Article

Improving Worst-Case Delay Analysis for Traffic of Additional Stream Reservation Class in Ethernet-AVB Network

Lin Zhao *, Feng He, Ershuai Li and Huagang Xiong

School of Electronic and Information Engineering, Beihang University, Beijing 100191, China; zhaolin@buaa.edu.cn (L.Z.); Ershuai_li@hotmail.com (E.L.); hgxiong@buaa.edu.cn (H.X.)

* Correspondence: robinleo@buaa.edu.cn; Tel.: +86-10-8233-8894

Received: 19 September 2018; Accepted: 6 November 2018; Published: 9 November 2018

Abstract: With the increase in the number of Electronic Control Units (ECUs) and future requirements for vehicle functions, two SR (Stream Reservation) traffic classes may not be sufficient to ensure fulfillment of constraints for multiple traffic types with individual timing requirements transmitted in the Ethernet-AVB (Audio Video Bridging) networks. The goal of this paper is to determine the worst-case delay for an additional SR traffic class under the CBS (Credit-Based Shaper) algorithm. Delay evaluation is based on the impact analysis of CBS on different priority flows, particularly depending on when the credits of both SR class A and B drain from the worst-case perspective. More specifically, both the impact of CBS and the evolution trends of credit on different priority class flows are first analyzed from the worst-case perspective. Then, for an additional SR class, two types of worst-case delay models are established with the CBS configuration suggestions. Finally, an approach to calculate the worst-case queuing delay is proposed. Moreover, the worst-case end-to-end delay is determined by the network calculus approach and simulation. Numerical results show that the delay bounds of our models are tighter than those of other models, which is beneficial to the development of Ethernet-AVB for in-vehicle networking.

Keywords: automotive; credit-based shaper; deterministic delay analysis; Ethernet-AVB

1. Introduction

Owing to the open standard, high bandwidth, simplicity and low cost characteristics, Ethernet-based networking solutions are most promising for vehicle networks [1]. Several solutions have been presented, including Avionics Full-Duplex Switched Ethernet (AFDX), Time-Triggered Ethernet (TTEthernet) and IEEE Ethernet-AVB (Audio Video Bridging). Thanks to the Virtual Link (VL) concept, bandwidth reservation strategy and redundancy mechanism, AFDX (ARINC 664-part 7 [2]) has been successfully used in civilian aircrafts such as the Boeing 787 and Airbus A380. An improvement on AFDX, TTEthernet (SAE AS6802 [3]) satisfies strict timing transmission by use of a mixture infrastructure to support Time-Triggered (TT), Rate-Constraint (RC) and Best-Effort (BE) traffics. The key feature of TTEthernet is a time-triggered communication paradigm according to the off-line schedule table based on global synchronization [4]. IEEE 802.1 introduces guaranteed timing behavior with the focus on transportation of audio and video streams. IEEE Ethernet-AVB [5–8] is designed as a real-time communication network for multimedia streams with low delay and low jitter. It adopts a Credit-Based Shaper (CBS) on top of the Strict Priority Queuing (SPQ) forwarding policy. Ethernet-AVB’s further proposal TSN (Time Sensitive Networking) [9] develops new shaping mechanisms for Control Data Traffic (CDT) to support hard real-time applications, but some TSN standards are still in progress.
1.1. Related Work

Much research concerns performance comparisons of the above technologies [10–12]. In particular, previous work [10] aims at giving a comprehensive overview of the basic principles behind Ethernet-AVB and mapping them to AFDX. In ref. [11], AFDX, TTEthernet, and Ethernet-AVB are compared in terms of cost, the physical layer, topology, redundancy, security, etc. In ref. [12], the authors present the difference between unsynchronized and synchronized communication and highlight the trends and new horizons in real-time network configurations. Among these technologies, Ethernet-AVB as well as TSN shows great interest as it provides dedicated credit-based bandwidth for multimedia flows relying on a CBS algorithm to satisfy the transmission requirements of the automotive industry [13–16]. The work [17] suggests that the scheduling architecture of Ethernet-AVB should be a candidate for the evolution of AFDX including BE flows. In ref. [18], the authors indicate that CBS could bring additional delays for the highest priority flows during the corresponding credit recovery interval but potentially provides a fairness condition for low priority flows forwarded in switches. Bordoloi et al. [19] builds a necessary condition for the schedulability of class A and B at an output port supporting the CBS algorithm.

In the context of time-critical systems, the challenge lies in the calculation of worst-case delay. Some network solutions, such as Ethernet-AVB and TTEthernet, are compared in terms of end-to-end delay [20,21]. The worst-case end-to-end delay for one frame of a flow transmitted over a network can be broken out into two fixed parts (transmission delay, switching delay) and one variable part (queuing delay) [22]. The transmission delay over a link depends on the frame length and link bandwidth. Switching delay is the delay in switches between input and output ports. It is a constant value with an upper bound of 16 $\mu$s. Queuing delay depends highly on each output port load at the time when the considered frame reaches it. It is not deterministic and lies in the defined scenario.

Existing approaches for the worst-case delay analysis are divided into three categories [23]: Model-checking approach, simulation approach and some mathematical analysis approaches. The model-checking [24] approach depends on an exhaustive exploration of all possible scenarios to obtain an exact worst-case end-to-end delay. Unfortunately, this approach cannot cope with large-scale industrial configurations due to the combinatorial explosion problem. The simulation approach achieves the maximum delay observed on a set of scenarios but cannot provide guarantees on worst-case end-to-end delay. Some mathematical analysis approaches compute a sure upper-bound delay with a certain pessimism, such as network calculus [25,26], the trajectory approach [27] and the holistic approach [28]. In particular, network calculus is a mature approach that has been successfully used for worst-case delay analysis in the AFDX network [29,30]. In refs. [31,32], the authors discuss Ethernet-AVB network calculus models. In refs. [33,34], network calculus is used to make timing analysis of AVB traffic in TSN networks. In ref. [35], an improved trajectory approach is considered to obtain tight flow delay bounds with serialization constraints in the Ethernet-AVB network.

1.2. Motivation and Contributions

According to 802.1Q [36], each Ethernet-AVB output port offers up to 8 traffic classes including a maximum of two SR traffic classes A and B with highest priority (A having higher priority than B) plus six non-SR traffic classes, such as Best Effort (BE) traffic. In a current premium car, there are up to 70 Electronic Control Units (ECUs) with more than 900 functions interconnected over the in-vehicle network [21]. Each ECU sends a set of flows through an output port. Multiple traffic types, like control signals, media signals, diagnostics signals, and infotainment signals, will share the same physical infrastructure. It is inevitable that some types of flows with individual timing requirements will be mapped to the same class [19]. Two SR traffic classes may not be enough to ensure fulfillment of all constraints for all flows with individual timing requirements. Now, some TSN specifications are in progress to guarantee time-critical transmission. On the other hand, for traffic with less-rigid timing requirements, additional SR classes should be considered.
Cao et al. [37] gives a tight bound on the worst-case interference analysis for individual priority classes H (high priority), M (medium priority) and L (BE priority) in an AVB switch. However, it still lacks of the analysis of multiple SR classes. IEEE 802.1Qav [8] has built a general formula to determine the worst-case queuing delay experienced by any SR class. When calculating the worst-case queuing delay for an additional SR class with lower priority than class A and B, it looks at higher priority classes (class A and B) together as a single class to perform the analysis and deduction. Thus, the delay result for the additional SR class is pessimistic. The approach proposed in this paper is to reduce the pessimism in the analysis to provide tighter delay bounds for an additional SR class traffic. The primary contributions are summarized as follows:

1. We evaluate the impact of CBS on different priority flows. Particularly, credit variation of class B under constraints of class A at an AVB switching output port is indicated, and the transmission condition for an additional SR class traffic is specified.

