



**HAL**  
open science

## Frame Packing Strategy within Gateways for Multi-cluster Avionics Embedded Networks

Hamdi Ayed, Ahlem Mifdaoui, Christian Fraboul

► **To cite this version:**

Hamdi Ayed, Ahlem Mifdaoui, Christian Fraboul. Frame Packing Strategy within Gateways for Multi-cluster Avionics Embedded Networks. 17th IEEE International Conference on Emerging Technologies & Factory Automation (ETFA 2012), IEEE, Sep 2012, Cracovie, Poland. pp.1-8, 10.1109/ETFA.2012.6489577 . hal-04087232

**HAL Id: hal-04087232**

**<https://hal.science/hal-04087232>**

Submitted on 3 May 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



## Open Archive Toulouse Archive Ouverte (OATAO)

OATAO is an open access repository that collects the work of Toulouse researchers and makes it freely available over the web where possible.

This is an author-deposited version published in: <http://oatao.univ-toulouse.fr/>  
Eprints ID: 5978

**To cite this document:** Ayed, Hamdi and Mifdaoui, Ahlem and Fraboul, Christian *Frame Packing Strategy within Gateways for Multi-cluster Avionics Embedded Networks*. (2012) In: 17th IEEE International Conference on Emerging Technologies & Factory Automation (ETFFA ) 2012, 17-21 Sep 2012, Krakow, Poland.

Any correspondence concerning this service should be sent to the repository administrator: [staff-oatao@inp-toulouse.fr](mailto:staff-oatao@inp-toulouse.fr)

# Frame Packing Strategy within Gateways for Multi-cluster Avionics Embedded Networks

Hamdi AYED  
University of Toulouse  
ENSEEIH/IRIT  
hamdi.ayed@isae.fr

Ahlem MIFDAOUI  
University of Toulouse  
ISAE  
Ahlem.Mifdaoui@isae.fr

Christian FRABOUL  
University of Toulouse  
ENSEEIH/IRIT  
Christian.Fraboul@enseeiht.fr

## Abstract

*The current avionics communication architecture has recently arisen some limitations for meeting the emerging requirements of new generation aircraft in terms of modularity, weight and integration costs. In order to cope with these limitations, an alternative avionics communication architecture based on a multi-cluster embedded network is proposed in this paper. The critical avionics subsystems still are interconnected via the backbone network AFDX, however the sensors and actuators are grouped to be interconnected via CAN buses, independently of the associated avionics subsystems. The obtained clusters will be interconnected via specific gateways to handle the heterogeneity between the AFDX and CAN buses. This paper focuses first on the relevant aspects of our proposal and the offered real time guarantees required by critical avionics applications. Then, since an efficient bandwidth consumption is inherently important to guarantee an easy incremental design process, an optimized gateway design with a new frame packing strategy is proposed and evaluated in our context.*

## 1 Motivations and Related Work

The complexity of avionics communication architecture is increasing rapidly due to the growing number of interconnected subsystems and the expansion of exchanged data quantity. To follow this trend, the current architecture of new generation aircrafts like the A380, 400M or A350 consists of a high rate backbone network based on the AFDX (Avionics Full Duplex Switched Ethernet) [1] to interconnect the critical subsystems. Then, each specific avionics subsystem could be directly connected to its associated sensors/actuators network based on low rate data buses like ARINC429 [2] and CAN [7]. Although this architecture simplifies the design process and reduces the time to market, it leads in the same time to inherent weight and integration costs due the important number of sensors/actuators networks. In addition, this architecture makes the avionics subsystems closely dependent on their Inputs/Outputs and no longer interchangeable.

However, for avionics applications, it is essential that the communication architecture fulfills the emerging requirements in terms of modularity and performances to guarantee an easy incremental design process and the possibility of adding new functions during the aircraft lifetime.

