The Trajectory approach for AFDX FIFO networks revisited and corrected

Xiaoting Li, Olivier Cros, Laurent George

The Trajectory approach for AFDX FIFO networks revisited and corrected

Xiaoting LI, Olivier CROS
ECE Paris - LACSC
37, quai de Grenelle
75015 Paris, France
Email: {xiaoting.li,cros}@ece.fr

Laurent GEORGE
Université Paris-Est / LIGM
Bat Copernic - 5, bd Descartes
77454 Champs sur Marne, France
Email: lgeorge@ieee.org

***Abstract*—We consider the problem of dimensioning real-time AFDX FIFO networks with a worst-case end-to-end delay analysis. The state-of-the-art has considered several approaches to compute these worst-case end-to-end delays. Among them, the Trajectory approach has received more attention as it has been shown to provide tight end-to-end delay upper bounds. Recently, it has been proved that current Trajectory analysis can be optimistic for some corner cases, leading in its current form, to certification issues. In this paper, we first characterize the source of optimism in the Trajectory approach on detailed examples. Then, we provide a correction to the identified problems. Two problems are solved: the first one is on the definition of the time interval to consider for the worst-case end-to-end response time computation of flows at their source nodes. The second one is on the way that serialized frames are taken into account in the worst-case delay analysis.**

I. Introduction

Reliability and safety are primary constraints in high-critical industrial systems such as aircraft and public transport systems. Furthermore, the needs in these systems are drastically increasing with the development of intelligent transport systems : in addition to safety requirements, we want to add comfort, usability, information management, etc.. This increase in terms of functionalities implies to exchange more and more data and resources through networks. That is why we need specific network architectures that can manage this workload.

High-critical real-time systems often rely on deterministic network architectures to exchange information between end-systems. When throughput is under concern, switched Ethernet networks are more and more considered. Several solutions have been proposed like Synchronous Ethernet (SyncE) [1], Avionics Full DupleX switched Ethernet (AFDX) [2] , Flexible Time Triggered Switched Ethernet (FTT-SE) or Time Triggered Ethernet (TTEthernet) [3]. They are network architectures which are able to bound, with clock synchronization (SyncE, FTT-SE and TTEthernet) or without clock synchronization (AFDX), the end-to-end delay of a message sent through the network, In order to guarantee deterministic communications, several worst-case analysis have been proposed to compute the upper-bounds of the end-to-end communication delays of all the flows sent in the network, and to assure their reliability.

There exist different approaches for computing end-to-end delays in real-time networks. Among them, we can cite:

- The Holistic approach, is a pessimistic way (showed in [4]), and more considered as a *general* approach [5]. It consists of computing end-to-end delay bounds by considering a worst-case scenario (possibly unreachable) in each node visited by a flow.
- The Network Calculus approach [6], [7], which considers worst-case scenario of a flow at each visited node according to service curves. This approach has been improved by considering the serialization effect by [8].
- The Trajectory approach [9], [10], [11], [4], which consists in representing the network not just as a set of nodes, but as a set of flow trajectories through nodes and where each trajectory (associated to one flow) is a potential source of delay for the other flows. This approach is based on the computation of end-to-end delays induced by other frames in the network, and specially the one induced by their serialization on the input ports of switches (called the serialization delay). It was shown that the classical Trajectory approach provide tighter upper bounds than the holistic one but can be pessimistic. Its pessimism has been analyzed in [12].

In [13], [8], the Trajectory approach has been improved by taking into account the serialization of frames sent on the same nodes (called serialization effect in the following). Bauer&al. has shown interesting properties of worst-case delay analysis in the avionics context as it can bring slightly more accurate delay upper bounds than the classical Network Calculus approach. In this work, we focus on the Trajectory approach.

More recently in [14], a counter-example has been given, showing that in its current form the Trajectory approach can bring optimistic results for corner cases. A discussion about the source of pessimism has been given in [14], while no solution has been proposed.

The main objective of this paper is to identify the source of optimism in the Trajectory approach and to propose a solution to it. The first contribution of our work is to characterize this optimism on a detailed example with a detailed analysis. Then, we give a correction of the optimism problem first on the considered example and then by formalizing a correction in the general case. We show that the error rate of the state-of-the-art Trajectory approach w.r.t. our is higher than 10% in the examples we considered.

The paper is organized as follows. II introduces the network

model and flow model considered in the paper. An real-time AFDX network is presented. III presents the classical Trajectory approach and its improvement with the serialization effect. IV shows a counter-example where the Trajectory approach introduces optimism in the computation. V demonstrates the reason of this optimism and proposes a correction to the identified problems in the Trajectory approach. VI concludes the paper.

II. Real-time AFDX network

Avionics Full DupleX Switched Ethernet (AFDX) [2] is a switched Ethernet network which has been defined for the avionics context and developed for modern aircraft such as Airbus A380. AFDX is one of the industrial applications of real-time switched Ethernet networks. The descriptions of the network and flow models are given in the following paragraphs.

A. Network model

In this paper, we study a real-time AFDX switched Ethernet network which is a network able to provide a deterministic data transmission service, through an Ethernet layer. The inputs and outputs of the network are source nodes, called $End\ Systems$ (ES) in the context of AFDX network. These source nodes are interconnected by a full duplex switched Ethernet. We consider a homogeneous single network.

Each source node sends a set of flows through an output port with a buffer supporting First In First Out (FIFO) scheduling. It can be connected to only one port of a switch and each port of a switch can be connected to at most one node. Traffic shaping technique is adopted at the output port of each source node in order to guarantee a minimum interval time between two consecutive frames of a flow (also called gap in AFDX).

Each switch uses a store and forward policy. It has one buffer at each output port which supports the FIFO scheduling. It receives frames from input ports and forwards them to the corresponding output ports based on a static routing table. There is a switching latency (technological latency) to deal with the frame forwarding between an input port and an output port of a given switch and it is upper bounded by a known value sl.

Links between switches are full-duplex, which guarantees no collisions on links. The bandwidth (transmission rate) of the network is denoted by R.

A classical AFDX architecture is depicted in Figure 1. It includes six $End\ Systems$ $ES_1, ES_2, ES_3, ES_4, ES_5$ and ES_6 interconnected by three switches S_1, S_2 and S_3 via full-duplex links.

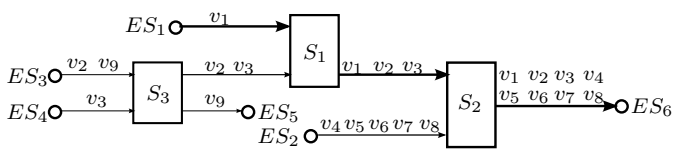

Fig. 1. An illustrative AFDX network example

B. Flow model

A $Virtual\ Link$ (VL) standardized by ARINC-664 is a concept of virtual unidirectional communication channel. Each VL flow v_i transmitted over an AFDX network is characterized by the $Bandwidth\ Allocation\ Gap$ (BAG_i), which is the minimum duration between two consecutive frames of v_i, as well as l_{min_i} and l_{max_i} , which are the minimum and maximum frame lengths.