2. We identify two types of worst-case queuing delay models for an additional SR class traffic and build a necessary condition for the appearance of each model.

3. We investigate the evolutions of credit A and B for each of the two worst-case queuing delay models and propose an algorithm to determine when the credits of both class A and B are negative. Then, the worst-case queuing delay bounds for an additional SR class are achieved.

1.3. Organization

The rest of this paper is organized as follows. Section 2 gives an overview of the Ethernet-AVB network and describes the system model for studying the worst-case queuing delay of an additional SR traffic class. In Section 3, the performance evaluation for the optimization model is implemented. Numerical results are presented in Section 4 for verification our analysis and are followed by our conclusion in Section 5.

2. System Model

2.1. Context of Ethernet-AVB

Ethernet-AVB [5–8] comprises some IEEE 802.1 standards for low latency flows. IEEE 802.1 BA [5] defines the Ethernet-AVB system and default configuration. IEEE 802.1AS [6] is specified to ensure the time synchronization requirements for time-sensitive applications, which is based on the IEEE 1588-precision time protocol and provides a synchronization error less than 1 µs over seven hops. IEEE 802.1Qat defines a Stream Reservation Protocol (SRP) [7] to accomplish the reservation request along the path in three steps: stream advertisement, registration and de-registration. According to 802.1Qat, the bandwidth requirement (bits/s units) of a given SR class flow is given by

\[
\text{Bandwidth} = (\text{MFS} + \text{Overhead}) \times 8 \times \frac{\text{MIF}}{\text{CMI}},
\]

where MFS (Maximum Frame Size) is the maximum frame size of the considered SR class flow. CMI (Class Measurement Interval) is a periodical time interval with 125 µs for class A and 250 µs for class B. In order to expand the application scope, CMI can perhaps be generalised with different values by some switch providers, as assumed in ref. [31]. MIF (Maximum Interval Frame) is the maximum number of frames transmitted during one CMI.

We focus on IEEE 802.1Qav [8], which defines queuing and forwarding policy. Each Ethernet-AVB output port imposes a CBS algorithm for each SR class to accomplish traffic shaping. The CBS process is the following, as depicted in Figure 1 (inspired by [8] (Fig L-4)).
If a SR class frame is waiting for transmission (there is conflicting traffic blocking the output port or the credit is negative), the credit increases at its idleSlope (idSl for short) rate. IdSl represents the maximum guaranteed bandwidth fraction allocated to a given SR class. At most, 75% bandwidth usage is allocated to all SR traffic classes.

If the corresponding credit allocated to a given SR class is not negative and the link is idle, the transmission of the SR frame is only allowed when there is no higher-priority traffic awaiting transmission or the corresponding higher priority class credit is not enough for transmission. As the transmission proceeds, the credit decreases at the rate of sendSlope (sdSl for short), and in the worst-case scenario, a maximum-sized frame continues its emission at zero credit up to completion, even if the credit becomes negative. The parameter sdSl obeys: sdSl = idSl − linkspeed.

If there is no further SR frame queued and the current credit is negative, credit will increase to zero at the rate of idSl. Otherwise, if there are no frames in the SR class queue and its credit is positive, credit is immediately set to zero.

The delay experienced by a frame in a queue can be decomposed into two parts.

First, the delay between the instant the frame is enqueued and the instant it becomes the head of queue.

Second, the delay between becoming first frame of queue and the instant it is selected for emission.

Let X represent an SR traffic class. For the non-preemptive SPQ scheduling mechanism, the queuing delay experienced by the first frame of SR class X queue can be broken out into two components [8]:

- The delay is caused by the frame that was selected for transmission an arbitrarily small time before frame X arrived. In the worst-case scenario, it is the transmission time of a maximum-sized frame with lower priority than frame X’s class.
- The delay is caused by queued-up frames with higher priority than frame X’s class.

As depicted in Figure 1 and from [8] (eq L.38), the queuing delay experienced by the first frame of SR class A queue is given by

\[ qDelay_A = t_{ab} = M_0 / R_0, \]

where \( R_0 \) represents the port transmission rate and \( M_0 \) is the maximum length of an interference frame in the BE class. Let \( R_X \) denote the idleSlope for SR class X and \( M_X \) the maximum length of a frame in SR class X. Then the sendSlope for SR class X is \( (R_X - R_0) \). In terms of class A, calculating the queuing delay experienced by the first frame of SR class B queue is easy, which is given by

\[ qDelay_B = t_{ab} + t_{bd} = M_0 / R_0 + \frac{\left[ R_A(M_0 / R_0) - (R_A - R_0)(M_A / R_0) \right]}{(R_0 - R_A)} / (R_0 - R_A) \]

\[ = M_0 / (R_0 - R_A) + M_A / R_0. \]
However, calculating the queuing delay experienced by an additional SR class with lower priority than class A and B is more difficult. When computing the worst-case queuing delay of an additional SR traffic, a trick in 802.1Qav is to looks at higher priority classes together as a single class. By using the sum of the credits available to all higher priority SR classes, a general formula for calculating the worst-case queuing delay experienced by any SR class X is expressed as follows

\[ q_{\text{Delay}}_X = \left( M_0 + \sum_{k<X} M_k \right) / W_{<X}, \]

where \( W_{<X} = R_0 - \sum_{k<X} R_k \). "<X" is a subscript for the sum of all classes with higher priority than class X [8] (eq L.37).

### 2.2. Optimization Model

In this section, we define a worst-case analytical model that we use throughout the paper. The existing two SR classes in the standard Ethernet-AVB network are extended into three SR classes A, B and C, as depicted in Figure 2. Class C has lower priority than class A and B. Denote that N represents the SR traffic class (\( N \in \{A,B,C\} \)) associated with the CBS algorithm and dedicated bandwidth allocation. Flows not belonging to class N are treated as BE flows with the lowest priority. The queue for each SR class is full, and the corresponding credit starts at zero. In ref. [35], it has been proved that for a frame of an AVB SR class, the worst-case scenario can always be found when considering zero initial credit of the corresponding SR class at each output port along the path. Frames belonging to one SR class and the class with higher priority arrive right after a maximum-sized conflicting frame of a lower prioritized class has started to transmit. When the transmission of frames in a given SR class is allowed, the corresponding credit decreases and a maximum-sized frame continues its emission at zero credit up to completion. The credits of the other SR classes increase in this process. The transmission of lower priority SR class frames is only allowed when the credits of all higher priority SR classes are negative.

![Diagram of SR classes and BE flows](image)

**Figure 2.** Analytical models (Model 1: Class B frames are forwarded until its credit drains; Model 2: Class B frames stop being forwarded when the credit of class A replenishes to zero).

