_n}^{\max} + CR_{\bigcup_{1 \leq i < n} H_i}^{\min},$$

and furthermore

$$C_{H_N}^{\max} \geq \frac{\alpha_{H_N}^+}{\alpha_{H_N}^-} \sum_{1 \leq j < N} C_{H_j}^{\max},$$

then the bound on the worst-case relative delay of Theorem 6 is tight.

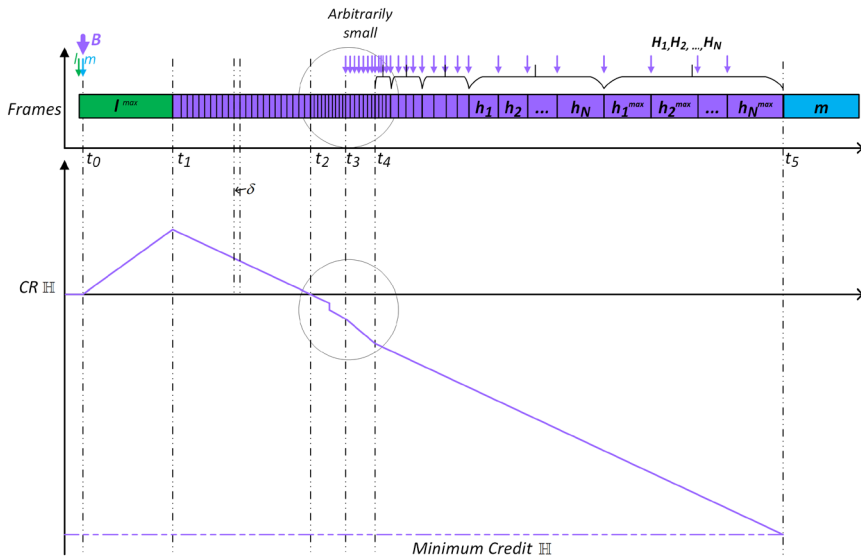


Fig. 10 Construction of an arbitrary approximation of the worst-case response time in five phases. In the first phase, higher priority credit is built up using a maximal lower priority frame. In the second phase, the built-up credit is slowly depleted until zero, and in the fifth and final phase, the credit is slowly depleted until a minimum credit is reached. The third and fourth phase allow for an arbitrarily small drop in credit during an arbitrarily small amount of time, constructed in such a way that a valid schedule between second and fifth phase is created

Proof In the appendix, we give a generic construction of a scenario u^ϵ that maintains a slowest possible decrease of credit at all times, except for an arbitrarily small time during which an arbitrarily small drop in credit may take place due to resets.

Admittedly, the scenario is quite complicated. It consists of five phases, illustrated in Fig. 10. The first-phase consists, like in the single higher-priority case, of a transmission of a lower priority frame, while all other shapers receive pending load just after the transmission of the lower priority frame has started. In the figure, this is indicated by a burst B .

In the second phase, the cumulative credit that has increased during the first phase is depleted using the burst B , which is a large burst of higher-priority frames of arbitrarily small size $\delta > 0$. As a consequence, the cumulative credit reaches 0, while all higher priority shapers have pending load at all times, and some higher priority is transmitting at all times. This means, using Corollary 2, that credit drops at its slowest possible speed during this second phase. In the third and fourth phase, which only take an arbitrarily short amount of time, the cumulative credit may drop by a small amount because some (but not all) shapers experience a credit reset. After this reset, a number of arbitrarily small transmissions take place to put the credit of all higher-priority shapers at the right (negative) amount to start with the fifth phase. Finally, in the fifth phase, credit of all higher-priority shapers remains negative at all times, except for the one that needs to

transmit, which has credit 0 exactly when needed. We repeatedly use transmissions in order of the sequence $H_1 \dots H_N \in \mathbb{H}$, with ever larger frame-sizes.

The construction of u^ϵ leads to a valid execution whenever the sequence $H_1 \dots H_N$ satisfies:

$$\frac{C_{H_i}^{\max}}{\alpha_{H_i}^+} \geq \frac{C_{H_{i+1}}^{\max}}{\alpha_{H_{i+1}}^+}$$

for all i . If furthermore the sequence satisfies the (strictly stronger) conditions given in the theorem, then the last iteration of transmissions $H_1 \dots H_N$ consist of maximal frame sizes, resulting in the minimum credit. Therefore, the execution u^ϵ is a witness of an arbitrary approximation of the worst-case response time for a frame $m \in M$. \square

Conjecture 1 We conjecture that the conditions given in Theorem 7 are in fact necessary and sufficient for tightness.

The conjecture, of course, is based on the example discussed in the previous section. It is certainly necessary to approximate minimum credit in order to approximate the worst-case response time, and any sequence of transmissions approximating minimum credit has to use frames approximating maximal size. (Assuming otherwise leads to a contradiction, since a larger frame-size immediately leads to lower credit.) The conditions in Theorem 7 exactly describe a possible last sequence of transmissions needed to actually reach minimum credit. We conjecture that this is, in fact, the only possible sequence that leads to minimum credit, but a formal proof of this is left for future research.

Finally, note that the complexity of checking the condition for tightness just involves one inequality check, once the order of the sequence to reach the minimum credit is determined. The complexity of the minimum credit analysis is factorial in the number of higher priority streams as explained in Sect. 4.4.

6 Comparison with earlier work

In this section, we perform a comparison with the busy period analyses presented in Axer et al. (2014) and Bordoloi et al. (2014). For this, we focus on scenarios in which there is only a single high priority. In this case, our own approach is tight, assuming there is no additional information regarding interference. However, in the work of Axer et al. (2014) and Bordoloi et al. (2014) interference is assumed to be periodic, or at least have a minimum inter-arrival time. This additional information makes a comparison still interesting. Furthermore, the comparison may illustrate how certain sources of pessimism present in the approaches of Axer et al. (2014) and Bordoloi et al. (2014) have been remedied by the use of eligible intervals.

It turned out to be difficult to perform a fair comparison in case of multiple high priorities. In both Axer et al. (2014) and Bordoloi et al. (2014), there is the implicit suggestion that multiple high priorities may be addressed by summing up the interference from all

Table 4 Example scenario, calculating worst-case response-times for class M sources τ_1 , τ_2 and τ_3 using the analysis presented in this paper, given a total bandwidth of 100 Mbps

	P	α^+ Mbps	C μ s	C_i μ s	T_i μ s	WR μ s
	H	40	1	-	-	-
τ_1	M	40	3	1	25	17.83
τ_2				3	30	14.83
τ_3				2	20	16.33
	L	-	2	-	-	-

The parameters on priority classes are in grey

high priorities. However, no clear calculation was given for this case, and the authors only studied examples that covered a single higher priority in their papers. Therefore, a comparison considering multiple high priorities would be rather artificial and subject to interpretation.