In this paper, we assume that n VL flows v_i, $i \in \{1, 2, ..., n\}$ are transmitted over the AFDX network in order to exchange data. A VL flow v_i has a path defined by a sequence of output ports $\mathcal{P}_i = \{first_i, ..., last_i\}$, where the $first_i$ is the source node of flow v_i and $last_i$ is the last visited output port of flow v_i along the path $\mathcal{P}_i$. For instance, v_1 in Figure 1 follows the path $\mathcal{P}_1 = \{ES_1, S_1, S_2\}$. Its source node is $first_1 = ES_1$. Its last visited output port is $last_1 = S_2$. This flow model can be extended to multicast flows where each path is associated to one receiver. For the sake of simplicity, we only consider unicast flows in this paper.

Sporadic VL flows are transmitted over the network. The temporal features of a given VL flow v_i are defined by the following parameters:

- the minimum inter-frame duration T_i, which corresponds to the value of BAG_i associated to flow v_i, and
- the worst-case transmission time (WCTT) of one frame f_i of flow v_i. It is denoted by C_i and can be computed by $C_i = \frac{l_{max_i}}{R}$. Indeed, instead of defining a specific WCTT for each node and each frame, we define the worst-case transmission time of a frame transmitted at all nodes.

In this paper, for the sake of simplicity, we do not consider release jitter for each flow, but a release jitter can be easily taken into account in the computation of worst case end-to-end delays.

III. Understanding the Trajectory approach

The Trajectory approach allows us to compute a bound on the worst-case transmission delay of any flow transmitted on a switched Ethernet network. It has been first proposed for the First In First Out (FIFO) scheduling in [9] and extended for non-preemptive Fixed Priority (FP) scheduling in [10], [11]. In the following paragraphs, we first recall the classical Trajectory approach principles. Then, we present the integration of serialization effect in the Trajectory approach proposed in the state-of-the-art.

A. Notations

For a flow v_i following a path $\mathcal{P}_i = \{first_i, ..., last_i\}$, we focus on a frame fi which arrives at the output port of $first_i$ at time t. The following notations are given for the computation.

- h represents any output port of a node or of a switch in our network.
- $h + 1$ (resp. $h - 1$) represents the previous (resp. following) output port of the output port h.

- bp^h represents the busy period of frame fi at the output h. A busy period means a time interval during which there is no idle time.
- FT^h_{min} is the smallest WCTT among the frames transmitted during bp^h. It is computed by $FT^h_{min} = \min_{h \in \mathcal{P}_j}(C_j)$.
- M^h_i is considered as the earliest arrival time of the first packet that will delay frame f_i on the output port h. It is computed by:

$$M^h_i = \sum_{k=first_i}^{h-1} (FT^k_{min} + sl)$$

- $S^h_{max_i}$ and $S^h_{min_i}$ are the maximum and the minimum delay experienced by frame f_i from its source node $first_i$ to the output port h.
- t is the arrival time of a frame under study f_i at the input port of node $first_i$. This arrival time should be considered w.r.t. a reference time 0, which is our time origin. Indeed, in the architectures we want to focus on, frames that do not necessarily arrive at time 0, but can be delayed and arrive later.
- We define a time interval used to compute the maximal number of frames about to delay our focused message, $A_{i,j}$, as defined in [14]

$$A_{i,j} = Smax^h_i - Smin^h_j - M^h_i + Smax^h_j \quad (1)$$

- $(a)^+ = \max\{a, 0\}$.

Let's illustrate these definitions by an example. We want to study the following elementary network

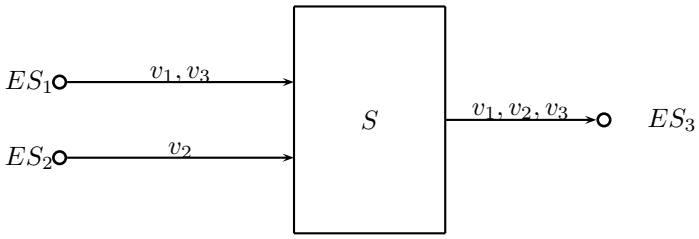

Fig. 2. Understanding the standard notations for the trajectory approach, given three flows v_1, v_2, v_3 arriving on the same switch S

We consider that the first input port of S receives two different flows, v_1 and v_3. v_1 emits a frame f_1, with $C_1 = 30$ and $T_1 = 4000$, and v_3 sends f_3, with $C_3 = 20$ and $T_3 = 4000$. Following the same way, the flow v_2 sends only one packet through the network, called f_2, with $C_2 = 40$ and $T_2 = 90$. We want to focus on the packet f_3, which arrives on ES_1 at time $t = 20$. Considering this system, we can represent its behavior with the figure 3.

B. Calculating an end-to-end delay

Actually, the computation of the Trajectory approach in FIFO context is summarized by the combination of 4 different

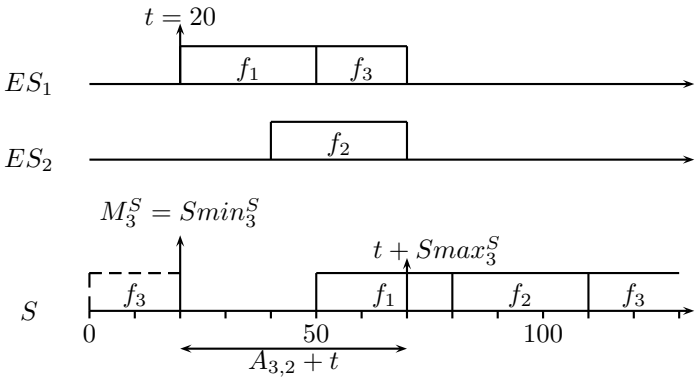

Fig. 3. Understanding the standard notations for the trajectory approach

terms. Combining these terms allows us to calculate, for a specific frame f_i released at time t, of a flow v_i, the latest starting time $W^{last_i}_{i,t}$ from its last visited output port $last_i$, corresponding to the last node of the network belonging to $\mathcal{P}_i$. We have the following expression :

$$W^{last_i}_{i,t} = \sum_{\substack{j \in \{1,\dots,n\} \\ \mathcal{P}_i \cap \mathcal{P}_j \neq \emptyset}} \left(1 + \left\lfloor \frac{t + A_{i,j}}{T_j} \right\rfloor\right)^+ \cdot C_j \quad (2)$$

$$+ \sum_{h \in \mathcal{P}_i / \{last_i\}} \left(\max_{\substack{j \in \{1,\dots,n\} \\ h \in \mathcal{P}_j}} (C_j) \right) \quad (3)$$

$$+ (|\mathcal{P}_i| - 1) \cdot sl \quad (4)$$

$$- C_i \quad (5)$$

- Term 2 is the delay due to competing flows which delay v_i along its trajectory as well as the transmission delay generated by v_i itself.
  $A_{i,j}$ enters in the composition of the term $(1 + \lfloor \frac{t + A_{i,j}}{T_j} \rfloor)^+$ which is the maximal number of frames generated by a flow v_j that can delay flow v_i
  To evaluate the end-to-end delay of a flow v_i, we need the value of $A_{i,j}$ for all encountered flow v_j. To do this, we first calculate the two terms $Smax^h_i and M^h_i$ which are the same for each flow v_j. Respectively, M^h_i is the earliest arrival time of the first packet which delays i on a node h , and $Smax^h_i$ (see Figure 3).
- Term 3 is the transition cost from one busy period to the next one. When each frame sequence is transmitted from one busy period to the following one, there is a transition delay which is taken as the largest frame transmission time in the frame sequence. For example, in Figure 4 the frame sequence at node ES_1 is composed by frames f_0 and f_1. When they are transmitted to node S, the transmission time of frame f_0 is taken as the transition cost since it is larger than that of frame f_1.
- Term 4 is the switching latencies along the considered path, considered as a worst-case constant, identical for each link. For each encountered node we add a switching latency to the end-to-end delay of a message f_i, corresponding to electronic delay. So, in an entire network, the induced latency corresponding

to f_i corresponds to sl, multiplied by the number of encountered nodes, represented by $|\mathcal{P}_i|$. As we consider the path from the first switch to the last one of the network, we need to minus our end-to-end delay of $1 * sl$: given the point that we want to evaluate the end-to-end delay of a flow in a network, we can focus on the delay between its entry point and the output of the last switch in the network. In figure 1, this latency corresponds to the delay between S_2 and ES_6.