However, it is difficult to determine the worst-case queuing delay experienced by SR class C, which is affected by class A and B’s flow shaping operation. In addition, the usage of \( \text{idSl} \) and \( \text{sdSl} \) provides great flexibility for Ethernet-AVB flow control. Particularly, if the corresponding \( \text{idSl} \) rate is increased, the transmission speed for SR class flows is accelerated and the credit recovers quickly. Otherwise, if the corresponding \( \text{idSl} \) value is decreased, the transmission speed is slowed down. When
calculating the worst-case queuing delay experienced by the first frame of SR class C queue, the key point is to determine the evolution trend of credit A and B.

Class A frames have the highest priority. As depicted in Figure 2, in the worst-case scenario, the transmission of class A frames is blocked by a maximum-sized conflicting frame with lower priority than class A, and class A’s credit increases. After the transmission of the conflicting frame has completed, class A frames start to be transmitted, and credit decreases. Credits of class B and C increase in this process. After the first transmission of class A frames has completed, class A’s credit is not enough for transmission and needs to accumulate. Then, the first transmission of class B frames is allowed and class B’s credit decreases. Meanwhile, credits of class A and C increase. However, the class B frame may stop being forwarded after the credit of class A replenishes to zero. As depicted in Figure 2, during the first transmission of class B frames, whether class B’s credit falls below zero lies in the speed of class A’s credit recovery and will lead to the corresponding Model 1 and Model 2.

Subsequently, some combination of class A and B frames are transmitted after the credit of class A and class B replenishes to zero. As depicted in Figure 2, the subsequent possibilities of frames being transmitted and the subsequent evolution of credit A and B are difficult to predict.

Thus, to determine the worst-case queuing delay experienced by the first frame of SR class C queue, the necessary conditions for the appearance of Model 1 and Model 2 should first be built; then, the evolution trend of credit A and B for each model should be determined. The next section will perform the analysis in detail.

3. Performance Evaluation

3.1. Necessary Conditions for the Appearance of Model 1 and Model 2

The following notations are used in the analysis models:

Assume that the queue for each SR class is full. Consider the frame sequence before the emission of the first C frame: it starts with an optional non SR frame, then it is an alternation of sequences of A frames and B frames. For example, if the output link is AAABBABAC, they are 3 sequences of A frames and B frames. Let \( Credit^\text{max}_N \) (\( Credit^\text{min}_N \)) be the maximal (minimal) bound of the credit during the transmission of the kth sequence of frames of the SR class N (\( k = 1, 2, ..., n \)) , as depicted in Figure 3.

![Figure 3](image-url)  

Figure 3. The impact of CBS on class B flows. (a) Class B’s credit has fallen to zero before that class A’s credit replenishes to zero; (b) Class B’s credit has not fallen to zero before that class A’s credit replenishes to zero.

\( R \) is the transmission rate of the network. \( L^\text{max}_N \) (\( L^\text{max}_B \)) is the maximum size of a frame of class N (BE class). \( L^\text{max}_{p(k)} \) represents the maximum size of a conflicting frame of a lower priority class than class N. Then, the expressions \( L^\text{max}_{p(A)} = \max \{ L^\text{max}_A, L^\text{max}_C, L^\text{max}_{BE} \} \), \( L^\text{max}_{p(B)} = \max \{ L^\text{max}_A, L^\text{max}_B \} \) and \( L^\text{max}_{p(C)} = \max \{ L^\text{max}_C, L^\text{max}_{BE} \} \) are achieved. \( t^\text{B}_0 \) represents the duration when class A’s credit increases from \( Credit^\text{min}_A \) to zero and \( t^\text{A}_0 \) represents the duration when class B’s credit decreases from \( Credit^\text{max}_B \) to zero; then,
\[ t_{A_k}^{r_0} = - \text{Credit}_{A_k}^{\text{min}} / idS_{A_k} \]  
\[ t_{B_k}^{r_0} = - \text{Credit}_{B_k}^{\text{max}} / sdS_{B_k}. \]  

If the transmission of class B frames is allowed and \( t_{A_k}^{r_0} > t_{B_k}^{r_0} \), class B’s credit has fallen to zero before that class A’s credit replenishes to zero and a maximum-sized class B frame can continue to be forwarded at credit zero in the worst-case scenario, as depicted in Figure 3a. Otherwise, if the transmission of class B frames is allowed and \( t_{A_k}^{r_0} \leq t_{B_k}^{r_0} \), class B’s credit cannot continue to decrease at the \( sdS_{B_k} \) rate at the time when class A’s credit replenishes to zero. It will increase at the \( idS_{B_k} \) rate until the transmission of a maximum-sized class A frame has been completed, as depicted in Figure 3b.

Thus, when the first transmission of class B is allowed (\( k = 1 \)), the necessary conditions are \( t_{A_1}^{r_0} > t_{B_1}^{r_0} \) for Model 1 and \( t_{A_1}^{r_0} \leq t_{B_1}^{r_0} \) for Model 2, as depicted in Figure 2, where \( t_{A_1}^{r_0} \) is the duration when class A’s credit increases from \( \text{Credit}_{A_1}^{\text{min}} \) to zero, and \( t_{B_1}^{r_0} \) is the duration when class B’s credit decreases from \( \text{Credit}_{B_1}^{\text{max}} \) to zero. Next, we discuss how to determine the values of these parameters.

Class A frames have priority over all other traffics. When the transmission is allowed, frames will be transmitted back-to-back without interrupting until the credit drains. In addition, in the worst-case scenario, a maximum-sized frame can still be forwarded when the credit decreases to zero. The check of the minimum credit value is performed at the end of the emission of this maximum-sized frame. So, the value of \( \text{Credit}_{A_k}^{\text{min}} \) is constant, which is given by

\[ \text{Credit}_{A_k}^{\text{min}} = \left( \frac{L_{\text{max}}}{R} \right) \times sdS_{A_k}. \]  

\( \text{Credit}_{A_1}^{\text{max}} \) is the amount of credit that can be accumulated during the transmission time of a maximum-sized conflicting frame with lower priority than class A; then,

\[ \text{Credit}_{A_1}^{\text{max}} = \left( \frac{L_{\text{max}}}{I_{\text{lp}(A)} / R} \right) \times idS_{A_1}. \]  

\( \text{Credit}_{B_1}^{\text{max}} \) is the amount of credit that can be accumulated during the transmission time of a maximum-sized conflicting frame with lower priority than class B plus the transmission time of the maximum numbers of class A frames, which is given by

\[ \text{Credit}_{B_1}^{\text{max}} = idS_{B_1} \left( \frac{I_{\text{lp}(B)}}{R} + \frac{\text{Credit}_{A_1}^{\text{max}} - \text{Credit}_{A_1}^{\text{min}}}{sdS_{A_1}} \right) \]
\[ = idS_{B_1} \left( \frac{I_{\text{lp}(B)}}{R} + \frac{L_{\text{max}}}{R} \frac{I_{\text{lp}(A)}}{I_{\text{lp}(A)} / R} \right) \]
\[ = idS_{B_1} \left( \frac{I_{\text{lp}(B)}}{R} + \frac{L_{\text{max}}}{R} \frac{I_{\text{lp}(A)}}{I_{\text{lp}(A)} / R} \right). \]  

where \( \text{Credit}_{A_1}^{\text{min}} = \left( \frac{L_{\text{max}}}{R} \right) \times sdS_{A_1} \), as seen in (7). Using (5) and (6), the values of \( t_{A_1}^{r_0} \) and \( t_{B_1}^{r_0} \) are given by

\[ t_{A_1}^{r_0} = \frac{- \text{Credit}_{A_1}^{\text{min}}}{idS_{A_1}} = \frac{L_{\text{max}}}{R} \left( - \frac{sdS_{A_1}}{idS_{A_1}} \right), \]  

and

\[ t_{B_1}^{r_0} = \frac{- \text{Credit}_{B_1}^{\text{max}}}{sdS_{B_1}} \]
\[ = \frac{idS_{B_1}}{(-sdS_{B_1})} \left( \frac{L_{\text{max}}}{R} - \frac{L_{\text{max}}}{R} \frac{I_{\text{lp}(A)}}{I_{\text{lp}(A)} / R} \right). \]
As seen in (10) and (11), the effect of CBS depends on flow loads and idleSlope configurations. Using this configuration information, the values of $t_{A_1}^{10}$ and $t_{B_1}^{10}$ can be calculated and compared. Then, the worst-case queuing delay model of class C is determined.