Therefore, a new avionics communication architecture is proposed in this paper to guarantee these new generation avionics applications requirements. This proposal is based on a multi-cluster embedded network where the critical avionics subsystems still are interconnected via the backbone network AFDX. However, the sensors and actuators are grouped to be interconnected via CAN buses, independently of the associated avionics subsystems. The obtained clusters will be interconnected via specific gateways to handle the heterogeneity between the AFDX and CAN buses. This proposed architecture will clearly enhance the avionics subsystems modularity, however the gateways characteristics become one of the major challenges in the design process of such critical multi-cluster embedded network. This paper focuses first on the relevant aspects of our proposal and the offered real time guarantees required by critical avionics applications. Then, since an efficient bandwidth consumption is inherently important to guarantee an easy incremental design process, an optimized gateway design with a new frame packing strategy is proposed and evaluated in our context.

In the area of optimizing the bandwidth consumption over embedded networks, some approaches based on frame packing have been proposed. Frame packing approach ([8]) consists in building frames from many elementary data packets that respect the maximal frame's size, in order to reduce the overhead compared to including one data packet in each frame. For homogeneous embedded networks, [10] presented a specific frame packing strategy for AFDX applications implemented inside each AFDX EndSystem. Combined to an optimal routing algorithm through the network, this approach enhances the end to end communication delays and the bandwidth utilization rate on the AFDX. In [8] a similar approach was proposed for automotive applications and specifically for CAN equipments to enhance the good throughput on CAN. In the context of heterogeneous multi-cluster em-

bedded networks interconnected via gateways, [6] presented an approach for the design space exploration to find the optimal software/hardware mapping. This latter satisfies the different system constraints and integrates frame packing inside source nodes. However, the gateway was considered as a simple frames converter and the issue of optimizing this interconnection function was not tackled. In this specific topic, [9] deals with this problem for industrial applications to enhance the CAN scalability using Ethernet bridging. Frame packing was proposed within the bridges to reduce the induced overhead from CAN bus to Ethernet and average communication latencies was analyzed using simulation. This last approach presents a similarity with the one proposed in this paper, which consists on introducing a frame packing strategy within the interconnection function, but in the specific case of an heterogeneous avionics network AFDX-CAN. However, the proposed interconnection equipment is, not as in [9] a simple bridge which encapsulates CAN frames into AFDX frames, but a specific gateway on the application level which deals with the different heterogeneity aspects. Then, the proposed frame packing strategy is defined and optimized to satisfy in the worst case the different system constraints and to reduce the overhead and the bandwidth consumption.

Hence, our main contributions in this paper are three fold. First, a gateway design is proposed to handle the heterogeneity aspects between the two communication networks AFDX and CAN, and to minimize the induced CANs overhead and bandwidth through the AFDX network thanks to a new frame packing strategy, named *Fixed Waiting Time* strategy. Second, in order to deal with the worst case performance prediction of such proposal, schedulability analysis is conducted and optimization process of the frame packing strategy in case of one CAN bus and then many CAN buses is proposed. Third, general analysis are illustrated in the case of a realistic avionics network to highlight our proposal ability to fulfill the emerging avionics requirements.

The remaining of the paper is organized in the following way. In the next section, the current avionics architecture and the main used networks are described, then the proposed multi-cluster architecture based on specific gateways is presented. Afterward, the performance analysis of such proposal is tackled as follows. First, the gateway design issues are explained in section 3 and mainly the introduced frame packing strategy named *Fixed Waiting Time* strategy and the associated Virtual Links allocation. Then, the optimization process of the gateway strategy to select the most accurate waiting time that guarantees the minimum induced bandwidth on the AFDX is detailed in section 4. Finally, the practical feasibility of our proposal is illustrated within a realistic application in section 5. Section 6 concludes the paper.

## 2 Problem statement

In this section, we first present the current avionics architecture and its limitations in terms of modularity and reconfiguration. Then, we introduce and detail the main issues of an alternative architecture to cope with these limitations.

### 2.1 Current avionics architecture

As shown in figure 1, the current avionics communication architecture consists of a backbone network AFDX to interconnect the avionics endsystems, and some specific subsystems admit dedicated sensors/actuators network based on CAN or ARINC429 buses. This architecture could be no longer effective in meeting the emerging avionics requirements in terms of modularity and reconfigurability. In fact, the subsystems are no longer interchangeable because they completely depend on their dedicated sensors/actuators network. In addition, any introduced modification on this latter will require the reconfiguration of the associated subsystem.