- Term 5 is subtracted because $W_{i,t}^{last_i}$ is the latest starting time at $last_i$. Indeed, $W_{i,t}^{last_i}$ corresponds to the delay between time t and the beginning of the transmission of i in $last_i$(and not the end of the transmission time). So, the value of C_{last_i} needs to be subtracted from the value of $W_{i,t}^{last_i}$.

From the latest starting time, the worst-case end-to-end delay upper bound of the flow v_i calculated by the Trajectory approach is obtained by [14]:

$$R_i = \max_{0 \leq t \leq \mathcal{B}_i} \{W_{i,t}^{last_i} + C_i - t\} \quad (6)$$

where $\mathcal{B}_i = \sum_{\substack{j \in [\![1,n]\!] \\ \mathcal{P}_i \cap \mathcal{P}_j \neq \emptyset}} \lceil \frac{\mathcal{B}_i}{T_j} \rceil \cdot C_j$. (see Figure 4)

B_i represents the largest possible length of the busy period in each encountered node of the network.

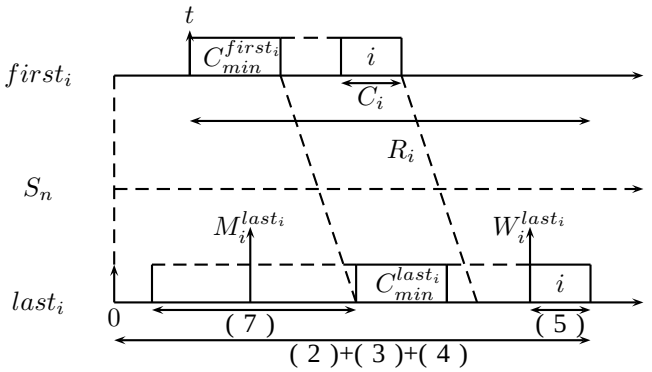

Fig. 4. Calculating the latest starting time of i with the trajectory approach

C. The serialization of frames

The classical Trajectory approach considers that competing frames of different flows can arrive at the output port at the same time. However, for frames transmitted from the same input link, their transmissions are necessarily serialized (physical constraint) and they cannot arrive at the output port at the same time. In the following paragraphs, the serialization effect and its solution to the Trajectory approach proposed in [8] are briefly recalled.

The frame serialization exists at each switch h as illustrated in Figure 5. Flows crossing an output port h are transmitted from $k_h + 1$ input links IP_k^h ($k \in \{0, ..., k_h\}$). For a frame f_i of flow v_i, it crosses h from the input link IP_0^h, which is the output link OP^{h-1}, to the output link OP^h. There are other k_h input links IP_k^h ($k \in \{1, ..., k_h\}$) which transmit competing frames to the output link OP^h. These frames can delay frame f_i in the busy period bp^h.

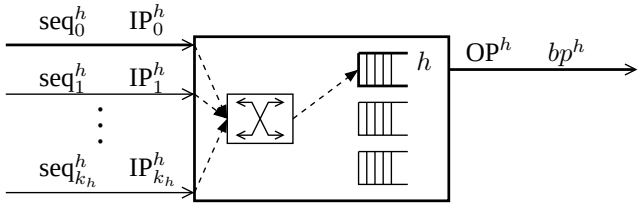

Fig. 5. Illustration on an output link OP^h and $k_h + 1$ input link IP_k^h

In [8], an optimization on the Trajectory approach has been proposed by taking into account the serialization effect. An illustration of the optimization is shown in Figure 6. Due to the FIFO scheduling at the output buffer of h, any frame arriving later than the arrival time of frame f_i (θ in Figure 6) at node h cannot delay f_i. Then the delay of f_i is maximized by postponing the last frame arrival of each frame sequence from each input link IP_k^h ($k \in \{1, ..., k_h\}$) till the time θ. It has been illustrated in [8] that at the output port OP^h, there can be frames transmitted before the first frame arrival from IP_0^h. These frame transmissions do not delay frame f_i and should be taken into account in the delay computation. The associated duration is denoted by $\Delta_{i,t}^h$.

From [8], the value of $\Delta_{i,t}^h$ is minimized when the first frame of IP_0^h is the smallest frame transmitted by IP_0^h and the first frame of IP_k^h is the largest frame transmitted by the corresponding input link IP_k^h. The scenario is illustrated in Figure 6 and the computation is given by:

$$\Delta_{i,t}^h = \max_{x \in \{1,...,k_h\}} \left\{ \sum_{v_j \in \text{IP}_x^h} \left(\left\lfloor 1 + \frac{t + A_{i,j}}{T_j} \right\rfloor \cdot C_j \right) - \max_{v_j \in \text{IP}_x^h} (C_j) \right\} - \sum_{v_j \in \text{IP}_0^h} \left(\left\lfloor 1 + \frac{t + A_{i,j}}{T_j} \right\rfloor \cdot C_j \right) - \min_{v_j \in \text{IP}_0^h} (C_j))$$

Then serialization factor at each visited output port along the studied path $\mathcal{P}_i$ is given by:

$$\sum_{h \in \mathcal{P}_i / \{first_i\}} (\Delta_{i,t}^h) \quad (7)$$

From [8], it has been shown that $W_i^{last_i}$ including the serialization factors is given by:

$$\begin{aligned} W_{i,t}^{last_i} = & \sum_{\substack{j \in \{1,...,n\} \\ \mathcal{P}_i \cap \mathcal{P}_j \neq \emptyset}} \left(1 + \left\lfloor \frac{t + A_{i,j}}{T_j} \right\rfloor \right)^+ \cdot C_j \\ & + \sum_{h \in \mathcal{P}_i / \{last_i\}} \left(\max_{\substack{j \in \{1,...,n\} \\ h \in \mathcal{P}_j}} (C_j) \right) \\ & + (|\mathcal{P}_i| - 1) \cdot sl \\ & - \sum_{h \in \mathcal{P}_i / \{first_i\}} (\Delta_{i,t}^h) \\ & - C_i \end{aligned}$$

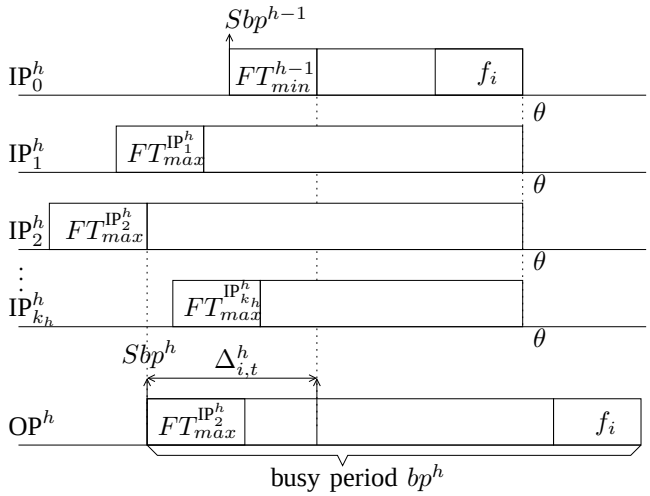

Fig. 6. Illustration of term $\Delta_{i,t}^h$

This formula has been shown to be optimistic in particular case [14]. We will now illustrate why it can be optimistic.