3.2. Analysis of the Evolution of Credit for Two Models

**Theorem 1.** (Credit evolution analysis for Model 1) If $t_{A_1}^{10} > t_{B_1}^{10}$, the maximum values of credit A and B ($Credit_{A_1}^{\max}$ and $Credit_{B_1}^{\max}$, respectively) become smaller and smaller as the value of k increases, and class B's credit will drain during every transmission of class B frames, as depicted in Figure 4. The following relationships are achieved: $Credit_{A_{k+1}}^{\max} < Credit_{A_k}^{\max}$, $Credit_{B_{k+1}}^{\max} < Credit_{B_k}^{\max}$ and $Credit_{B_k}^{\min} = (L_B^{\max}/R) \times sdSl_B$.

![Figure 4. Credit evolution for Model 1.](image)

**Proof of Theorem 1.** Since $t_{A_1}^{10} > t_{B_1}^{10}$, $Credit_{A_1}^{\min} = (L_{A_1}^{\max}/R) \times sdSl_A$ is obtained according to the previous analysis. Let $\Delta t_2$ be the duration when class A's credit increases from zero to $Credit_{A_2}^{\max}$ (see $t_4,t_5$ in Figure 4), so $Credit_{A_2}^{\max} = \Delta t_2 \times idSl_A$. Then, it is possible to express the relationship: $\Delta t_2 < (L_B^{\max}/R)$ with $t_{A_1}^{10} > t_{B_1}^{10}$ assumed. Please note that the maximum size of a conflicting frame for class A is $L_{A_2}^{\max} = \max\{L_{B_2}^{\max}, L_{C_2}^{\max}, L_{BE_2}^{\max}\}$; comparing the value of $Credit_{A_2}^{\max}$ with the value of $Credit_{A_1}^{\max}$ seen in (8), the following relationship is achieved:

$$Credit_{A_2}^{\max} < Credit_{A_1}^{\max}. \tag{12}$$

Class B’s credit reaches the value of $Credit_{B_{k+1}}^{\max}$ when class A’s credit reaches the value of $Credit_{A_k}^{\min}$. The value of $Credit_{B_{k+1}}^{\max}$ is computed by the following equation:

$$Credit_{B_{k+1}}^{\max} = idSl_B \left( \frac{Credit_{A_{k+1}}^{\max} - Credit_{A_{k+1}}^{\min}}{-sdSl_A} \right) + Credit_{B_k}^{\min}, \tag{13}$$

where $Credit_{A_{k+1}}^{\min} = (L_A^{\max}/R) \times sdSl_A$ and $Credit_{B_k}^{\min} = (L_B^{\max}/R) \times sdSl_B$. Comparing the value of $Credit_{B_k}^{\max}$ computed from (13) with the value of $Credit_{B_k}^{\max}$ seen in (9), we have

$$Credit_{B_k}^{\max} < Credit_{B_1}^{\max}. \tag{14}$$
Thus, the relationship \( t_{A_1}^{B_2} < t_{A_1}^{B_1} < t_{A_1}^{A_2} \) is deduced. In addition, since the value of \( \text{Credit}_{A_k}^{\text{min}} \) is constant, \( t_{A_1}^{A_2} = t_{A_1}^{A_2} \) is obtained by using (5). Then, \( t_{A_0}^{B_2} < t_{A_0}^{B_0} \) and \( \text{Credit}_{B_k}^{\text{min}} = \left( L_B^{\text{max}} / R \right) \times \text{sdSL}_B \) are achieved. Furthermore, class A’s credit reaches the value of \( \text{Credit}_{A_k}^{\text{max}} \) when class B’s credit reaches the value of \( \text{Credit}_{B_k}^{\text{min}} \). The value of \( \text{Credit}_{A_{k+1}}^{\text{max}} \) is computed by the following equation:

\[
\text{Credit}_{A_{k+1}}^{\text{max}} = \text{idSL}_{A} \left( \frac{\text{Credit}_{B_k}^{\text{max}} - \text{Credit}_{B_k}^{\text{min}}}{-\text{sdSL}_B} \right) + \text{Credit}_{A_k}^{\text{min}}. \tag{15}
\]

Using (12) and (14), (13) and (15) continue to be computed iteratively. Then, we have: \( \text{Credit}_{A_{k+1}}^{\text{max}} < \text{Credit}_{A_k}^{\text{max}} \) and \( \text{Credit}_{B_{k+1}}^{\text{max}} < \text{Credit}_{B_k}^{\text{max}} \). The relationship \( t_{A_k}^{A_0} > t_{B_1}^{B_1} > t_{B_2}^{B_0} > \ldots > t_{B_0}^{B_0} \) is derived. Finally, \( \text{Credit}_{B_k}^{\text{min}} = \left( L_B^{\text{max}} / R \right) \times \text{sdSL}_B \) is obtained.

**Theorem 2.** (Credit evolution analysis for Model 2) If \( t_{A_1}^{A_0} \leq t_{B_1}^{B_1} \), during the first or first few times of class B transmission, the minimum value of credit B is not less than zero (see \( t_2, t_5 \) in Figure 5). However, subsequently, credit evolution will follow what was analyzed in Model 1 (see \( t_5, t_7 \) in Figure 5). The following relationships are achieved: \( \text{Credit}_{B_{k+1}}^{\text{max}} < \text{Credit}_{B_k}^{\text{max}} \), and \( \text{Credit}_{A_k}^{\text{max}} > 0 \Rightarrow \text{Credit}_{A_i}^{\text{max}} < \inf \{ \text{Credit}_{A_j}^{\text{max}} \mid j < i, \text{Credit}_{A_i}^{\text{max}} > 0 \} \).