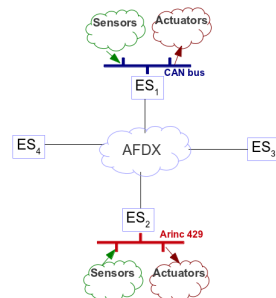


Figure 1. Current avionics architecture

**AFDX** The AFDX [1] network is based on Full Duplex Switched Ethernet protocol at 100Mbps, successfully integrated into new generation civil aircraft like the Airbus A380. This technology succeeds to support the important amount of exchanged data thanks to policing mechanisms added in switches and the Virtual Link (VL) concept. This latter gives a way to reserve a guaranteed bandwidth to each traffic flow. The VL represents a multicast communication which originates at a single End System and delivers packets to a fixed set of End Systems. Each VL is characterized by: (i) BAG (Bandwidth Allocation Gap), ranging in powers of 2 from 1 to 128 milliseconds, which represents the minimal inter-arrival time between two consecutive frames; (ii) MFS (Maximal frame size), ranging from 64 to 1518 bytes, which represents the size of the largest frame that can be sent during each BAG.

**CAN** The CAN bus [7] is a 1 Mbps data bus that operates according to an event-triggered paradigm where messages are transmitted using a priority-based access mechanism. There are two protocol formats: CAN 2.0 A (standard version) and CAN 2.0 B (extended version). The

former is considered herein for our case study. CAN bus works using a producer/consumer communication scheme based on unique identifier per message type. The CAN messages are broadcasted on the bus, then each CAN equipment will filter the consumed data based on the CAN identifier. The collisions on the bus are resolved following a CSMA/CR protocol (Carrier Sense Multiple Access/Collision Resolution) thanks to the bit arbitration method. The CAN frame consists of a payload up to 8 bytes and an overhead of 6 bytes due to the different headers and bit stuffing mechanism.

## 2.2 Proposed avionics architecture

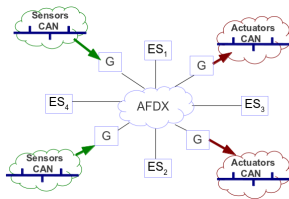


Figure 2. Proposed avionics architecture

In order to enhance the modularity of avionics network our proposal consists on keeping the AFDX network as a backbone network to interconnect the critical avionics systems, and to dissociate the sensors and actuators from their attached endsystems. As described in figure 2, we organize sensors and actuators in one or multiple peripheral networks based on a specific criterion, such as geographical proximity. In order to avoid the possible contentions between the flows coming from AFDX to CAN and from CAN to AFDX that can lead to performance degradation, the peripheral subnetworks are specific either for sensors or for actuators. The obtained clusters are interconnected via specific gateways functions. Compared to the current architecture, this alternative makes the endsystems interchangeable and enhances the reconfiguration process. In fact, in this case, the gateway becomes the only node that needs to be reconfigured in case of sensors or actuators modification. The work presented in this paper is mainly focused on the communications between the CAN sensors network and AFDX and excludes the communication between AFDX and CAN actuators.

The main heterogeneity parameters for the proposed avionics communication architecture concern the communication paradigms and the protocols characteristics dissimilarities between CAN and AFDX. The main arising issues to define and integrate these specific interconnection equipment are three fold.

**Gateway Design** The key function of this specific equipment is to keep the communication transparency between an AFDX calculator and a CAN sensor or actuator to avoid the alteration of existent hardware in these equipments. Hence, for an AFDX calculator the source or the destination of the transmitted Virtual Link is the interconnection equipment, while for a CAN sensor or actuator the

transmitted data is consumed or generated by the interconnection equipment. Consequently, the conversion of CAN frames on AFDX frames is exclusively performed in the interconnection equipment which guarantees the required communication transparency and spares the end to end communication semantics definition between AFDX and CAN nodes. Hence, an interconnection function on the application level seems the most suitable solution to handle the end to end communication semantics problem from CAN sensors to AFDX calculators and it is defined as a gateway. Two main parameters arisen during the design of these CAN to AFDX gateways.