IV. Problem: counter-example with optimism

The Trajectory approach considers the worst-case scenario that can happen to a frame along its trajectory in order to guarantee the delay upper bound computation. Recently, a counter-example has been shown in [14] to point out that for some corner cases, the Trajectory approach can lead to optimistic computed delay upper bounds. This example is given in Figure 7.

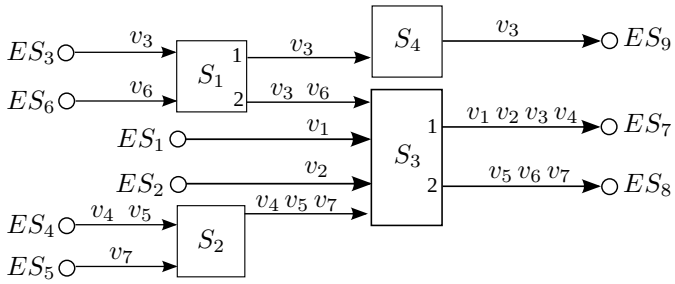

Fig. 7. An illustrative AFDX example

It means that the exact worst-case delay of a frame can be larger than the one obtained with the current state-of-the-art Trajectory approach. For example, for frame f_1 of flow v_1, which can be delayed by frames of flows v_2, v_3, v_4 at the output of switch S_3, its exact worst-case delay is $180\ \mu s$ as shown in Figure 8. However, the delay upper bound of frame f_1 computed by the Trajectory approach is $160\ \mu s$, which is lower than the exact worst-case delay. Therefore the Trajectory approach gives an optimistic result ($20\ \mu s$ of pessimism) in this case.

The Trajectory approach considers all the possible values of time t in the time interval $[0, B_i]$. Time t corresponds to the arrival time of frame f_i at its source node $first_i$, and each value of t corresponds to a release time scenario for frame f_i that must be analyzed. Therefore, the Trajectory approach searches a time t in a given time interval leading to the maximum value of R_i in Formula 6, which corresponds to the worst-case scenario. In the example, the computed delay upper bound R_1 is obtained when $t = 0$. The corresponding scenario is illustrated in Figure 9.

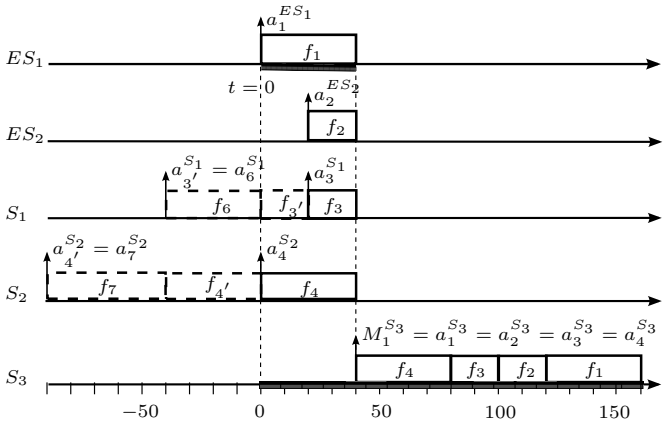

Fig. 9. Delay of frame f_1 computed by the Trajectory approach

In the case of frame f_1, since it is the only frame transmitted from ES_1, we have $M_1^{S_3} = t + S_{max_1}^{S_3} = 40\ \mu s$, giving the interval $[M_1^{S_3}, t + S_{max_1}^{S_3}]$ which leads to only a time instant $40\ \mu s$. It means that for flows v_2, v_3 and v_4 competing with flow v_1 at the output port of S_3 (port 1 in Figure 7), only the frames arriving at this output port at time $t = 40\ \mu s$ can delay frame f_1. Therefore, for each flow there is at most one frame that can delay frame f_1, among frame f_2, frame f_3 and frame f_4 as shown in Figure 9.

However, as shown in Figure 8, frames $f_{3'}$ and $f_{4'}$ which can actually delay frame f_1 are not taken into account since they arrive before time $M_1^{S_3} = 40\ \mu s$. Therefore, the interval $[M_1^{S_3}, t + S_{max_1}^{S_3}]$ is not correct (under-estimated). Indeed, in the example $t = 0$ does not lead to the worst-case scenario.

Actually, for $t = 40\ \mu s$, the Trajectory approach computes a longer interval $[M_1^{S_3}, t + S_{max_1}^{S_3}] = [40, 80]$ as shown in Figure 10. In this case, both frames f_3 and $f_{3'}$ as well as frames f_4 and $f_{4'}$ are taken into account in the computation. However, the computation still gives an optimistic result which is $R_1(t = 40) = 140\ \mu s$.

The computation subtracts a serialization factor $\Delta_1^{S_3} = 40\ \mu s$ (Term 7) since the transmission of frame $f_{4'}$ at S_3 does not delay the frame f_1 due to the serialization effect. Meanwhile, a time $t = 40\ \mu s$ is also subtracted in Formula 6 since frame f_1 arrives at time t and therefore the time interval $[0, t]$ is not part of the delay of frame f_1. In fact, these two parts of subtractions overlap and the same time interval is subtracted twice from the computation (detailed explanation will be given in the next section). It then results in an optimistic computed delay $140\ \mu s$ of frame f_1, as illustrated in Figure 10.

In the next paragraphs, a detailed analysis of the optimism will be given and a solution to the optimism problem will be proposed.