![Figure 5. Credit evolution for Model 2.](image)

**Proof of Theorem 2.** During the first transmission of class B, if \( t_{A_1}^{A_0} \leq t_{B_1}^{B_1} \), class B frames cannot continue to be forwarded at the time when class A’s credit replenishes to zero (see \( t_3 \) in Figure 5). Define that \( P \) is a periodical time interval that is characterized by the duration when class A’s credit increases from \( \text{Credit}_{A_k}^{\text{min}} \) to zero plus the duration when the class A’s credit decreases from zero to \( \text{Credit}_{A_{k+1}}^{\text{min}} \) (see \( t_2, t_4 \) in Figure 5). During a period of \( P \), let \( t_{A_i}^{B_k} \) represent the duration when class B’s credit decreases from \( \text{Credit}_{B_k}^{\text{max}} \) to \( \text{Credit}_{B_k}^{\text{min}} \), and let \( t_{B_k}^{B_{k+1}} \) represent the duration when the class B’s credit increases from \( \text{Credit}_{B_k}^{\text{min}} \) to \( \text{Credit}_{B_{k+1}}^{\text{max}} \), which are given by
\[ t_B^\uparrow = t_A^{0\mathrm{A}} = -\text{Credit}_{A_k}^{\min} / idSL_A, \] (16)
\[ t_B^\downarrow = t_A^{0\mathrm{B}} = \text{Credit}_{A_k+1}^{\min} / sdSL_A, \] (17)

where \( t_A^{0\mathrm{A}} \) is the duration when class A’s credit increases from \( \text{Credit}_{A_k}^{\min} \) to zero and \( t_A^{0\mathrm{B}} \) is the duration when class A’s credit decreases from zero to \( \text{Credit}_{A_k+1}^{\min} \).

Define that \( \text{Credit}_{B}^{\downarrow} \) and \( \text{Credit}_{B}^{\uparrow} \) are the credit variation during the duration \( t_B^\downarrow \) and \( t_B^\uparrow \) respectively; then,

\[ \text{Credit}_{B}^{\downarrow} = \text{Credit}_{A_k}^{\min} \times sdSL_B, \] (18)
\[ \text{Credit}_{B}^{\uparrow} = \text{Credit}_{A_k+1}^{\min} \times idSL_B. \] (19)

According to 802.1Qat, assume that \( idSL_A + idSL_B = S \leq 75\% \times R \); then

\[ \frac{-sdSL_B}{idSL_A} = \frac{(R - S) + idSL_A}{idSL_A} > 1, \] (20)
\[ \frac{sdSL_B}{idSL_B} = \frac{(R - S) + idSL_B}{idSL_B} < 1. \] (21)

Hence, the expression \( \text{Credit}_{B}^{\uparrow} > \text{Credit}_{B}^{\downarrow} \) is obtained. After the number of \( m (m = 1, 2, ..., m < k) \) periods of \( P \) (see \( [t_2, t_3] \) in Figure 5) only if \( m \) meets the following relationship:

\[ \text{Credit}_{B_1}^{\uparrow} \leq \text{Credit}_{B_1}^{\max} - m(\text{Credit}_{B}^{\uparrow} - \text{Credit}_{B}^{\downarrow}) < \text{Credit}_{B}^{\downarrow}, \] (22)

the value of \( \text{Credit}_{B_{m+1}}^{\max} \) is small enough to satisfy: \( t_{A_{m+1}}^{0\mathrm{A}} > t_{B_{m+1}}^{0\mathrm{B}} \) (see \( [t_5, t_6] \) in Figure 5) and computed by

\[ \text{Credit}_{B_{m+1}}^{\max} = \text{Credit}_{B_1}^{\max} - m(\text{Credit}_{B}^{\uparrow} - \text{Credit}_{B}^{\downarrow}). \] (23)

From then on (see \( t_5 \) in Figure 5), during the transmission of class B frames, class B’s credit value will fall below zero. The subsequent credit evolution trend is just as described in Theorem 1. Thus, we have: \( \text{Credit}_{B_{m+1}}^{\max} < \text{Credit}_{B_k}^{\min} \) and \( \text{Credit}_{A_i}^{\max} > 0 \Rightarrow \text{Credit}_{A_j}^{\min} < \inf \{ \text{Credit}_{A_i}^{\max} | j < i, \text{Credit}_{A_j}^{\max} > 0 \} \).

3.3. Determining the Worst-Case Queuing Delay Experienced by the First Frame of SR Class C Queue

As in the previous analysis, when the CBS configurations are given, the worst case queuing delay model is determined. According to the discussion on credit evolution for the two models depicted in Theorems 1 and 2, the values of credit A and B become smaller and smaller except that the credit of class A is bigger at one moment in Model 2 (see \( t_6 \) in Figure 5). Finally, both credits of class A and B are negative, and then the transmission of class C frames is allowed. In addition, the worst-case queuing delay experienced by the first frame of SR class C queue is obtained. Using Theorems 1 and 2, the detailed algorithm steps of the iterative method are presented in Algorithm 1.
Algorithm 1: The worst-case queuing delay experienced by the first frame of SR class C queue

**Input:** $R, idSl_N, L_{max}^A, L_{max}^B$

**Output:** The worst-case queuing delay experienced by the first frame of SR class C queue: $T_C$

1. Initialize the model parameters: $Credit_{A1}^{max}, Credit_{B1}^{max}, Credit_{A_k}^{min}, t_{A1}^{↑}, t_{B1}^{↓}, Credit_{B}^{↑}, Credit_{B}^{↓}$
2. if ($t_{A1}^{↑} < t_{B1}^{↓}$) then
3. repeat
4. $Credit_{B_k+1}^{max} \leftarrow (Credit_{B_k}^{max} - Credit_{B}^{↑} + Credit_{B}^{↓})$
5. until ($Credit_{B_k}^{max} < Credit_{B}^{↑}$)
6. end if
7. repeat
8. $Credit_{A_{k+1}}^{max} \leftarrow \frac{Credit_{A_k}^{max} - L_{max}^{A} / R \times sd_{Sl_{B}} - sd_{Sl_{A}}}{-sd_{Sl_{A}}} + t_{A1}^{↑} sd_{Sl_{A}}$
9. if ($Credit_{A_{k+1}}^{max} \geq 0$) then
10. $Credit_{B_{k+1}}^{max} \leftarrow \frac{Credit_{A_{k+1}}^{max} - L_{max}^{A} / R \times sd_{Sl_{A}} - sd_{Sl_{A}}}{-sd_{Sl_{A}}} + t_{B1}^{↓} sd_{Sl_{B}}$
11. else
12. break;
13. end if
14. until ($Credit_{B_k}^{max} < 0$)
15. return $T_C$

4. Numerical Results

4.1. Simple Cases Illustration

First, two simple cases are studied to verify the correctness and advantage of our models. Four flows assigned with class A, B, C and BE are forwarded at an Ethernet-AVB switch output port. The transmission rate of the network is 100 Mbits/s. The CBS configurations are given in Table 1.

| Case | Class | MaximumFrameSize (Byte) | idleSlope (Mbits/s) |
|------|-------|-------------------------|--------------------|
| 1    | A     | 520                     | 35                 |
|      | B     | 1000                    | 25                 |
|      | C     | 1518                    | 15                 |
|      | BE    | 1518                    | 25                 |

|      | A     | 84                      | 30                 |
|      | B     | 800                     | 25                 |
|      | C     | 1200                    | 20                 |
|      | BE    | 1518                    | 25                 |

The results for Case 1 are shown in Figure 6a. Using (5) and (6), the expressions $t_{A1}^{↑} = 77$ μs and $t_{B1}^{↓} = 48$ μs are obtained. The evolution of credit is consistent with what is described in Theorem 1 with $t_{A1}^{↑} > t_{B1}^{↓}$. The results for Case 2 are shown in Figure 6b. Using (5) and (6), the expressions $t_{A1}^{↑} = 16$ μs and $t_{B1}^{↓} = 56$ μs are obtained. The evolution of credit is consistent with what is described in Theorem 2 with $t_{A1}^{↑} < t_{B1}^{↓}$. 
Figure 6. The worst-case queuing delay results. (a) Results for Case 1. (b) Results for Case 2.