- *Gateway Strategy and addressing scheme:* we need to define an accurate method to map CAN frames to AFDX VLs. A first basic strategy consists in associating for each CAN frame one AFDX VL thanks to a static mapping table. This latter method will be shown to be non optimal in terms of reserved bandwidth on AFDX.
- *Data formatting:* the gateway strategy implementation requires an appropriate data formatting in order to send one or many CAN frames on the same AFDX VL. The choice of this structure should take into account the AFDX standard and reduce as much as possible the induced overhead to guarantee the bandwidth consumption efficiency on the AFDX.

**Schedulability Analysis** For avionics embedded applications, it is essential that the communication network fulfills certification requirements, e.g. predictable behavior under hard real time constraints and temporal deadlines guarantees. The use of gateways may increase the communication latencies and real time constraints have to be verified. In order to deal with the worst case performance analysis of such network, an appropriate schedulability analysis has to be considered.

**Optimization Process** As it can be noticed, the sensors to calculators communication performances depend mainly on the gateway parameters. In order to increase the efficiency of bandwidth consumption on the AFDX and enhance margins for future avionics functions addition, an optimization process of the gateway strategy is required. This process will define the most accurate waiting time necessary for packing frames, which respects the different system constraints (hardware and temporal) and minimizes the induced overhead and bandwidth utilization rate.

## 3 Fixed Waiting Time strategy

In this section, the main gateway's design challenges are detailed. First, a new frame packing strategy is proposed to minimise the induced CAN bandwidth through the AFDX. Then schedulability analysis and VLs allocation process are presented.

### 3.1 Gateway strategy

A first basic strategy for the gateway, called (1:1) strategy, is possible where one CAN frame is converted to one AFDX frame. As described in figure 3, it proceeds as follows: first, each received CAN frame on the CAN interface is decapsulated to extract the payload. Then, thanks to the static mapping table, the associated Virtual Link is identified and the obtained AFDX frame is sent through the AFDX interface. The delay in the gateway in this case corresponds simply to a technological latency denoted by  $\epsilon$ . This strategy is simple to implement, but associating a dedicated AFDX frame of 64 bytes for each payload less or equal to 8 bytes implies a very bad bandwidth management on AFDX. For a CAN rate of 1 Mbps, we get almost 4Mbps on AFDX which is significant when considering several CAN buses. Thus, this solution is not optimized to have an efficient bandwidth consumption on the AFDX. This strategy will be considered through the paper as a reference point to conduct comparative analysis with our proposed strategy.

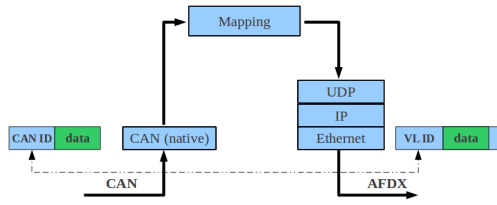


Figure 3. (1:1) gateway strategy

In order to reduce the induced bandwidth rate from the gateway to the AFDX, a new frame packing strategy is proposed, named Fixed Waiting Time (FWT) strategy. This strategy is based on a waiting timer, as shown in figure 4. This timer allows the cumulation of many CAN messages at the gateway's CAN interface. Then, when the timer expires, the gathered data will be sent in the same AFDX frame. Hence, the static mapping is no longer based on associating one Virtual Link for each CAN message but one Virtual Link to multiple CAN messages. This strategy will reduce the induced overhead which will be directly reflected on the induced bandwidth rate. The implementation of this new strategy arises some interesting questions about the AFDX frame structure and the addressing to reduce the overhead and to guarantee the deliverance correctness of messages to their destinations. Furthermore, this strategy requires an accurate VLs allocation scheme to minimize the reserved bandwidth on the AFDX backbone.