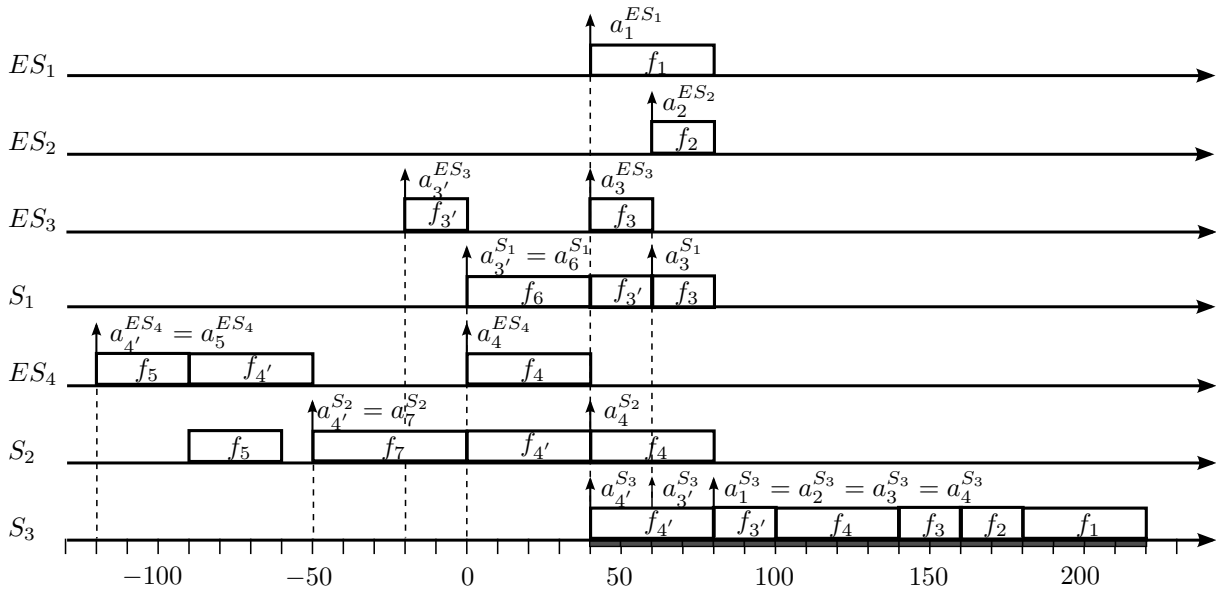

Fig. 8. Worst-case ETE delay of frame f_1 at time $t = 40$

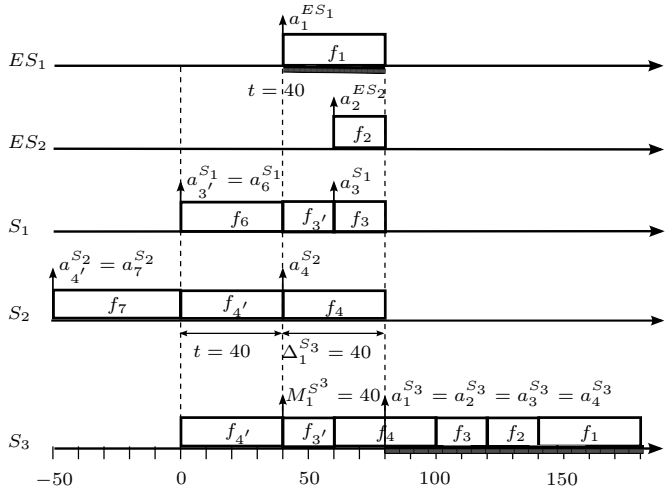

Fig. 10. Delay of frame f_1 computed by the Trajectory approach at time $t = 40\ \mu s$

V. A CORRECTION OF THE TRAJECTORY APPROACH

As shown in the previous paragraphs, for a frame f_i of a flow v_i the problem of the under-estimated interval $[M_i^h, t + S_{max_i}^h]$ disappears with the examinations of all the possible values of t. However, at the same time subtracting time t from the delay of frame f_i (Equation 6) can overlap with part of the serialization factor (Term 7). The following paragraphs first examine the optimism of the Trajectory approach by an illustrative example, with both the classical approach and the serialization optimization respectively. Then the analysis is developed in the general case and a correction of the optimism problem is proposed.

A. Computation with the classical approach

In order to explain this problem in details, let us consider the example given in Figure 1. The flow temporal characteristics are given in Table I. In the example, the switching latency is considered as null.

v_i	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9
$C_i(\mu s)$	40	40	40	40	40	40	40	40	40
$T_i(\mu s)$	4000	120	4000	320	4000	4000	4000	4000	4000

TABLE I. FLOW TEMPORAL CHARACTERISTICS OF THE NETWORK EXAMPLE

Frame f_1 of flow v_1 is under study. It follows the path $\mathcal{P}_1 = \{ES_1, S_1, S_2\}$ and it can be delayed by frames of flows v_2 and v_3 at the output port of S_1 as well as by flows v_4, v_5, v_6, v_7 and v_8 at the output port of S_2.

One worst-case scenario of frame f_1 is given in Figure 11 when $t = 120\ \mu s$. The worst-case delay of frame f_1 is 320 μs. As illustrated in this figure, flow v_2 has two frames f_2 and $f_{2'}$ delaying frame f_1 at the output port of S_1, and flow v_4 has two frames f_4 and $f_{4'}$ delaying frame f_1 at the output port of S_1.

Let us now consider the Trajectory approach. Since flow v_1 is the only flow emitted by ES_1, then $M_1^{S_1} = 40\ \mu s$ and $S_{max_1}^{S_1} = S_{min_1}^{S_1} = C_1 = 40\ \mu s$. First, for each flow encountered by v_1, we need to calculate the value of $A_{i,j}$. For this, we apply the Equation 1. Flow v_2 is emitted by ES_2 and can be delayed by frames of flow 9 and then it can be delayed by frames of flow v_3 at the output port of S_3, which gives $S_{max_2}^{S_1} = 160\ \mu s$ and $S_{min_2}^{S_1} = 80\ \mu s$. Therefore, the computation gives $A_{1,2}$ as follows:

$$\begin{aligned} A_{1,2} &= S_{max_1}^{S_1} - S_{min_2}^{S_1} - (M_1^{S_1} - S_{max_2}^{S_1}) \\ &= 40 - 80 - (40 - 160) \\ &= 80\ \mu s \end{aligned}$$

In the same way, the values of $A_{1,3} = 40\ \mu s$ and $A_{1,4} = A_{1,5} = A_{1,6} = A_{1,7} = A_{1,8} = 200\ \mu s$ are obtained. So, we obtain the following results :

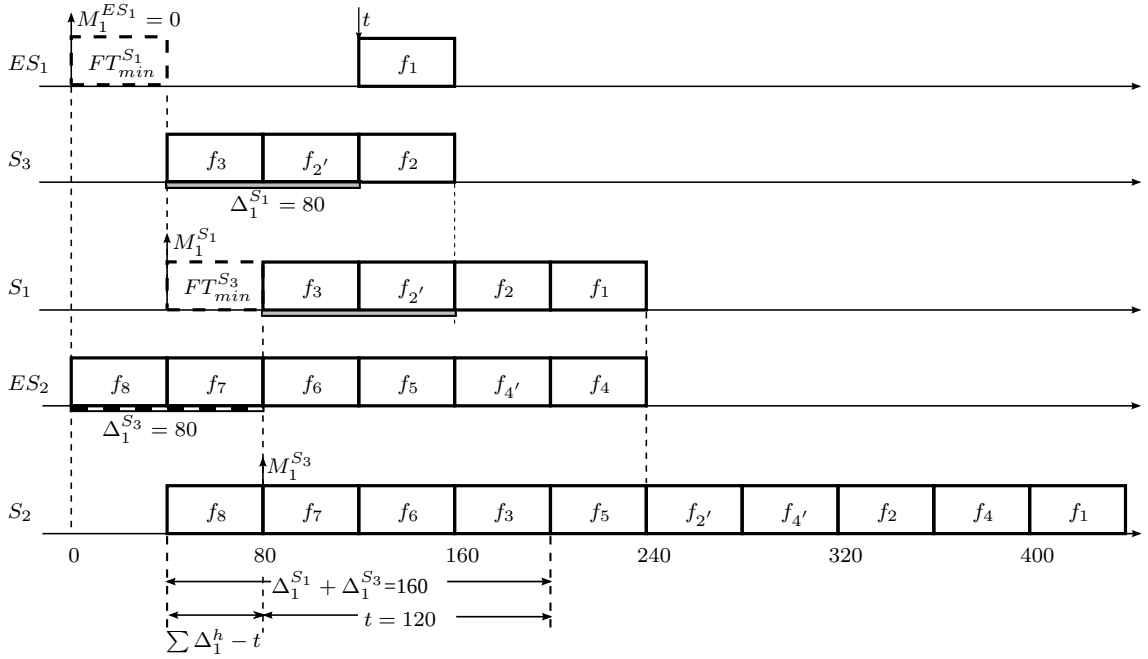

Fig. 11. Worst-case scenario of frame f_1

j	1	2	3	4	5	6	7	8	9
$A_{1,j}$	0	40	200	200	200	200	200	200	200

TABLE II. VALUES OF $A_{i,j}$ FOR EACH FLOW

Therefore at time $t = 120\ \mu s$, the $W_{1,120}^{S_2}$ is given by:

$$\begin{aligned} W_{1,120}^{S_2} &= \sum_{j\in\{1,\ldots,8\}} (1+\lfloor\frac{120+A_{i,j}}{T_j}\rfloor)^+ \cdot C_j \\ &\quad + \sum_{h\in\{ES_1,S_1\}} (\max_{\substack{j\in\{1,\ldots,8\}\\ h\in\mathcal{P}_j}}(C_j)) \\ &\quad - \sum_{h\in\{S_1,S_2\}} (\Delta_{1,120}^h) \\ &\quad -C_1 \\ &= (1+\lfloor\frac{120}{4000}\rfloor)^+ \cdot 40 + (1+\lfloor\frac{120+80}{120}\rfloor)^+ \cdot 40 \\ &\quad +(1+\lfloor\frac{120+40}{4000}\rfloor)^+ \cdot 40 + (1+\lfloor\frac{120+200}{320}\rfloor)^+ \cdot 40 \\ &\quad +(1+\lfloor\frac{120+200}{4000}\rfloor)^+ \cdot 40 \times 4 \\ &\quad +40+40-\Delta_{1,120}^{S_1}-\Delta_{1,120}^{S_2}-40 \\ &= 440-\Delta_{1,120}^{S_1}-\Delta_{1,120}^{S_2} \end{aligned}$$

First, let us consider the classical Trajectory approach without serialization factor. In this case, the $W_{1,120}^{S_2} = 440\ \mu s$ and then $R_1 = W_{1,120}^{S_2} + C_1 - 120 = 360\ \mu s$, which is pessimistic compared to the worst-case delay 320 μs in Figure 11. This scenario is illustrated in Figure 12. The pessimism is generated because the computation is done based on pessimistic assumption that all the frames of different flows arrive at the same time even if they are serialized by some input link (the serialization factor is not considered).

It has been illustrated in Figure 12 that although frames $f_{4'}$, f_5, f_6, f_7 and f_8 are considered arriving at the same time, frames f_6, f_7 and f_8 do not delay frame f_1. Indeed, the

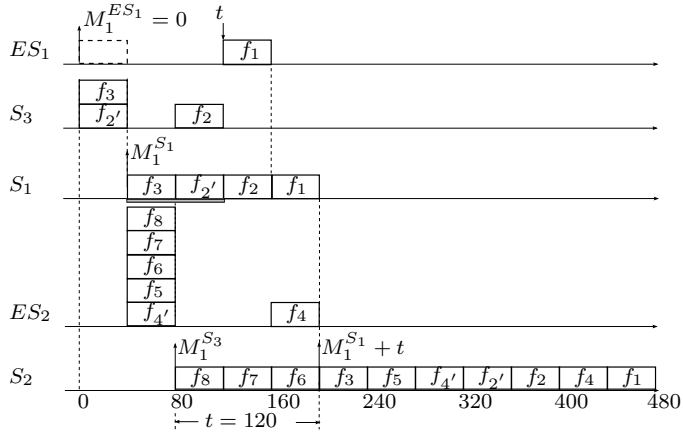

Fig. 12. One worst-case scenario of frame f_1 considered by the classical Trajectory approach

computation of $W_1^{S_2}$ is based on the maximized busy period bp^{S_2} which starts at time $M_1^{S_2}$ (shown in Figure 12). Since frame f_1 arrives at ES_1 at time t, there is a time interval of t which does not contribute to the delay of frame f_1. It means that during the busy period bp^{S_2}, the frame transmission during $[M_1^{S_2}, M_1^{S_2}+t]$ do not delay frame f_1, which are exact the frames f_6, f_7 and f_8 in the example.

Suppose that the delay upper bound of frame f_i is obtained when f_i arrives at $first_i$ at a given time t. The corresponding scenario of R_i is illustrated by the first bar in Figure 13 where the time interval t is subtracted since it is not part of delay. Then the shadow part in the bar represents the delay R_i.

Since the classical Trajectory approach considers that the competing frames arrive after time M_i^h at each node h and there is no frame transmitted before M_i^h, then the subtrac-

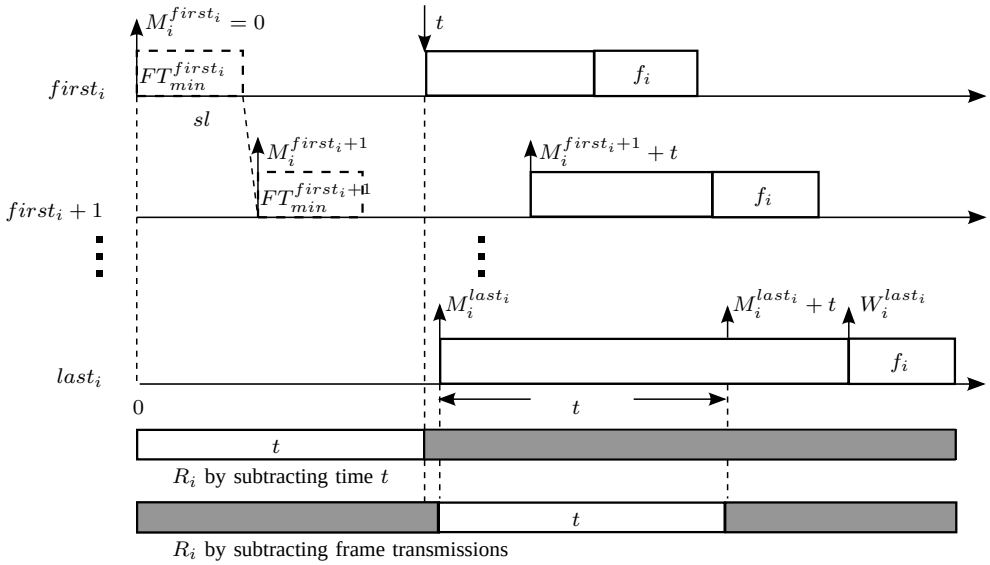

Fig. 13. General scenario of frame f_i of the classical Trajectory approach

tion of the time interval t in Formula 6 is equivalent to the subtraction of the WCTT of frames transmitted between $[M_i^{last_i}, M_i^{last_i} + t]$ at the last visited node $last_i$. The corresponding scenario is also illustrated by the second bar in Figure 13 where the time interval t (white bar) is moved between $M_i^{last_i}$ and $M_i^{last_i} + t$.

B. Computation with Serialization effect

As explained in Section III-C, frames transmitted from the same input link are serialized and cannot arrive at the output at the same time. Therefore, the scenario shown in Figure 12 where frames $f_{4'}$, f_5, f_6, f_7 and f_8 arrive at S_2 at the same time is impossible and pessimistic.