6.1 Illustrative example given a single high priority

Theorem 6, which we consider the main theorem of this paper, is a relative result. In case of a single high priority, it states that the worst-case response-time of any class M frame is tightly bound by its worst-case response-time in an un-interfered schedule plus a constant term determined by the maximum frame size of high and low priority traffic and the reservation of the high priority shaper. In order to compute the actual response-time of a frame, just knowing the relative delay compared to an un-interfered schedule is not sufficient. We also need to calculate the worst-case response-time of that un-interfered schedule. For this, it is necessary to make some assumptions on the class M traffic, which causes intra-class interference. Up to this point, such assumptions did not play a part in our system model. For the main result of this paper, it is not important how the un-interfered schedule is analysed, but when comparing our work to that of others, it is important that we have a way to do so.

As an arbitrarily chosen illustrative example, let us consider three sources of class M traffic: τ_1 , τ_2 , τ_3 . In line with the assumptions in Bordoloi et al. (2014), we assume that each source has periodic behavior characterized by the inter-arrival time T_i , and a maximum frame-size C_i . Note, that we only need to assume this for the class M traffic, not for the high- and low-priority sources of interference. For the latter, we only assume maximum frame sizes C_L^{\max} and C_H^{\max} , and the reservation of high priority α_H^+ . This is summarized in Table 4, where we assign values to those parameters. Finally, we assume a bandwidth $BW = 100$ Mbps in this table, and choose the respective reservations of the shapers to be 40% of this, so: $\alpha_H^+ = \alpha_M^+ = 40$ Mbps and $\alpha_H^- = \alpha_M^- = 60$ Mbps.

To calculate the response-time of the un-interfered M -priority schedule, we use the fact that this schedule behaves as a FIFO queue with added recovery times. Under the condition that the utilization of the sources is less than the reservation, i.e.: $\sum_{\tau_i \in M} \frac{C_i}{T_i} \leq$

$\frac{\alpha_M^+}{BW}$, we can perform a simple busy period analysis [see appendix in Cao et al. (2016a)] to find that the worst-case response-time for any frame from source τ_i is given by:

$$WR_{U_0}(\tau_i) = \left(\sum_{\tau_j \in M, j \neq i} C_j \cdot \left(1 + \frac{\alpha_M^-}{\alpha_M^+} \right) \right) + C_i.$$

Note, that the utilization condition is necessary and sufficient in order to guarantee that the un-interfered schedule has finite worst-case response times. Also note that the formula above is tight, because the worst-case scenario is simply achieved when all M sources start simultaneously. Then, adding the interference according to Theorem 6 gives us:

$$WR_U(\tau_i) = WR_{U_0}(\tau_i) + C_L^{\max} \cdot \left(1 + \frac{\alpha_H^+}{\alpha_H^-} \right) + C_H^{\max},$$

which returns the values in the right part of Table 4.

6.2 Applying Axer et al. (2014) and Bordoloi et al. (2014) to the same example

Now, in order to compare our approach with that of Axer et al. (2014) and Bordoloi et al. (2014), we must first observe that in those approaches more knowledge regarding the interfering traffic is assumed than in ours. In particular, in Bordoloi et al. (2014) it is assumed that also the high priority traffic is characterized as a set of periodic sources. Moreover, a deadline associated with each high-priority task is used in the calculations of Bordoloi et al. (2014) that compensate for the fact that some high-priority traffic transmits during the recovery time. In Axer et al. (2014), the Compositional Performance Analysis approach is used, meaning that high and medium priority traffic are characterized by arrival curves instead of as periodic sources. An arrival curve specifies the maximal and minimal amount of traffic in an arbitrary interval of a certain size. In this comparison, we follow the assumptions of Bordoloi et al. (2014), because it is a strictly stronger assumption than that of Axer et al. (2014). Any periodic source can easily be represented as a (periodic) arrival curve.

Potentially, the addition of information on the interference means that the worst-case performance bounds can be improved over ours, even though our approach is tight for our system model. Further on, we present examples when this is indeed the case, but also examples where both Axer et al. (2014) and Bordoloi et al. (2014) are pessimistic.

It is outside the scope of this paper to repeat the entire algorithms presented by Axer et al. (2014) and Bordoloi et al. (2014), but for completeness we must mention a small adjustment that is needed before we can compare the work of Bordoloi et al. (2014) to ours and that of Axer et al. (2014). We observe that the definition of worst-case response-time in Bordoloi et al. (2014) includes the recovery time after the transmission of interest, while in our approach and that of Axer et al. (2014), the response-time ends immediately

Table 5 Example scenario, calculating worst-case response-times for class M sources τ_1 , τ_2 and τ_3 using the approaches in Axer et al. (2014) and Bordoloi et al. (2014), given a total bandwidth of 100 Mbps

	P	α^+ Mbps	C μs	C_i μs	T_i μs	D_i μs	WR (Axer) μs	WR (Bordoloi) (μs)
τ_4	H	40	1	1	10	10	-	-
τ_5				1	5	10	-	-
τ_1	M	40	3	1	25	-	23.5	18.5
τ_2				3	30	-	17.5	16.5
τ_3				2	20	-	20	19
-	L	-	2	-	-	-	-	-

The parameters on priority classes are in grey

after the transmission. In order to provide a fair comparison, we have (straightforwardly) adapted the Eqs. (20) and (21) in Bordoloi et al. (2014) to reflect this. Also, in the second improvement of Bordoloi et al. (2014), the calculation of medium-priority recovery time introduces unnecessary pessimism, and we adapt it according to Axer et al. (2014). Furthermore, in Axer et al. (2014), Eq. (18) contains a term I_{HPB} , used to bound the time during which credit of a shaper can build up before a burst. The exact way in which the authors calculate I_{HPB} , however, remains unclear from the paper. Therefore, we decided to use our own estimate of the maximum attainable credit instead.

Table 5 contains the results of applying the methods from Axer et al. (2014) and Bordoloi et al. (2014) to our illustrating example. We have refined the information on high-priority tasks, by distinguishing two sources τ_4 and τ_5 , and we have added deadlines for both tasks. It can be noted immediately, that despite this added information, the results for this particular example are pessimistic compared to those in Table 4. This is surprising, because the worst-case scenario that goes with this example, displayed in Fig. 11, shows that the burst of high-priority behavior is too small to fully deplete the credit that has been built up. The maximum credit for M is not reached and the worst-case response time predicted by our approach is thus not attained, because there is not enough flexibility to generate the necessary worst-case interference. Next, we set out to identify the sources of the pessimism in Axer et al. (2014) and Bordoloi et al. (2014) that cause the over-estimation of the worst-case response time, and we suggest a solution to improve our own prediction in case details on the limited burst-size are known or computable.