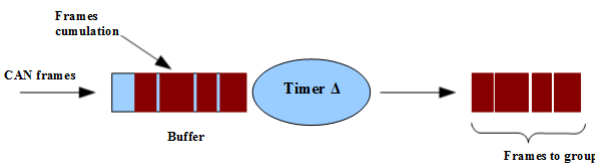


Figure 4. Fixed Waiting Time strategy

### 3.2 Data Formatting

In order to define the most accurate AFDX frame structure for each considered gateway strategy, we consider the concept of Functional Data set (FDS), introduced in the standard ARINC664 [1]. It allows grouping multiple data messages in the same AFDX frame and consists of two fields: (i) *Data Sets (DS)* that can contain several data primitives; (ii) *Functional Status Set (FSS)* which is a 4 Bytes field to encode the correctness and the status of at maximum 4 Data Sets. Further, the standard ARINC664 defines two types of frame structuring based on the FDS concept:

- *Implicit structure*: The frame has a fixed size and a predefined number of data fields with a specific size and order. Padding is used to keep the same structure of AFDX frame between successive transmissions.
- *Explicit structure*: The frame includes one label per data type for identification where the number and order of data messages can vary, with respect of the maximum frame size.

It is worth to note that the implicit structure reduces the AFDX overhead but in the same time it requires a good knowledge of data packets included in each AFDX frame. Therefore, this structure should be used under (1:1) strategy since there is exactly one CAN data in each AFDX frame. However, under the FWT strategy the choice is less obvious since we don't know exactly the packed data in each AFDX frame. To cope with this problem, we consider the worst case scenario of data packing during the introduced waiting time to define the maximum AFDX frame size. This latter will fit all possible scenarios of data packing. Therefore, the explicit structure will be more appropriate in this case to optimize the overhead and to support a variable number of packed data.

### 3.3 Schedulability analysis and VLs Allocation

In order to conduct the schedulability analysis for the proposed avionics communication architecture, the main metric that has been chosen is the worst case end to end delay that will be compared to the temporal deadline of each message. Let's consider  $\{SC_i\}_{i \in \{1..n\}}$  the set of CAN sensors buses and  $M_i = \{m_j^i\}_{j \in \{1..k\}}$  the set of messages sent on each bus  $SC_i$ . For each message  $m_j^i$ , we define four characteristics  $\{T_j^i, L_j^i, D_j^i, E_j^i\}$  which represent respectively the period, maximal length, deadline and the set of AFDX destinations. The end to end delay of a given message  $m_j^i$  sent from the CAN sensor  $SC_i$  to an AFDX calculator via the gateway can be defined as follows (figure 5) :

$$EED(m_j^i) = d_{CAN}(m_j^i) + d_{GTW}(m_j^i) + d_{AFDX}(m_j^i) \quad (1)$$

where,

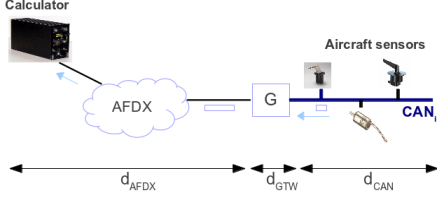


Figure 5. End to end delay definition

- $d_{CAN}(m_j^i)$  is a bound on the maximal response time of a CAN frame. In our case, the CAN bus is modeled as a rate monotonic scheduler and the tool *Cheddar* [4] is used to compute this bound.
- $d_{GTW}(m_j^i)$  is the duration the message might be delayed in the gateway and is equal to the payload extraction and mapping latency which depends on the gateway strategy.
- $d_{AFDX}(m_j^i)$  is a bound on the maximal delay submitted by the AFDX frame including  $m_j^i$ . Schedulability analysis for AFDX network, based on *Network Calculus* formalism, has been introduced in recent work [5]. The tool *WoPANets* [3] will be used throughout this paper to analyze this delay bound.

The schedulability condition of the considered set of messages can be written as follows:

$$\forall i, \forall j, EED(m_j^i) \leq D_j^i \quad (2)$$

As it can be noticed from (1), the end to end delays depend on the gateway strategy which will impact the gateway delay but also the AFDX delay. Actually, for the considered gateway strategy, the VLs allocated to send CAN data on the AFDX should be clearly defined. These output VLs sent from the gateway to the AFDX network will be in contention with the existent set of AFDX VLs. This contention could impact the messages' schedulability and has to be considered for the performance analysis. The VLs allocation induced by the gateway depends mainly on the waiting time inside the gateway and on the characteristics of CAN sensors messages.