The comparison results of a worst-case queuing delay between our method and Formulas (2)–(4) mentioned in 802.1Qav [8] are shown in Table 2.

Table 2. Queuing delay comparison. The ratio shows factor of improvement.

| Case | Class | By Qav [µs] | By Our Method [µs] | Ratio |
|------|-------|-------------|--------------------|-------|
| 1    | A     | 121         | 121                | 1:1   |
|      | B     | 228         | 228                | 1:1   |
|      | C     | 608         | 469                | 1:0.77|
| 2    | A     | 121         | 121                | 1:1   |
|      | B     | 180         | 180                | 1:1   |
|      | C     | 427         | 370                | 1:0.87|

For class A and B, the worst-case queuing delay obtained from our models are the same as those obtained by the 802.1Qav standard. The correctness of our models is verified. For class C, the results show that our method brings significant improvement of up to 23% in Case 1 and 13% in Case 2. The advantage of our models is illustrated.

4.2. Evaluation with Different Size and IdleSlope

In this section, the influence of frame lengths and the idleSlope of class A and B on the worst-case queuing delay of class C is investigated. Assume that the value of idSlC is 0.15R Mbits/s (R = 100 Mbits/s); then, the sum value of idSlA and idSlB is 0.6R Mbits/s. Let the value of idSlA increase from 0.1R Mbits/s to 0.5R Mbits/s by 0.05R; then, the value of idSlB decreases from 0.5R Mbits/s to 0.1R Mbits/s by 0.05R correspondingly. Frame lengths of class B, C and BE are 1000 Byte, 1518 Byte and 1518 Byte, respectively. Small-sized frames are assigned to class A, and the sizes of 64 Byte, 84 Byte and 184 Byte are taken as a reference. As depicted in Figure 7, the results show that our approach produces tighter queuing delay bounds in the case of class C and is capable of reflecting the delay variation over the increasing of frame lengths and the idleSlope of higher priority traffic classes.

4.3. End-to-End Delay Evaluation

In addition, a case study on the worst-case end-to-end delay, which is inspired by [38], is depicted in Figure 8. Ten end systems, E1~E10, are connected to three Ethernet-AVB switches: S1~S3. Four types of streams mapped to class A, B, C and BE are transmitted over the network. Control signals (CS) have the highest priority to map into class A streams; media signals (MS) and some diagnostics signals (DS) with less rigid timing requirements are mapped to class B and class C, respectively;
background signals are regarded as BE streams, which broadcast in the network. The transmission rate of the network is 100 Mbits/s, and the technology switching delay of each switch is 16 µs. The detailed configurations are shown in Table 3.

Table 3. Configurations.

| Traffic Type | Bandwidth (Mbits/s) | Payload (Byte) | CMI (ms) | Class (Priority) |
|--------------|---------------------|----------------|----------|-----------------|
| CS           | 0.672               | 84             | 1        | A               |
| MS           | 8.0                 | 1200           | 1.2      | B               |
| DS           | 4.0                 | 500            | 1        | C               |
| BE           | Bursts              | 1518           |          | BE              |

Before Section 4.3, an approach for determining worst-case queuing delay experienced by the first frame of SR class C queue is proposed. In this section, network calculus approach is used to obtain the worst-case end-to-end delay for a class C flow. The network calculus approach has been applied to the switched Ethernet network to guarantee real-time communication even if it provides delay upper bounds with pessimistic computation that the simulation method cannot calculate [23]. The basic concept of network calculus can be found in Appendix A. A flow in class N has a maximum frame size and a class measurement interval. Its arrive curve obeys the leaky bucket model [25]. A typical service curve is the rate-latency function [25]. The worst-case queuing delay calculated by our models and 802.1Qav [8] is regarded as the service latency of the service curve. The service curve is built and proved by Lemma A1 in Appendix A. Then, the flow delay bound is the horizontal deviation
between the corresponding arrive curve and service curve. The results show that our method gives tighter end-to-end delay bounds. In addition, we use the OMNeT++ simulator to obtain the maximum observed end-to-end delays as a reference, as depicted in Figure 9.

![Comparison results on end-to-end delay of class C](image)

**Figure 9.** Comparison results on end-to-end delay of class C (Simulation is to use the OMNeT++ simulator to get the maximum observed end-to-end delays. The traditional method and improved method are both employed to use the network calculus approach to obtain the end-to-end delay bounds, in which the service latency of the service curve is calculated by 802.1Qav [8] and our models described in the previous section).

5. Conclusions

In this paper, the worst-case upper-bounded delay for an additional SR traffic class has been investigated. First, the impact of CBS on different priority flows is researched in detail. Second, according to the CBS configurations, two types of worst-case queuing delay models for the additional SR traffic class are identified, and the necessary conditions for each type of model is built. Based on the analytical results, the worst-case execution time required for flows mapped into an additional SR class may be satisfied with appropriate configuration suggestions on the premise of ensuring timing requirements for higher priority SR flows. Furthermore, the credit evolution of each model is analyzed, and an algorithm for calculating the worst-case queuing delay is given. Compared with the method mentioned in 802.1Qav, our method brings a significant improvement of up to 22.9% in Case 1 and 13.3% in Case 2. Finally, our approach reduces the pessimism to provide tight end-to-end delay bounds by using network calculus. In addition, it is important to note that our approach can be extended to an arbitrary number of SR classes. It is expected that our approach may guide the design of Ethernet-AVB for in-vehicle networking. The impact of time-critical class CDT traffic on our models will be evaluated in the future.

**Author Contributions:** Conceptualization, L.Z. and F.H.; formal analysis and data curation, L.Z. and E.L.; writing—original draft preparation, L.Z.; writing—review and editing, F.H. and H.X.

**Funding:** This research was funded by the National Natural Science Foundation of China grant number 61301086.

**Acknowledgments:** The authors want to express their thanks to the reviewers for their helpful constructive suggestions to keep improving the quality of this paper.

**Conflicts of Interest:** The authors declare no conflict of interest.
Appendix A. Network Calculus Approach

Specify that \( R(t) \) (\( R^*(t) \)) is the input (output) function of data flows through a network element; then, \( R(t) \) is constrained by the arrive curve \( a(t) \) [25] if and only if for all \( s \leq t, R(t) - R(s) \leq a(t-s) \).

A typical arrive curve is the leaky bucket function, which is given by

\[
\alpha_{\sigma, \rho}(t) = \begin{cases} 
\sigma + \rho t & \text{if } t > 0 \\
0 & \text{otherwise,} 
\end{cases}
\]  

(A1)

where \( \sigma \) is the maximum burst tolerance and \( \rho \) is the long-term constant rate. If the flow is transmitted through a link with the rate \( R \), a tight arrive curve is defined by \( \alpha(t) = \min\{R \times t, \alpha_{\sigma, \rho}(t)\} \), as depicted in Figure A1. For a flow belonging to class \( N \) in the Ethernet-AVB network, \( \sigma = L^N_{\text{max}} \times \text{MIF} \times (1 - \rho/R) \) and \( \rho = L^N_{\text{max}} \times \text{MIF}/\text{CMI} \) [32].