6.3 Exploring pessimism

The busy period analyses of Axer et al. (2014) and Bordoloi et al. (2014) start out from the same basic analysis. Both papers initially identify a model in which four sources of interference are identified: high-priority interference, low-priority interference, medium-priority interference, and interference due to shaping. Incidentally this basic analysis in the two papers coincides for periodic sources, and for reference we have added the results

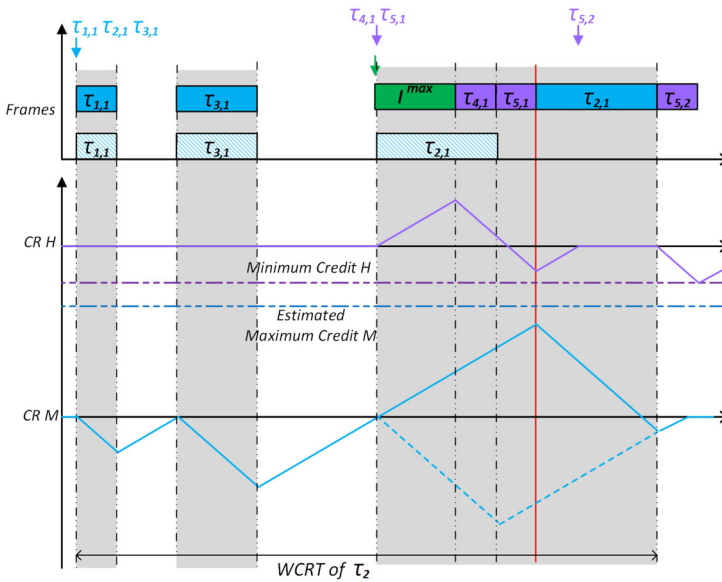


Fig. 11 Worst-case scenario for τ_2 when adding knowledge regarding periodic interference (see Table 5). A maximum low-priority frame is released just before transmission of τ_2 is enabled, with subsequent arrivals of frames from high-priority τ_4 and τ_5 . Note how these two frames are insufficient to deplete the credit of H to its theoretical minimum. Therefore, a maximum built-up of credit of M is prevented, causing the transmission of τ_2 to start earlier than estimated

of this analysis (under the name ‘basic busy period analysis’) in the figures that occur in this section.

Both Axer et al. (2014) and Bordoloi et al. (2014) start by observing that the traditional calculation of high-priority interference, adding up all arrivals during a busy period, disregards the fact that the high priority is shaped. This source of pessimism is taken care of by adding a burst-rate model for the shaper.

In Fig. 12, we illustrate the influence of this improvement, by considering the same shaper reservations and maximum frame sizes as in the grey parts in Table 4. We determine the worst-case response time of a single medium-priority source with $C = 3$, and $T = 30$, interfered by n_H identical high-priority sources with $C_i = 1$, $T_i = D_i = 5 \cdot n_H$, as well as a low-priority stream. Note, that we keep the total utilization of H at 20 Mbps (below its reservation α_H^+) as n_H ranges from 1 to 20.

The result of varying the number of high priority sources, is that the possible burst size of high priority traffic increases. The figure clearly shows how, for low values of n_H , the burst size is smaller than the limits set by the credit based shaper. For those low values, the predictions of Axer et al. (2014) and Bordoloi et al. (2014) turn out to be better than ours, because we do not consider the load generated by high priority sources. For higher values of n_H , the high priority shaping becomes dominant, and those of Axer et al. (2014) and Bordoloi et al. (2014) eventually stabilize at a fixed value, while the basic busy period analysis keeps growing because it does not take high priority shaping

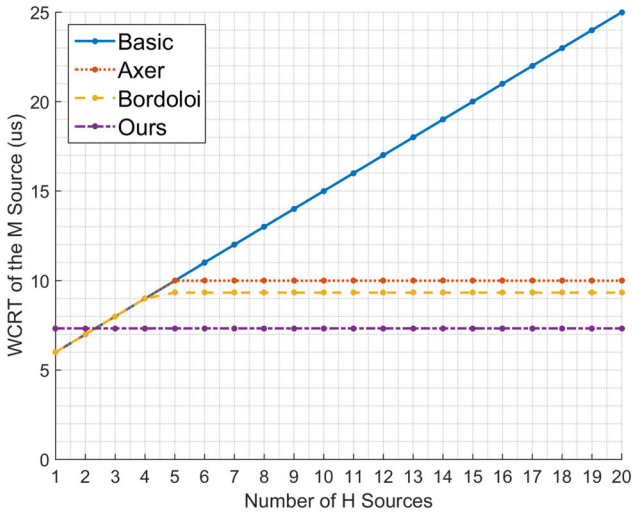


Fig. 12 Worst-case response time of a medium-priority source given interference by n_H identical high-priority sources and a low-priority stream

into account at all. Our approach always remains the same value, which turns out to be tight and independent of the pattern of high priority traffic.

The reason why Bordoloi et al. (2014) performs better than Axer et al. (2014) in Fig. 12, is because of a second cause of pessimism, which was not addressed in Axer et al. (2014). This cause of pessimism was already briefly discussed in the introduction, when we mentioned that it is complicated (to say the least) to apply busy period analysis to idling servers. The four types of interference considered in the basic busy period analysis turn out not to be independent. If part of the high priority traffic transmits during a recovery period of the medium priority shaper, then this high priority traffic should actually not be counted as interference. This is because it does not contribute to the busy period. The authors of Bordoloi et al. (2014) realised this and have made an attempt to take this effect in to account.

This is illustrated in Fig. 13, in which we again consider the same shaper reservations and maximum frame sizes as in Table 4. As before, we determine the worst-case response time of a single medium-priority source with $C = 3$, and $T = 30$, but this time the interference consists of a low-priority stream, a single high-priority source with $C = 1$, and $T = D = 5$, and n_M medium-priority sources with $C_i = 1$, $T_i = D_i = 5 \cdot n_M$.

By increasing the number of medium priority tasks, we increase the recovery time as well. Furthermore, by choosing a single high priority task with relatively small frame size, we ensure that the maximally built-up credit cannot be fully depleted. As a consequence, the approach of Bordoloi et al. (2014) is consistently better than ours in Fig. 13, while the approach of Axer et al. (2014) coincides with the basic busy period analysis and becomes more and more pessimistic as n_M grows.

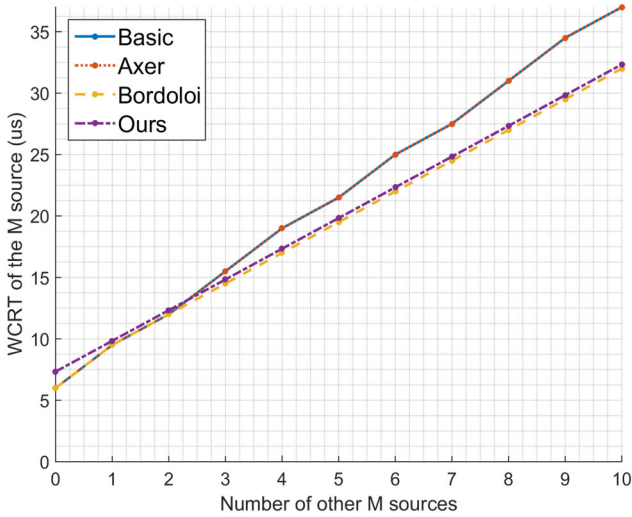


Fig. 13 Worst-case response time of a medium-priority source given interference by 1 high-priority source, n_M medium priority sources, and a low-priority stream