For each bus  $SC_i$ , we assign a waiting time  $\Delta_i$  to its associated gateway; and we denote by  $V_i$  the set of VLs allocated to the output traffic from this gateway to the AFDX. The characteristics of  $V_i$  are defined as follows:

**BAG** Since each allocated VL to the gateway should be able to transmit the most urgent message coming from the associated CAN bus to the AFDX, we define the BAG as the closest value in power of 2 to the smallest period of the messages set  $M_i$ :

$$BAG_i = 2^{k_i}, \quad k_i = \left\lceil \frac{\log(\min_{m_j^i \in M_i} T_j^i)}{\log(2)} \right\rceil \quad (3)$$

**MFS** the largest AFDX frame generated by the gateway depends on the waiting time  $\Delta_i$ . According to the explicit

structure detailed in the previous section, a bound on the maximal frame size (in bytes) is as follows:

$$MFS_i = \max\left(84, \min\left(S_i, \Delta_i * \frac{10^6}{14}\right) + 67 + N_{lb}(\Delta_i)\right) \quad (4)$$

where,

- $S_i$  is the sum of payloads corresponding to the messages set  $M_i$ ;
- $\Delta_i * \frac{10^6}{14}$  is the maximal payload (in bytes) received from CAN to the gateway during  $\Delta_i$ ;
- $67 + N_{lb}(\Delta_i)$  is the induced overhead imposed by the AFDX explicit structure where  $N_{lb}(\Delta_i)$  is the maximal number of CAN messages received in the gateway during  $\Delta_i$ . Padding is used to guarantee a minimum AFDX frame size of 84 bytes (IFG (Inter Frame Gap) included).

**VLs number** in the worst case, the gateway will send as much VLs as the occurrences of the waiting time  $\Delta_i$  during the smallest period of the messages set  $M_i$ . Hence, the number of required VLs to handle CAN traffic sent from the gateway to the AFDX is:

$$N_{vls}(\Delta_i) = \left\lceil \frac{\min_{m_j^i \in M_i} T_j^i}{\Delta_i} \right\rceil \quad (5)$$

**Destinations** the set of AFDX destinations for each allocated VL is  $ES_i = \cup_{m_j^i \in M_i} E_j^i$ . Since any VL can include any type of CAN data, it could be then received by any AFDX endsystem that initially consumes at least one of these data.

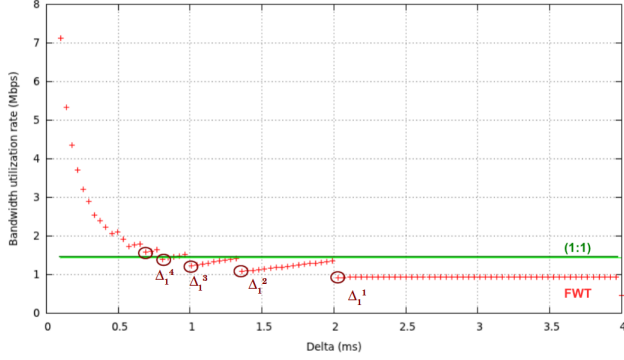
Giving the waiting time  $\Delta_i$  and its AFDX VLs characteristics, the gateway strategy is well defined and can be easily implemented. It is worth to note that the gateway's strategy mainly depends on its input CAN traffic. Therefore, any modification on this latter requires the re-configuration of the gateway by defining the appropriate waiting time and its allocated VLs.

## 4 Optimization Process

The aim of this part is to find the most accurate waiting time duration for the FWT strategy to enhance the efficiency of the bandwidth consumption on the AFDX. The problem is formalized as an optimization problem which takes into account the different system constraints; and has as objective to minimize the induced gateway bandwidth through the AFDX network. This problem is defined for one CAN bus and then generalized to the case of many CAN buses.

The optimization problem associated to the FWT strategy in case of one CAN bus is formulated as in (6). The objective function  $R(\Delta_i)$  to minimize is the induced gateway bandwidth through the AFDX backbone.