Indeed, as shown in Figure 11, at the output port of S_1, frames f_2, $f_{2'}$ and f_3 arrive from the same link $\{S_3, S_1\}$. Since frame f_1 is the only frame transmitted from ES_1, only one frame from the input link $\{S_3, S_1\}$ can delay frame f_1. In this illustration, it is frame f_2. Then the serialization factor is computed by $\Delta_{1,120}^{S_1} = C_3 + C_2 = 80\ \mu s$. Similarly, at the output port of S_2, the serialization factor is computed by $\Delta_{1,120}^{S_2} = 80\ \mu s$. Then by taking into account the serialization effect, we have $W_{1,120}^{S_2} = 440 - \Delta_{1,120}^{S_1} - \Delta_{1,120}^{S_2} = 280\ \mu s$ and $R_1(120) = W_{1,120}^{S_2} + C_1 - t = 200\ \mu s$, which is optimistic compared to the worst-case delay $320\ \mu s$ in Figure 11.

The optimism is introduced since the subtraction of serialization factors is partially overlapped with the subtraction of the time interval t in Equation 6. The classical approach subtracts the transmissions of frames f_6, f_7 and f_8 by the subtraction of time t as shown in Figure 12, while the serialization factors subtract the transmissions of frames f_3, f_6, f_7 and f_8 as illustrated in Figure 11. In this case, the frames f_6, f_7 and f_8 are actually subtracted twice and therefore leading to the optimistic result.

In the general case, the serialization factors subtract frame transmissions of $\sum_{\substack{h \in \mathcal{P}_i \\ h \neq first_i}} \Delta_{i,t}^h$ from the computation at the last visited output $last_i$. For the purpose of illustration, we first consider the scenario in Figure 6. The starting instant of the busy period bp^h is denoted Sbp^h as shown in Figure 6. Since the input IP_0^h is the output busy period bp^{h-1} in the previous output port $h-1$, then Sbp^{h-1} is also indicated in Figure 6. Therefore, we have:

$$Sbp^h = Sbp^{h-1} + FT_{min}^{h-1} + sl - \Delta_{i,t}^h \tag{8}$$

At the source node $first_i$, the worst-case scenario is when all the frames arrive at the same time as frame f_i and delay f_i. Then Sbp^{first_i} is equal to the arrival time of f_i, i.e. $Sbp^{first_i} = t$. Based on the Equation 8, the computation propagates till $last_i$ where we have:

$$\begin{aligned} Sbp^{last_i} &= t + \sum_{\substack{h \in \mathcal{P}_i \\ h \neq last_i}} (FT_{min}^h + sl) - \sum_{\substack{h \in \mathcal{P}_i \\ h \neq first_i}} \Delta_{i,t}^h \\ &= t + M_i^{last_i} - \sum_{\substack{h \in \mathcal{P}_i \\ h \neq first_i}} \Delta_{i,t}^h \end{aligned}$$

This scenario is illustrated in Figure 14 which shows that at the last visited output $last_i$, the frame transmissions between Sbp^{last_i} to $t + M_i^{last_i}$ are subtracted from the computation.

In the given example, we want to evaluate the error rate induced by the optimism in the Trajectory approach(compared to the standard hand-built method). So, we make the following evaluation :

$$E(ErrorRate) = (320 - 280) * 100/320 = 12,5\%.$$

In this simple example, the optimism induced represents more than 10 % of the result. In strong-constrained domains concerned by network and real time, like aeronautics or public transports, which have industrial very-high needs in terms of

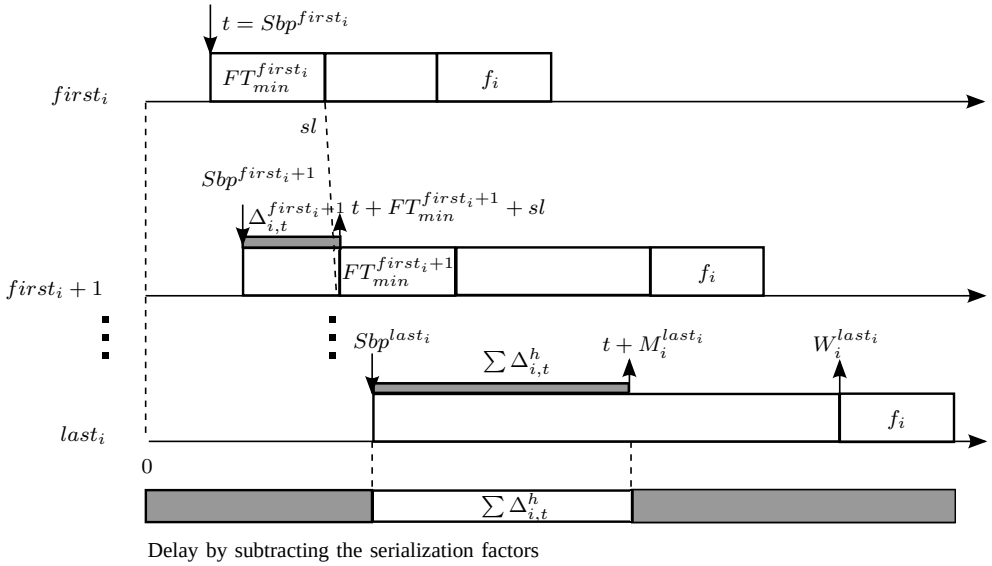

Fig. 14. General scenario of frame f_i considered by the Trajectory approach with serialization effect

reliability and accuracy, such a high error rate directly corresponds to invalidating the method of the Trajectory approach. That is why we need to find a solution to this optimism.

C. A solution to optimism

Let us focus on the comparison between Figure 13 where a time t is subtracted from $M_i^{last_i}$ to $M_i^{last_i} + t$ and Figure 14 where the serialization factors are subtracted from Sbp^{last_i} to $M_i^{last_i} + t$. These exists an overlapped time interval $[\max(M_i^{last_i}, Sbp^{last_i}), M_i^{last_i} + t]$ which is subtracted twice in the computation of delay upper bounds, and therefore leading to the optimistic results. In the example, this time interval is $[80, 200]$ as shown in Figure 11.

One solution to solve this problem is to take into account the subtraction of time t in the serialization factors. More precisely, instead of subtracting the time of frame transmissions before time $t + M_i^{last_i}$, consider to subtract it at the last visited node the frame transmissions before $M_i^{last_i}$ since only this part is not considered by the classical Trajectory approach.

Theorem 1. *The serialization effect in the Trajectory approach is taken into account by a serialization correction:*

$$\left(\sum_{\substack{h \in \mathcal{P}_i \\ h \neq first_i}} (\Delta_{i,t}^{h}) - t \right)^{+} \tag{9}$$

Proof: The serialization correction considers the serialization effect by taking the difference between the serialization factors and the subtracted time t. More precisely, two cases are considered and illustrated below.

First, we considers the case $\sum_{\substack{h \in \mathcal{P}_i \\ h \neq last_i}} (\Delta_{i,t}^{h}) \leq t$, and the corresponding scenario is illustrated in Figure 15. Since the overlapped time interval $[Sbp^{last_i}, M_i^{last_i} + t]$ is covered by the time interval $[M_i^{last_i}, M_i^{last_i} + t]$, the serialization effect is taken into account by subtracting time t in Formula 6. Therefore, the serialization correction is taken as value 0.