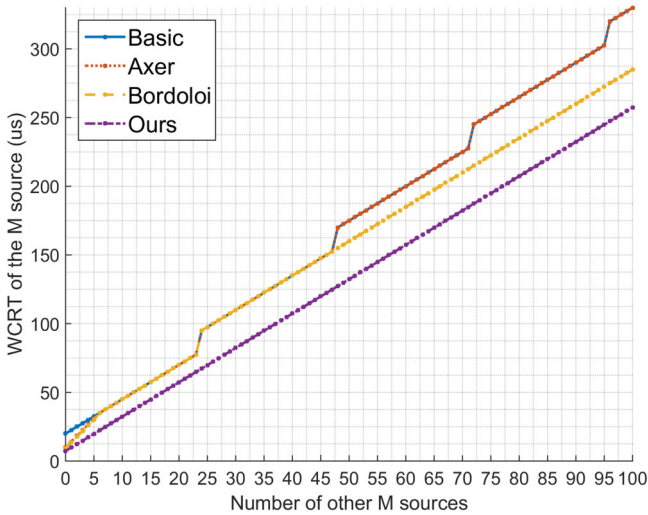


Fig. 14 Worst-case response time of a medium-priority source given interference by 15 high-priority sources, n_M medium priority sources, and a low-priority stream

Finally, if we adapt the scenario in which there are still n_M interfering medium priority sources, and consider $n_H = 15$ identical high priority sources instead of just 1, we see a completely different picture, shown in Fig. 14. Now, our approach is tight again, because there is sufficient high priority traffic to deplete the credit to reach its minimum. Furthermore, if we range n_M from 1 to 100, we see that the approaches of Axer et al. (2014) and Bordoloi et al. (2014) give better estimates than the basic analysis for low

values ($n_M \leq 6$), since the shaping of high-priority is taken into account. For high values, the approach of Axer et al. (2014) coincides with the basic analysis, and shows a ‘staircase’ behavior in its pessimism which indicates that the arrival of high priority interference is the limiting factor in the analysis. Interestingly, the analysis from Bordoloi et al. (2014) suffers from the same over approximation until $n_M = 47$, and only then merits from the implemented improvements.

The complexity of the approach set out in Bordoloi et al. (2014) makes it difficult to find a satisfactory explanation for the pessimism that still is present. Instead, it may be more promising to consider whether our own approach can be improved in cases where the burstiness of high priority traffic is insufficient to generate the worst-case behavior predicted by our approach.

6.4 Adding information

From the comparison so far, we conjecture that the only pessimism that remains in our approach occurs when the high priority traffic cannot produce a sufficiently large burst. Using an iterative approach similar to busy period analysis, we can calculate the maximum time *Burst* during which there can be a continuous transmission of high priority traffic. Considering the scenario in Fig. 11, we claim that, given knowledge of this burst, our worst-case response-time analysis can be adapted to:

$$WR_{\mathbb{U}}(\tau_i) = WR_{\mathbb{U}_0}(\tau_i) + \min \left(C_L^{\max} \cdot \left(1 + \frac{\alpha_H^+}{\alpha_H^-} \right) + C_H^{\max}, C_L^{\max} + Burst \right).$$

Obviously, the burstiness only makes a difference if *Burst* is smaller than time needed to deplete the built-up credit plus a single maximum high priority frame transmission.

Naturally, one can go even further, and consider the fact that the last transmission must be a frame of maximum size, and that the transmissions preceding this must exactly fit the depletion time of the maximum credit. If those transmissions do not fit exactly (for instance, because frames have a fixed size) the worst-case can also not be attained. But such considerations only complicate the analysis, and are expected to lead to relatively small improvements compared to the results already achieved by our eligible interval analysis.

7 Conclusions and future work

In this paper, we analyzed the worst-case response time of transmitting frames under credit-based shaping in an Ethernet AVB switch. We improved our work of Cao et al. (2016a, b), extending its application to the case where there are multiple higher and lower priority classes that interfere with the transmission of a frame. We presented a technique to perform worst-case response time analysis, independent of knowledge of

the interfering traffic (except those assumptions enforced by the Ethernet AVB standard), and proved sufficient (and conjectured necessary) conditions under which our analysis yields a tight bound. Notably, the complexity of our worst-case relative delay analysis mainly depends on the complexity of the minimum credit calculation, which is only factorial in the number of higher priority classes (see Theorem 4), and the complexity of checking the sufficient conditions just involves an additional inequality check. Unlike the more traditional busy period analysis, that was proposed in, amongst others, Axer et al. (2014) and Bordoloi et al. (2014), our method does not rely on recursions of which the depth is dependent on the chosen system parameters.

Our comparison with the state-of-the-art [e.g. Axer et al. (2014) and Bordoloi et al. (2014)], focussing on single higher and lower priority streams, suggests that our approach is superior in situations where the burstiness of high priority traffic is large enough. In particular, our approach is superior when the burst is larger than the depletion time of the maximum credit built up by the low priority traffic plus a single maximum high priority frame transmission. Furthermore, given the input models assumed in Axer et al. (2014) or Bordoloi et al. (2014), we conjecture it is fairly easy to estimate the maximum burst size and improve our analysis slightly. This would give an equivalent result to the state of the art. However, such an improvement requires additional assumptions on the sources of interference, which we think is unreasonable considering the increasing number of connected components in a network. Furthermore, the improvements would complicate our algorithm.

We attribute the tightness as well as the simplicity of our approach to the use of eligible intervals, during which there is both pending load and credit available to a shaper. The use of these intervals leads to a separation of concerns in which we first relate eligible intervals to a known un-interfered schedule, and secondly relate the credit at the start of a transmission to the relative delay caused by the interfering traffic. Subsequently, we bound the relative delay by bounding the maximum attainable credit. The main contribution of this paper, is that we show that this approach is still feasible in case of multiple (shaped) priority classes.

As a topic for future research, we intend to investigate whether eligible intervals can be applied in the analysis of other shaping strategies as well. For example, we aim at applying them in the context of the Ethernet TSN standard (IEEE 2012), which is currently being developed. In Ethernet TSN other shapers are considered in combination with credit-based shaping to guarantee low latencies for control frames, adding another source of heterogeneity to the analysis. It is interesting to see whether the use of eligible intervals has advantages in the analysis of such shapers in isolation, but also to see whether it can be used when different shapers are being combined. As a further topic of investigation, we would like to see if the Compositional Performance Analysis approach that was used in Axer et al. (2014) can be simplified using eligible intervals. In principle, if our eligible intervals can indeed be used to transform arrival curves into output curves, the study of multi-hop scenarios of idling servers becomes much easier.