![Illustration of the arrive curve and the service curve.](image)

**Figure A1.** Illustration of the arrive curve and the service curve.

A network element offers a flow with \( R(t) \) and \( R^*(t) \) a service curve \( \beta(t) \) [25] if and only if for all \( t \geq 0, R^*(t) \geq \inf_{s \leq t} (R(s) + \beta(t-s)) \). A typical service curve is the rate-latency function, which is given by

\[
\beta_{R,T}(t) = R(t - T)^+ = \begin{cases} 
R(t - T) & \text{if } t > 0 \\
0 & \text{otherwise,} 
\end{cases}
\]  

(A2)

where \( R \) and \( T \) represent the service rate and the service latency, respectively.

**Lemma A1.** For a flow belonging to class \( N \) in the Ethernet-AVB network, if the AVB shaper is allocated with parameters \( \text{idSl}_N \) (the idleSlope of class \( N \)) and \( T_N \) (the worst-case queuing delay experienced by the first frame of \( \text{SR} \) class \( N \) queue) at an output port, the service curve of class \( N \) can be defined as \( \beta_N(t) = \text{idSl}_N[t - T_N]^+ \).

**Proof.** SR class \( N \)'s credit is limited by a higher and lower bound. The lower bound is reached when a maximum-sized frame continues its transmission at zero credit up to completion. The credit at the end of the transmission is \( \text{sdSl}_N(L^N_{\text{max}}/R) \).

For class A and B, the higher bound of the credit is reached after the first sequence (Theorems 1 and 2), which is given by

\[
\text{Credit}_A^{\text{max}} = \text{Credit}_{A_1}^{\text{max}} = \text{idSl}_A \times T_A, 
\]  

(A3)

\[
\text{Credit}_B^{\text{max}} = \text{Credit}_{B_1}^{\text{max}} = \text{idSl}_B \times T_B. 
\]  

(A4)

For class C, the relationship \( \text{Credit}_C^{\text{max}} = \text{idSl}_C \times T_C \) can be achieved from Theorems 1 and 2. SR class C’s credit accumulates up to the higher bound at the rate of \( \text{idSl}_C \) during the transmission time of the maximum numbers of conflicting frames including a non SR frame and an alternation of sequences of A frames and B frames. Assume that \( \text{Credit}_C^{\text{max}} \neq \text{Credit}_{C_1}^{\text{max}} \), then it can be deduced that
$T_C$ is not the worst-case queuing delay of the first frame of SR class C queue, which contradicts the definition of worst-case scenario of Model 1 and Model 2. Thus, we have

$$Credit_{C_{\text{max}}} = Credit_{C_1}^{\text{max}} = idSl_C \times T_C. \quad (A5)$$

So the higher bound is the amount of credit that is accumulated during its worst-case queuing delay time, and it is equal to $(idSl_N \times T_N)$. Then in the worst-case scenario, the credit bounds of class $N$ is expressed as follows:

$$sdSl_N(L_N^{\text{max}} / R) \leq Credit_N \leq (idSl_N \times T_N). \quad (A6)$$

Credit variation of a SR class during $\Delta t$ has been built and proved in ref. [32], and it is given by

$$\Delta Credit_N = idSl_N\Delta t - [R_N(t + \Delta t) - R_N(t)]\leq (idSl_N \times T_N), \quad (A7)$$

where $N'(t)$ is the output data flow. Assume that $R_N(t)$ and $R_N'(t)$ are the input and output cumulative function of flows belonging to Class N, and $s$ represents the initial congestion time of a switch port, then $Credit_N(s) = 0$ and $R_N'(s) = R_N(s)$. Using (A6) and (A7), considering the higher bound of the credit, the credit variation after a period $(t - s)$ is given by

$$Credit_N(t) - Credit_N(s) = idSl_N(t - s) - [R_N(t) - R_N'(s)] \leq (idSl_N \times T_N), \quad (A8)$$

then the last relationship is modified:

$$R_N(t) \geq R_N(s) + idSl_N[(t - s) - T_N]$$
$$= R_N(s) + \beta_N(t - s) \geq \inf_{0 \leq s \leq t} \{R_N(s) + \beta_N(t - s)\}. \quad (A9)$$

If a flow is constrained by $\alpha(t)$ through a network element offering $\beta(t)$, the delay bound $h(\alpha, \beta)$ is the horizontal deviation between $\alpha(t)$ and $\beta(t)$ [25], which is given by

$$h(\alpha, \beta) = \sup_{u \geq 0} \{\inf_{\tau : \tau \geq 0} \{\alpha(s) \leq \beta(s + \tau)\}\}. \quad (A10)$$

In addition, $\alpha^*$ constrains the output flow of the given node and is the arrive curve of the input flow of the next node, which is given by

$$\alpha^* = \sup_{u \geq 0} \{\alpha(t + u) - \beta(u)\}. \quad (A11)$$

Then, the worst-case end-to-end delay of a considered flow is the sum of all delay bounds in network elements along its transmission path.

References
1. Bello, L.L. The case for Ethernet in automotive communications. ACM SIGBED Rev. 2011, 8, 7–15. [CrossRef]
2. ARINC-664(2005), ARINC Specification 664: Aircraft Data Network, Parts 7: Avionics Full-Duplex Switched Ethernet Network; Aeronautical Radio Inc.: Annapolis, MD, USA, 2005.
3. SAE International. Time-Triggered Ethernet; Technical Report AS 6802; SAE International: Warrendale, PA, USA, 2011.
4. Steiner, W.; Hall, B.; Paulitsch, M.; Varadarajan, S. TTEthernet Dataflow Concept. In Proceedings of the 2009 Eighth IEEE International Symposium on Network Computing and Applications, Cambridge, MA, USA, 9–11 July 2009; pp. 319–322.
5. IEEE. IEEE Standard for Local and Metropolitan Area Networks—Audio Video Bridging (AVB) Systems; 802.1BA-2011; IEEE: New York, NY, USA, 2011.

6. IEEE. IEEE Standard for Local and Metropolitan Area Networks—Timing and Synchronization for Time-Sensitive Applications in Bridged Local Area Networks; 802.1AS-2011; IEEE: New York, NY, USA, 2011.

7. IEEE. IEEE Standard for Local and Metropolitan Area Networks—Virtual Bridged Local Area Networks Amendment 14: Stream Reservation Protocol (SRP); 802.1Qat-2010; IEEE: New York, NY, USA, 2010.

8. IEEE. IEEE Standard for Local and Metropolitan Area Networks—Virtual Bridged Local Area Networks Amendment 12: Forwarding and Queuing Enhancements for Time-Sensitive Streams; 802.1Qav-2009; IEEE: New York, NY, USA, 2009.

9. Time-Sensitive Networking Task Group. Available online: http://www.ieee802.org/1/pages/tns.html (accessed on 8 November 2018).