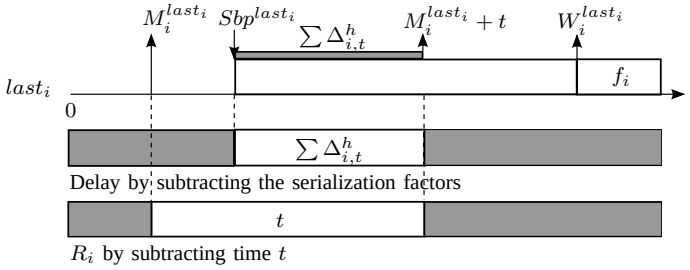

Fig. 15. Scenario of $\sum \Delta_{i,t}^{h} \leq t$

Second, we consider the case $\sum_{\substack{h \in \mathcal{P}_i \\ h \neq last_i}} (\Delta_{i,t}^{h}) > t$, and the corresponding scenario is illustrated in Figure 16. The frame transmissions before $M_i^{last_i}$ do not delay frame f_i and their time needs to be subtracted, which corresponds to the serialization correction. The rest part of the serialization factors is equal to the overlapped time interval $[M_i^{last_i}, M_i^{last_i} + t]$ and it is subtracted in Formula 6.

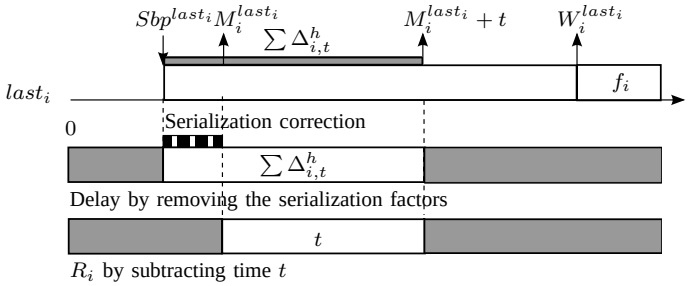

Fig. 16. Scenario of $\sum \Delta_{i,t}^{h} > t$

Therefore, the Term 9 is a correction of Term 7 of serialization factors. ■

As it has been demonstrated, the overlapped time interval $[\max(M_i^{last_i}, Sbp^{last_i}), M_i^{last_i} + t]$ is subtracted twice in the computation and it can lead to optimistic result of delay upper bound. The optimism introduced by the Trajectory approach presented in III is then computed by:

$$\min(\sum_{\substack{h \in \mathcal{P}_i \\ h \neq first_i}} \Delta_{i,t}^h, t)$$

The overlapped time interval is equal to 0 only when $t = 0$ or when $\sum_{\substack{h \in \mathcal{P}_i \\ h \neq first_i}} \Delta_{i,t}^h = 0$.

In the example, the value of the serialization correction is given by:

$$-((\Delta_{1,120}^{S_1} + \Delta_{1,120}^{S_2}) - t)^+ = -((80+80) - 120)^+ = -40 \ \mu s$$

It is illustrated by the transmission time of frame f_8 in Figure 11. Thus, the $W_{1,120}^{S_2}$ computed by the Trajectory approach with the serialization correction is:

$$W_{1,120}^{S_2} = 440 - 40 = 400 \ \mu s$$

The delay upper bound of frame f_1 is then obtained by

$$R_1 = 400 + 40 - 120 = 320 \ \mu s$$

which is the exact worst-case delay of frame f_1 in the example.

VI. Conclusion

In this paper, we have considered the problem of computing worst-case end-to-end delays of flows sent on an AFDX FIFO network. We considered the Trajectory approach, known to provide tight worst-case end-to-end delay upper bounds. Recently, it has been shown that this approach can lead to optimistic end-to-end delays thus leading to certification issues. Our goal was to characterize this optimism problem and to provide a solution to it.

We presented the different sources of optimism in the Trajectory approach on pathological examples, including the underestimation of the release time interval used to compute the worst-case end-to-end delay of a flow and a problem on the computation of the serialization factors for flows sent on the same link. The error rate found in our examples can reach 10%, a significant error. We then solved the optimism problem in the general case. As a perspective, we would like to compute the average error rate for random sets of flows.

References

[1] P. R.Santos, A.Vieira, "Flexible, efficient and robust real-time communication with server-based ethernet switching," *Factory Communication Systems (WFCS), 2010 8th IEEE International Workshop*, pp. 131–140, 2010.

[2] *ARINC 664*, ACCE Std. 664, 2002-2008.

[3] W. Steiner and G. Bauer, "Ttethernet dataflow concept," *Network Computing and Applications, 2009. NCA 2009.*, pp. 319–322, 2009.

[4] S. Martin, P. Minet, and L. George, "End-to-end response time with fixed priority scheduling: trajectory approach versus holistic approach," Feb. 2004, pp. 37–56.

[5] M. H. J.J. Gutirrez, J.C. Palencia, "Holistic schedulability analysis for multipacket messages in afdx networks," in *Real-Time Systems*, Mar. 2014, pp. 230–269.

[6] J.-Y. L. Boudec, "Application of network calculus to guaranteed service networks," *IEEE Trans. Inf. Theory*, vol. 44, no. 3, pp. 1087–1096, May 1998.

[7] J.-P. Georges, E. Rondeau, and T. Divoux, "Evaluation of switched Ethernet in an industrial context by using the network calculus," in *4th IEEE Int. Workshop on Factory Communication Systems(WFCS)*, Västerås, Sweden, Aug. 2002, pp. 19–26.

[8] H. Bauer, J.-L. Scharbarg, and C. Fraboul, "Improving the worst-case delay analysis of an AFDX network using an optimized Trajectory approach," *IEEE Trans. Ind. Informat.*, vol. 6, no. 4, pp. 521–533, 2010.

[9] S. Martin and P. Minet, "Schedulability analysis of flows scheduled with FIFO: application to the expedited forwarding class," in *Proc. IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, Rhodes Island, Greece, Apr. 2006, pp. 8–pp.

[10] ——, "Worst case end-to-end response times of flows scheduled with FP/FIFO," in *Proc. IEEE Int. Conf. on Networking*, Mauritius, Apr. 2006, pp. 54–60.

[11] S. Martin, P. Minet, and L. George, "The Trajectory approach for the end-to-end response times with non-preemptive FP/EDF*," in *Software Engineering Research and Applications*. Springer, 2006, pp. 229–247.

[12] X. Li, J.-L. Scharbarg, and C. Fraboul, "Analysis of the pessimism of the Trajectory approach for upper bounding end-to-end delay of sporadic flows sharing a switched Ethernet network," in *Proc. 19th Int. Conf. on Real-Time and Network and Systems (RTNS)*, Nantes, France, Sep. 2011.

[13] H. Bauer, J.-L. Scharbarg, and C. Fraboul, "Applying and optimizing trajectory approach for performance evaluation of AFDX avionics network," in *Proc. IEEE Emerging Technologies and Factory Automation (ETFA)*, Marllorca, Spain, Sep. 2009, pp. 1–8.

[14] G. Kemayo, F. Ridouard, H. Bauer, and P. Richard, "Optimistic problem in the trajectory approach in FIFO context," in *Proc. IEEE Int. Conf. on Emerging Technologies and Factory Automation (ETFA)*. IEEE, Sep. 2013, pp. 1–8.