Acknowledgements This research is supported by the Dutch Technology Foundation STW under the Robust CPS Program, Project 12697.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

Appendix: a proof of tightness

In this appendix, we set out to prove Theorem 7 in more detail. In other words, we prove conditions for tightness of the relative worst-case response-time bound given in Theorem 6. This is done by showing that for any margin $\epsilon > 0$ there exists an execution u^ϵ in which some frame in M is delayed by more than

$$C_{\mathbb{L}}^{\max} \cdot \left(1 + \frac{\alpha_{\mathbb{H}}^+}{\alpha_{\mathbb{H}}^-} \right) - \frac{CR_{\mathbb{H}}^{\min}}{\alpha_{\mathbb{H}}^-} - \epsilon.$$

compared to its un-interfered schedule u_0^ϵ . The construction of this execution u^ϵ consists of five parts, illustrated in Fig. 10, and is worked-out using a series of lemmas, addressing the different phases in isolation. Incidentally, this construction also serves as a proof-sketch that the axioms we assumed to describe the operation of credit-based shaping are not self-contradictory.

Throughout this appendix, we consider a given stream of priority class $M \in \mathbb{P}$. We use $\mathbb{H} = \{X \in \mathbb{P} \mid X > M\}$ to denote all streams of higher priority and $\mathbb{L} = \{X \in \mathbb{P} \mid X < M\}$ to denote all streams of lower priority. Furthermore, we always assume that M and \mathbb{H} are under credit based shaping, and $\alpha_{\mathbb{H}}^+ + \alpha_M^+ \leq BW$. Finally, we assume an arbitrarily small value $\epsilon > 0$, by which we strive to approximate the worst-case response time. We construct u^ϵ by defining at which point in time, which events occur in which order.

Lower priority interference (phase 1) Construct an execution u^ϵ such that at time $t_0 = 0$ a lower priority frame $l \in \mathbb{L}$ of maximum size $C^{u^\epsilon}(l) = C_{\mathbb{L}}^{\max}$ arrives, immediately followed by the arrival of a frame $m \in \mathbb{M}$ and a burst B of higher priority frames, such that for each $H \in \mathbb{H}$ there is an $h \in B$ with $h \in H$. Then, at time $t_1 = C_{\mathbb{L}}^{\max}$, only the lower priority frame l has transmitted, and the credit of all $X \in M \cup \mathbb{H}$ will be given by:

$$CR_X^{u^\epsilon}(t_1) = \alpha_X^+ \cdot C_{\mathbb{L}}^{\max}$$

and the cumulative credit of all higher-priority shapers is given by

$$CR_{\mathbb{H}}^{u^\epsilon}(t_1) = \alpha_{\mathbb{H}}^+ \cdot C_{\mathbb{L}}^{\max}$$

Depletion (phase 2) Next, assuming the burst B is large enough, observe that at time

$$t_2 = t_1 + \frac{\alpha_{\mathbb{H}}^+}{\alpha_{\mathbb{H}}^-} C_{\mathbb{L}}^{\max}$$

we find that the cumulative credit has decreased to zero, i.e.

$$CR_{\mathbb{H}}^{u^\epsilon}(t_2) = 0.$$

The reason for this, is that given that cumulative credit of higher priority shapers is non-negative, and given sufficient pending load, at least one higher priority shaper has both non-negative credit and pending load and is therefore eligible for transmission. As a consequence, higher priority transmissions will continue at least until cumulative credit drops below 0.

Now, if we consider a burst B that consists of frames that all have size smaller than $\delta > 0$. Then for each higher priority shaper $H \in \mathbb{H}$ we find:

$$-\alpha_H^- \delta \leq CR_H^{u^\epsilon}(t_2) \leq \sum_{X \in \mathbb{H}} \alpha_X^- \delta$$

Furthermore, because cumulative credit is 0 at t_2 , there exists some higher priority shaper Y with $CR_Y^{u^\epsilon}(t_2) \geq 0$.

Reset (phase 3) Next, we adapt the burst B a bit. We remove all frames from B that would transmit after t_2 , and add a frame of size $\eta_H = \frac{CR_H^{u^\epsilon}(t_2)}{\alpha_H^-}$ for every priority class $H \in \mathbb{H}$ that has positive credit at time t_2 . This will not change the execution upto time t_2 , but ensures that time t_3 , taken to be the finishing time of the last η_H frame in the burst, satisfies

$$t_2 \leq t_3 \leq t_2 + \sum_{Y \in \mathbb{H}, CR_Y^{u^\epsilon}(t_2) \geq 0} \frac{\sum_{X \in \mathbb{H}} \alpha_X^- \cdot \delta}{\alpha_Y^-}$$

At t_3 there is now no more pending load from higher priority frames. Furthermore, all priority classes $H \in \mathbb{H}$ satisfy

$$-\alpha_H^- \cdot \delta \leq CR_H^{u^\epsilon}(t_3) \leq 0$$

because the size of η_H ensures that there is a reset of credit due to lack of pending load at the end of each transmission. Note that for one priority class $Y \in \mathbb{H}$, namely the η_Y transmission that occurs first, credit actually reaches 0 without a reset, because the size of η_Y is chosen to exactly deplete credit.

Connection (phase 4) In the fourth phase, we try to make a connection to the fifth phase, which requires a little look-ahead. The challenge in the fifth phase is that at its start, all

credits except one have a particular negative value. The fourth phase intends to create this starting credit.

So, consider a function $f : \mathbb{H} \rightarrow \mathbb{R}^+$ and a value $\zeta > 0$. Observe that we can then create a sequence of higher priority classes $H_1 \dots H_N$ in which each priority class of \mathbb{H} occurs exactly once and $H_1 = Y$ represents the priority classes Y that reaches a credit of 0 by depleting it exactly to 0. I.e. η_Y is the first of the η frames to transmit in phase 3. Subsequently, construct a sequence σ_i such that:

$$\sigma_i = \frac{-f(H_i) \cdot \zeta + \alpha_{H_i}^+ \cdot \sum_{i < j \leq N} \sigma_j}{\alpha_{H_i}^-}$$

and extend the execution u^ϵ with an arrival of a frame of priority H_i and size σ_i at (or just before) time

$$t_4^i = t_3 + \sum_{1 \leq j < i} \sigma_j.$$

Then, at time $t_4 = t_4^N$ we find by construction:

$$CR_{H_i}^{u^\epsilon}(t_4) = -f(H_i) \cdot \zeta$$

assuming ζ is chosen sufficiently small, and δ is chosen much smaller than ζ .

To see why this construction works, notice that at t_3 we find that Y is eligible. It has just received a pending load of σ_1 and is able to transmit. Furthermore, its previous transmission reaches 0 credit (instead of resulting in a reset), so the arrival of σ_1 does not change the evolution of credit constructed in phase 3. The frame σ_1 from Y will transmit, and after transmission, by construction, will be recovering until time t_4 , reaching the intended credit. After Y has transmitted, all other shapers that had negative credit will have recovered, assuming that the value of δ is sufficiently small. Their credit will remain 0 because none of those shapers have pending load. At the end of the transmission of Y , the next shaper in the sequence $H_1 \dots H_N$ will obtain pending load and transmit. By construction, this transmission is chosen such that the desired credit at t_4 is obtained. Naturally, for this construction to be legal, ζ must be chosen small enough to ensure none of the required frames exceeds the maximum frame size. In the construction of the final scenario, we therefore must determine ζ before determining δ .