10. Schneele, S.; Geyer, F. Comparison of IEEE AVB and AFDX. In Proceedings of the 2012 IEEE/AIAA 31st Digital Avionics Systems Conference (DASC), Williamsburg, VA, USA, 14–18 October 2012; pp. 7A1-1–7A1-9. [CrossRef]

11. Cummings, R.; Kai, R.; Ernst, R.; Diemer, J.; Ghosal, A. Exploring use of Ethernet for in-vehicle control applications: AFDX, TTEthernet, EtherCAT, and AVB. SAE Int. J. Passeng. Cars Electron. Electr. Syst. 2012, 5, 72–88. [CrossRef]

12. Steiner, W.; Gutierrez, M.; Matyas, Z.; Pozo, F. Current techniques, trends, and new horizons in avionics networks configuration. In Proceedings of the 2015 IEEE/AIAA Digital Avionics Systems Conference (DASC), Prague, Czech Republic, 13–17 September 2015; pp. 6C3-1–6C3-10. [CrossRef]

13. Lee, Y.; Jin, K.; Jeon, J. FlexRay and Ethernet AVB synchronization for high QoS automotive gateway. IEEE Trans. Veh. Technol. 2017, 66, 5737–5751. [CrossRef]

14. Bello, L.L. Novel trends in automotive networks: A perspective on Ethernet and the IEEE Audio Video Bridging. Energ. Technol. Fact. Autom. 2014, 1–8. [CrossRef]

15. Park, J.D.; Cheoun, B.M.; Jeon, J.W. Worst-case analysis of Ethernet AVB in automotive system. In Proceedings of the IEEE International Conference on Information and Automation, Lijiang, China, 8–10 August 2015; pp. 1696–1699.

16. He, F.; Zhao, L.; Li, E. Impact analysis of flow shaping in Ethernet-AVB/TSN and AFDX from network calculus and simulation perspective. Sensors 2017, 17, 1181.

17. Geyer, F.; Heidinger, E.; Schneele, S.; Bodisco, A.V. Evaluation of audio/video bridging forwarding method in an avionics switched ethernet context. In Proceedings of the 2013 IEEE Symposium on Computers and Communications (ISCC), Split, Croatia, 7–10 July 2014; pp. 711–716.

18. Zhao, L.; He, F.; Lu, J. Comparison of AFDX and audio video bridging forwarding methods using network calculus approach. In Proceedings of the 2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC), St. Petersburg, FL, USA, 17–21 September 2017; pp. 1–7.

19. Bordoloi, U.D.; Aminifar, A.; Eles, P.; Peng, Z. Schedulability analysis of Ethernet AVB switches. In Proceedings of the IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA), Chongqing, China, 20–22 August 2014; pp. 1–10.

20. Alderisi, G.; Caltabiano, A.; Vasta, G.; Ianuzzotto, G.; Steinbach, T.; Bello, L.L. Simulative assessments of IEEE 802.1 Ethernet AVB and time-triggered Ethernet for advanced driver assistance system and in-car infotainment. In Proceedings of the 2012 IEEE Vehicular Networking Conference (VNC), Seoul, Korea, 16–18 November 2012; pp. 187–194.

21. Lim, H.T.; Borf, E.; Herrscher, D.; Wolisz, A. Beware of the hidden! How cross-traffic affects quality assurances of competing real-time Ethernet standard for in-car communication. In Proceedings of the 2015 IEEE 40th Conference on Local Computer Networks (LCN), Clearwater Beach, FL, USA, 26–29 October 2015; pp. 1–9.

22. Benmammar, N.; Ridouard, F.; Bauer, H.; Richard, P. Forward end-to-end delay analysis for AFDX networks. IEEE Trans. Ind. Inform. 2018, 14, 858–865. [CrossRef]

23. Li, X. Worst-Case Delay Analysis of Real-Time Switched Ethernet Net-Works with Flow Local Synchronization. Ph.D. Thesis, Université de Toulouse, Toulouse, France, 2013.

24. Berard, B.; Bidoit, M.; Finkel, A.; Laroussinie, F.; Petit, A.; Petrucci, L.; Schnoebelen, P. Systems and Software Verification: Model-Checking Techniques and Tools; Springer: Berlin, Germany, 2001.

25. Boudic, J.Y.L.; Thiran, P. Network Calculus: A Theory of Deterministic Queuing Systems for the Internet; Springer: Berlin, Germany, 2001.
26. Cruz, R.L. A calculus for network delay. I. Network elements in isolation. *IEEE Trans. Inf. Theory* 1991, 37, 114–131. [CrossRef]

27. Bauer, H.; Scharbarg, J.L.; Fraboul, C. Applying trajectory approach with static priority queuing for improving the use of available AFDX resources. *Real-Time Syst.* 2012, 48, 101–133. [CrossRef]

28. Tindell, K.; Clark, J. Holistic schedulability analysis for distributed hard real-time systems. *Microprocess. Microprogr.* 1994, 40, 117–134. [CrossRef]

29. Scharbarg, J.L.; Ridouard, F.; Fraboul, C. A probabilistic analysis of end-to-end delays on an AFDX avionic network. *IEEE Trans. Ind. Inf.* 2009, 5, 38–49. [CrossRef]

30. Bauer, H.; Scharbarg, J.L.; Fraboul, C. Improving the worst-case delay analysis of an AFDX network using an optimized trajectory approach. *IEEE Trans. Ind. Inform.* 2010, 6, 521–533. [CrossRef]

31. Queck, R. Analysis of Ethernet AVB for automotive networks using Network Calculus. In Proceedings of the IEEE International Conference on Vehicular Electronics and Safety (ICVES 2012), Istanbul, Turkey, 24–27 July 2012; pp. 61–67.

32. Boyer, M. Complete modelling of AVB in network calculus framework. In Proceedings of the International Conference on Real-Time Networks and Systems (RTNS 2014), Versailles, France, 8–10 October 2014; pp. 55–64.

33. Zhao, L.X.; Pop, P.; Zheng, Z.; Li, Q. Timing analysis of AVB traffic in TSN networks using network calculus. In Proceedings of the 2018 IEEE Real-time and Embedded Technology and Application Symposium (RTAS), Porto, Portugal, 11–13 April 2018; pp. 25–36.

34. Mohammadpour, E.; Stai, E.; Mohiuddin, M.; Boudec, J.Y.L. End-to-end latency and backlog bounds in time-sensitive networking with credit-based shapers and asynchronous traffic shaping. *arXiv* 2018, arXiv:1804.10608.

35. Li, X.; George, L. Deterministic delay analysis of AVB switched Ethernet networks using an extended trajectory approach. *Real-Time Syst.* 2017, 53, 1–66. [CrossRef]

36. IEEE. *IEEE Standard for Local and Metropolitan Area Networks—Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks; 802.1Q-2011*; IEEE: New York, NY, USA, 2011.

37. Cao, J.; Cuijpers, P.J.L.; Bril, R.J.; Lukkien, J.J. Independent yet tight WCRT analysis for individual priority classes in Ethernet AVB. In Proceedings of the International Conference on Real-Time Networks and Systems (RTNS 2016), Brest, France, 19–21 October 2016; pp. 55–64.

38. Meyer, P.; Steinbach, T.; Korf, F.; Schmidt, T.C. Extending IEEE 802.1 AVB with time-triggered scheduling: A simulation study of the coexistence of synchronous and asynchronous traffic. In Proceedings of the 2013 IEEE Vehicular Networking Conference, Boston, MA, USA, 16–18 December 2014; pp. 47–54.

© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).