Slow descend (phase 5) The fifth phase is aimed at maintaining a slow descend while aiming at a lowest possible value of cumulative credit in the end. We know from Corollary 2 that a slowest descend is attained if all credits are recovering except for the one that is transmitting. In order to get a credit that is as low as possible, we reason backwards, starting at the end of the scenario. We are attempting to create a worst-case scenario, where at its end (time t_5) the transmission of the medium priority frame starts. We therefore know that at t_5 credit of all higher-priority shapers is negative, otherwise another higher-priority interference would have been possible.

Now, assume shaper H performed the last transmission, ending at t_5 , then at the start of H , credit of all other shapers must have been lower than at t_5 , because all those shapers are in recovery. We need the execution we are constructing to be a valid credit-based schedule. Therefore, we must take care that the credits of individual shapers do not drop below their minimum values (i.e. $-\alpha_H^- \cdot C_H^{\max}$). In the construction of this phase, we start at the end, and prefix smaller and smaller frames, aiming to avoid such credit violations. We know that cumulative credit will always be decreasing throughout a valid execution, and aim to carry out this decrease for as long as possible. In the final paragraph of this appendix we study under which conditions the construction in phase 5 actually attains minimum credit.

Since we intend to prefix smaller and smaller frame sizes until they are arbitrarily small (of magnitude ζ as aimed at in the previous phase), we need some way to construct an infinite sequence. This we do by iterating over a sequence $H_1 \dots H_N$ in which each higher priority class occurs once. Unlike in the previous phase, this sequence does not have a designated first priority class, but instead it should satisfy:

$$\frac{C_{H_i}^{\max}}{\alpha_{H_i}^+} \geq \frac{C_{H_{i+1}}^{\max}}{\alpha_{H_{i+1}}^+}$$

for all $1 \leq i < N$. Note that such a sequence can always be constructed, simply by ordering the priorities in \mathbb{H} according to the ratio between maximum frame size and recovery rate.

A second observation, if we are to build a worst-case execution, is that the last transmission should have maximum frame size. If it does not, a larger delay would have been possible, resulting in a worse case. Therefore, the last transmission before t_5 will have a maximum frame size. Given the sequence $H_1 \dots H_N$, we define:

$$\sigma_N^0 = C_{H_N}^{\max}$$

Subsequently, the one-but-last transmission H_{N-1} needs to fit three requirements. Firstly, it should be maximal size if possible. Secondly, if it is chosen too large, credit of H_N may drop below its individual minimum. And thirdly, it should be a strictly positive value. To guarantee that it is a strictly positive value, we assume a margin $\mu > 0$ which can be taken arbitrarily small, and obtain

$$\sigma_{N-1}^0 = \min \left\{ C_{H_{N-1}}^{\max}, \frac{\alpha_{H_N}^-}{\alpha_{H_N}^+} (C_N^{\max} - \mu) \right\}$$

For an arbitrary transmission in the first sequence $H_1 \dots H_N$, we need to verify that all shapers that transmit later do not drop below their minimum value. Iterating the previous construction over the entire sequence $H_1 \dots H_N$ we find:

$$\sigma_i^0 = \min \left\{ C_{H_i}^{\max}, \inf_{i < k \leq N} \frac{\alpha_{H_k}^-}{\alpha_{H_k}^+} (C_{H_k}^{\max} - \mu) - \sum_{i < j < k} \sigma_j^0 \right\}$$

The values σ_i^0 now represent the sizes of the last transmissions in phase 5 in Fig. 10. The transmissions that are prefixed are determined completely by the requirement that they recover exactly at the start-times of the next frame. So, reasoning backwards along the execution we find:

$$\sigma_i^{k+1} = \frac{\alpha_{H_i}^+}{\alpha_{H_i}^-} \left(\sum_{1 \leq j < i} \sigma_j^k + \sum_{i < j \leq N} \sigma_j^{k+1} \right)$$

With the arrival times of the frames simply given by

$$t_i^k = t_5 - \left(\sum_{i \leq j \leq N} \sigma_j^k \right) - \left(\sum_{1 \leq j \leq N} \sum_{l < k} \sigma_j^l \right)$$

It is clear by construction that for each of the priority classes H_1 the time of recovery of the previous transmission coincides with the transmission of the next frame by that priority class. To guarantee that the sequence leads to a valid scenario, we therefore only need to prove that frame sizes are strictly positive and never larger than maximum frame sizes. This follows from the following induction:

We prove by induction that

$$\sigma_j^k < \frac{\alpha_{H_j}^+}{\alpha_{H_i}^+} C_{H_i}^{\max}$$

for all k and all $1 \leq j \leq i \leq N$, so in particular, $\sigma_i^k < C_{H_i}^{\max}$.

As a base-case, assuming $j = i$ and $k = 0$, the proof is trivial.

As a second base case, assume $j < i$ and $k = 0$. Then:

$$\begin{aligned} \sigma_j^0 &= \min \left\{ C_{H_j}^{\max}, \inf_{j < n \leq N} \frac{\alpha_{H_n}^-}{\alpha_{H_n}^+} (C_{H_n}^{\max} - \mu) - \sum_{j < s < n} \sigma_s^0 \right\} \\ &\leq \inf_{j < n \leq N} \frac{\alpha_{H_n}^-}{\alpha_{H_n}^+} C_{H_n}^{\max} - \sum_{j < s < n} \sigma_s^0 \\ &\leq \frac{\alpha_{H_i}^-}{\alpha_{H_i}^+} C_{H_i}^{\max} - \sum_{j < s < i} \sigma_s^0 \\ &\leq \frac{\alpha_{H_i}^-}{\alpha_{H_i}^+} C_{H_i}^{\max} \end{aligned}$$

$$\begin{aligned}
 &= \frac{BW - \alpha_{H_i}^+}{\alpha_{H_i}^+} C_{H_i}^{\max} \\
 &< \frac{\alpha_{H_j}^+}{\alpha_{H_i}^+} C_{H_i}^{\max}
 \end{aligned}$$

using the assumption that $\alpha_{H_j}^+ + \alpha_{H_i}^+ < BW$.

Finally, assume $\sigma_p^q < \frac{\alpha_{H_p}^+}{\alpha_{H_r}^+} C_{H_r}^{\max}$ whenever $q = k + 1$ and $i < p \leq r$, or $q \leq k$ and $p \leq r$ then

$$\begin{aligned}
 \sigma_j^{k+1} &= \frac{\alpha_{H_j}^+}{\alpha_{H_j}^-} \left(\sum_{1 \leq n < j} \sigma_n^k + \sum_{j < n \leq N} \sigma_n^{k+1} \right) \\
 &= \frac{\alpha_{H_j}^+}{\alpha_{H_j}^-} \left(\sum_{1 \leq n < j} \sigma_n^k + \sum_{j < n \leq i} \sigma_n^{k+1} + \sum_{i < n \leq N} \sigma_n^{k+1} \right) \\
 &\leq \frac{\alpha_{H_j}^+}{\alpha_{H_j}^-} \left(\sum_{1 \leq n < j} \frac{\alpha_{H_n}^+}{\alpha_{H_i}^+} C_{H_i}^{\max} + \sum_{j < n \leq i} \frac{\alpha_{H_n}^+}{\alpha_{H_i}^+} C_{H_i}^{\max} + \sum_{i < n \leq N} C_{H_n}^{\max} \right) \\
 &\leq \frac{\alpha_{H_j}^+}{\alpha_{H_j}^-} \left(\sum_{1 \leq n < j} \frac{\alpha_{H_n}^+}{\alpha_{H_i}^+} C_{H_i}^{\max} + \sum_{j < n \leq i} \frac{\alpha_{H_n}^+}{\alpha_{H_i}^+} C_{H_i}^{\max} + \sum_{i < n \leq N} \frac{\alpha_{H_n}^+}{\alpha_{H_i}^+} C_{H_i}^{\max} \right) \\
 &= \frac{\sum_{n \neq j} \alpha_{H_n}^+ \alpha_{H_j}^+}{\alpha_{H_j}^- \alpha_{H_i}^+} C_{H_i}^{\max} \\
 &\leq \frac{BW - \alpha_{H_j}^+}{\alpha_{H_j}^-} \frac{\alpha_{H_j}^+}{\alpha_{H_i}^+} C_{H_i}^{\max} \\
 &= \frac{\alpha_{H_j}^+}{\alpha_{H_i}^+} C_{H_i}^{\max}
 \end{aligned}$$

Note, that in the fourth line we use the assumption that $H_1 \dots H_n$ are ordered such that

$$\frac{C_{H_i}^{\max}}{\alpha_{H_i}^+} \geq \frac{C_{H_{i+1}}^{\max}}{\alpha_{H_{i+1}}^+},$$

and the last lines again rely on the fact that the total reservation does not exceed the bandwidth.

To connect the sequence to phase four, we need to find a starting point for the fifth phase. One should notice that credit is decreasing by the slowest rate, by construction of the fifth phase. As a consequence, cumulative credit approaches zero when going from t_5 towards t_4 . Furthermore, by construction credit remains negative at all times. For every

$\zeta > 0$ there is a k such that $\sigma_i^k \leq \zeta$ for all $1 \leq i \leq N$. Therefore, by choosing a sufficiently large k , we can use phase four to connect to phase five.

Response time of u^ϵ The scenario u^ϵ maintains a slowest decrease of credit at all times during phase 2 and phase 5. Furthermore, phase 3 and phase 4 have an arbitrarily small duration, and during phase 4 an arbitrarily small drop in credit occurs due to resets. At all times there is a higher priority shaper eligible, which means that the frame $m \in M$ cannot transmit until u^ϵ is over. From this we conclude that, during u^ϵ , m experiences a response time arbitrarily close to $t_2 + t_5 - t_4$. Furthermore, $t_5 - t_4$ is arbitrarily close to the cumulative credit achieved at t_5 divided by the slowest credit slope $\alpha_{\mathbb{H}}^-$.

Obviously, the credit achieved at the end of phase 5 is completely determined by its last sequence of transmissions $H_1 \dots H_N$. We calculate:

$$CR_{\mathbb{H}}^{u^\epsilon}(t_5) = - \sum_{1 \leq i \leq N} \alpha_{H_i}^- \cdot \sigma_i^0 - \alpha_{H_i}^+ \cdot \sum_{i < j \leq N} \sigma_j^0$$

and conclude that the scenario u^ϵ generates a response time of

$$t_5 - t_0 \approx \left(1 + \frac{\alpha_{\mathbb{H}}^+}{\alpha_{\mathbb{H}}^-} \right) C_{\mathbb{L}}^{\max} + \frac{\sum_{1 \leq i \leq N} \left(\alpha_{H_i}^- \sigma_i^0 - \alpha_{H_i}^+ \sum_{i < j \leq N} \sigma_j^0 \right)}{\alpha_{\mathbb{H}}^-}$$

Achieving the worst-case response time Finally, we observe that one way to pick a sequence $H_1 \dots H_N$ is by following the construction of minimal credit. Given the definition of minimal credit, it is always possible to construct a sequence $H_1 \dots H_N$ such that for all i we find:

$$CR_{\bigcup_{1 \leq i \leq n} H_i}^{\min} = -\alpha_{\bigcup_{1 \leq i \leq n} H_n}^- \cdot C_{H_n} + CR_{\bigcup_{1 \leq i < n} H_i}^{\min}$$

One may verify that such a sequence also satisfies such that

$$\frac{C_{H_i}^{\max}}{\alpha_{H_1}^+} \geq \frac{C_{H_{i+1}}^{\max}}{\alpha_{H_{i+1}}^+}$$

holds for all i , and therefore can serve as a basis for our construction of u^ϵ .

Finally, if for this sequence we find

$$C_{H_N}^{\max} \geq \frac{\alpha_{H_N}^+}{\alpha_{H_N}^-} \sum_{1 \leq j < N} C_{H_j}^{\max}$$

then it is easy to verify that the credit attained at t_5 is in fact the minimum credit. As a consequence, if the sequence based on minimal credit satisfies this property, we have a witness showing our worst-case response time estimate in Theorem 6 is tight.

References

- Ashjaei M, Patti G, Behnam M, Nolte T, Alderisi G, Bello LL (2017) Schedulability analysis of Ethernet audio video bridging networks with scheduled traffic support. *Real-Time Syst* 53(4):526–577. <https://doi.org/10.1007/s11241-017-9268-5>
- Axer P, Thiele D, Ernst R, Diemer J (2014) Exploiting shaper context to improve performance bounds of Ethernet AVB networks. In: Proceedings of the 51st ACM/EDAC/IEEE design automation conference (DAC), pp 1–6. <https://doi.org/10.1145/2593069.2593136>
- Bordoloi UD, Aminifar A, Eles P, Peng Z (2014) Schedulability analysis of Ethernet AVB switches. In: Proceedings of the 20th IEEE international conference on embedded and real-time computing systems and applications (RTCSA), pp 1–10. <https://doi.org/10.1109/RTCSA.2014.6910530>
- Bril RJ, Lukkien JJ, Verhaegh WFJ (2009) Worst-case response time analysis of real-time tasks under fixed-priority scheduling with deferred preemption. *Real-Time Syst* 42(1):63–119. <https://doi.org/10.1007/s11241-009-9071-z>
- Cao J, Cuijpers PJJ, Bril RJ, Lukkien JJ (2016a) Independent yet tight WCRT analysis for individual priority classes in Ethernet AVB. In: Proceedings of the 24th international conference on real-time networks and systems (RTNS), pp 55–64. <https://doi.org/10.1145/2997465.2997493>
- Cao J, Cuijpers PJJ, Bril RJ, Lukkien JJ (2016b) Tight worst-case response-time analysis for Ethernet AVB using eligible intervals. In: Proceedings of the 12th IEEE world conference on factory communication systems (WFCS), pp 1–8. <https://doi.org/10.1109/WFCS.2016.7496507>
- Davis RI, Burns A, Bril RJ, Lukkien JJ (2007) Controller area network (CAN) schedulability analysis: refuted, revisited and revised. *Real-Time Syst* 35(3):239–272. <https://doi.org/10.1007/s11241-007-9012-7>
- Diemer J, Rox J, Ernst R (2012a) Modeling of Ethernet AVB networks for worst-case timing analysis. In: Proceedings of the 7th Vienna international conference on mathematical modelling (MATHMOD), pp 848–853. <https://doi.org/10.3182/20120215-3-AT-3016.00150>
- Diemer J, Thiele D, Ernst R (2012b) Formal worst-case timing analysis of Ethernet topologies with strict-priority and AVB switching. In: Proceedings of the 7th IEEE international symposium on industrial embedded systems (SIES), pp 1–10. <https://doi.org/10.1109/SIES.2012.6356564>
- Henia R, Hamann A, Jersak M, Racu R, Richter K, Ernst R (2005) System level performance analysis—the SymTA/S approach. *IEEE Proc Comput Digit Tech* 152(2):148–166. <https://doi.org/10.1049/ip-cdt:20045088>
- IEEE (2005) IEEE 802.1 audio/video bridging task group. <http://www.ieee802.org/1/pages/avbridges.html>
- IEEE (2012) IEEE 802.1 time-sensitive networking task group. <http://www.ieee802.org/1/pages/tsn.html>
- IEEE (2014) IEEE Std. 802.1Q-2014, standard for local and metropolitan area networks, bridges and bridged networks
- IEEE (2016) IEEE Standard 802.1Qbv, standard for local and metropolitan area networks bridges and bridged networks amendment 25: enhancements for scheduled traffic
- Imtiaz J, Jasperneite J, Han L (2009) A performance study of Ethernet audio video bridging (AVB) for industrial real-time communication. In: Proceedings of the 14th IEEE conference on emerging technologies factory automation (ETFA), pp 1–8. <https://doi.org/10.1109/ETFA.2009.5347126>
- Lehoczky J (1990) Fixed priority scheduling of periodic task sets with arbitrary deadlines. In: Proceedings of the 11th real-time systems symposium (RTSS), pp 201–209. <https://doi.org/10.1109/REAL.1990.128748>
- Li X, George L (2017) Deterministic delay analysis of AVB switched ethernet networks using an extended trajectory approach. *Real-Time Syst* 53(1):121–186. <https://doi.org/10.1007/s11241-016-9260-5>
- Martin S, Minet P (2006) Schedulability analysis of flows scheduled with FIFO: application to the expedited forwarding class. In: Proceedings of the 20th IEEE international parallel distributed processing symposium (IPDPS). <https://doi.org/10.1109/IPDPS.2006.1639424>
- Maxim D, Song YQ (2017) Delay analysis of AVB traffic in time-sensitive networks (TSN). In: Proceedings of the 25th international conference on real-time networks and systems (RTNS), pp 18–27. <https://doi.org/10.1145/3139258.3139283>
- Reimann F, Graf S, Streit F, Glas M, Teich J (2013) Timing analysis of Ethernet AVB-based automotive E/E architectures. In: Proceedings of the 18th IEEE conference on emerging technologies factory automation (ETFA), pp 1–8. <https://doi.org/10.1109/ETFA.2013.6648024>

- Thangamuthu S, Concer N, Cuijpers PJL, Lukkien JJ (2015) Analysis of Ethernet-switch traffic shapers for in-vehicle networking applications. In: Proceedings of the design, automation & test in Europe conference & exhibition (DATE 2015). Piscataway, IEEE Service Center, Grenoble, pp 55–60
- Thiele D, Ernst R (2016) Formal worst-case timing analysis of Ethernet TSN's burst-limiting shaper. In: Proceedings of the design, automation test in Europe conference exhibition (DATE), pp 187–192. https://doi.org/10.3850/9783981537079_0276
- Thiele D, Ernst R, Diemer J (2015) Formal worst-case timing analysis of Ethernet TSN's time-aware and peristaltic shapers. In: Proceedings of the IEEE vehicular networking conference (VNC), pp 251–258. <https://doi.org/10.1109/VNC.2015.7385584>
- Wandeler E, Thiele L, Verhoef M, Lieverse P (2006) System architecture evaluation using modular performance analysis: a case study. *Int J Softw Tools Technol Transf* 8(6):649–667. <https://doi.org/10.1007/s10009-006-0019-5>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Jingyue Cao is a Ph.D. candidate at the System Architecture and Networks (SAN) group at the Eindhoven University of Technology (TU/e). She is currently conducting her research in the project Control based on Data-Intensive Sensing. This project is part of the program on Robust Design of Cyber Physical Systems (rCPS). It aims at providing a multi-disciplinary design approach for complex and data-intensive control problems.



Pieter J. L. Cuijpers is an assistant professor in the System Architecture and Networking Research group at Eindhoven University of Technology (TU/e). After developing a process algebra for hybrid systems (HyPA), his current research interests include the application of quantitative formal methods to cyber physical systems, with a focus on performance analysis and scheduling of distributed embedded systems. Important techniques that play a role in this are: max-plus algebra and dataflow modeling, event-based modeling and simulation and axiomatic reasoning about execution intervals.



Reinder J. Bril is an associate professor and chair of Real-Time Systems in the Department of Mathematics and Computer Science at Eindhoven University of Technology (TU/e) and a Visiting Professor at the Division of Embedded Systems at Mälardalen University in Västerås, Sweden. His research is on networked embedded systems with time- and resource constraints, where scheduling and resource management are considered to make networked platforms predictable in composition during system development and deployment. Specific topics include reservation-based resource management, limited pre-emptive scheduling, and resource access protocols for multi-core platforms. His main research focuses on modelling and analysis of systems. Main application domains are automotive and industrial, software intensive systems, such as in health-care and manufacturing.



Johan J. Lukkien is a full professor in the section Security and Embedded Networked Systems (SENS) at the Department of Mathematics and Computer Science, Eindhoven University of Technology (TU/e). Since 2002 he, has been Chair of the System Architecture and Networking Research within SENS. Key areas of expertise include probability theory, statistics, software, algorithms, control systems, resource constrained systems, real-time systems and system architecture. Until 2000, Johan was primarily involved in large-scale simulations in physics and chemistry. Since 2000, his research focus has shifted to the application domain of networked resource-constrained embedded systems with a focus on architecture and protocols. Contributions of the SAN group are in the area of component-based middleware for resource-constrained devices, distributed coordination, Quality of Service in networked systems and schedulability analysis in real-time systems.

Affiliations

Jingyue Cao¹ · Pieter J. L. Cuijpers¹  · Reinder J. Bril¹  · Johan J. Lukkien¹

Jingyue Cao
J.Cao@tue.nl

Reinder J. Bril
R.J.Bril@tue.nl

Johan J. Lukkien
J.J.Lukkien@tue.nl

¹ Eindhoven University of Technology, Eindhoven, Netherlands