The first and second constraints correspond to the schedulability condition for the CAN messages set  $M_i$  and



**Figure 6.** Impact of Waiting Time in induced Gateway bandwidth through the AFDX

existent AFDX frames  $F_A$ , respectively. The third constraint represents the non overflow of the gateway's memory, where  $W_i$  is the associated maximum size. The delay inside the gateway is equal to the sum of the technological latency  $\epsilon_i$  and the waiting time for frame packing  $\Delta_i$ .

$$\begin{aligned}
 & \underset{\Delta_i}{\text{minimize}} && R(\Delta_i) \\
 & \text{subject to} && \forall m_j^i \in M_i, EED(m_j^i) \leq D_j^i \\
 & && \forall f_k \in F_A, d_{AFDX}(f_k) \leq \text{Deadline}_k \\
 & && \forall i, 10^6 * \Delta_i \leq W_i \\
 & && d_{GTW}(m_j^i) = \Delta_i + \epsilon_i \quad (6)
 \end{aligned}$$

The main difficult point to resolve this optimization problem lies on the fact that the AFDX delays could not be written as a closed form function of  $\Delta_i$ . Hence, to cope with this limitation, the impact of the waiting time in the induced gateway bandwidth through the AFDX has been considered. The main idea here is to choose the most accurate waiting time duration that gives the minimum induced bandwidth. Knowing the characteristics of the allocated VLs associated to each waiting time  $\Delta_i$ , the expression of  $R(\Delta_i)$  is as follows:

$$R(\Delta_i) = N_{vls}(\Delta_i) * \frac{MFS_i}{BAG_i} \quad (7)$$

Figure 6 shows an example of the function  $R(\Delta_i)$  for CAN sensors traffic described in table 2. The induced gateway bandwidth obtained with the basic gateway strategy, denoted by (1:1) on the figure 6, is considered as a reference to compare the two strategies performances. This latter corresponds to the sum of allocated VLs rates obtained when sending each CAN data in a separate VL and it is as follows:

$$R_{(1:1)} = \sum_{m_j^i \in M_i} \frac{84 * 8}{T_j^i} \quad (8)$$

As shown in figure 6, the FWT strategy offers a better bandwidth consumption compared to the basic strategy

(1:1) for a waiting time greater than 0, 8ms. Moreover, as it can be noticed, there are many local minima of the induced gateway bandwidth, corresponding to waiting time durations equal to integer divisors of  $BAG_i$ . The waiting time durations list corresponding to these minima is considered and sorted in a decreasing order as follows:

$$\{\Delta_i^1 = \frac{T_{min}^i}{2}, \Delta_i^2 = \frac{T_{min}^i}{3}, \dots, \Delta_i^{K_i} = \frac{T_{min}^i}{K_i + 1}\} \quad (9)$$

where  $K_i$  is the maximum integer giving a better bandwidth consumption than the strategy (1:1). In figure 6,  $K_i = 4$ .

According to the plotted curve, the induced bandwidth rate decreases when the waiting time increases. However, we could not simply consider the biggest waiting time duration because there is no guarantee on the schedulability and memory constraints feasibility. Hence, the idea is to explore this defined waiting time set and to verify for each iteration the temporal and memory constraints. If these constraints are verified, the exploration is stopped and the selected value is kept as the optimal value of the waiting time inside the gateway, which minimizes the induced bandwidth rate and respects all the system constraints.

The optimization for multiple CAN buses cannot simply be done by optimizing the gateway strategy for each CAN sensors bus. In fact, these buses are in direct contention on AFDX which leads to an interdependency between delay bounds of data sent from the gateways to the AFDX. Hence, an appropriate approach is introduced to find the optimal solution that minimizes the bandwidth consumption for the global network.

Let's consider  $(\Delta_1^{p_1}, \Delta_2^{p_2}, \dots, \Delta_n^{p_n})$  the configuration of a set of gateway connecting  $n$  CAN buses, where  $\Delta_i^{p_i}$  is the selected waiting time for CAN bus  $SC_i$  among the different possible values presented in the list (9). In order to find the best configuration, the following algorithm is introduced:

- (1) For each gateway associated to CAN sensors bus  $SC_i$ , compute the list of waiting times to explore  $\{\Delta_i^1 = \frac{T_{min}^i}{2}, \Delta_i^2 = \frac{T_{min}^i}{3}, \dots, \Delta_i^{K_i} = \frac{T_{min}^i}{K_i+1}\}$ .
- (2) Then, construct the list  $S = \{(\Delta_1^1, \Delta_2^1, \dots, \Delta_n^1), \dots, (\Delta_1^{K_1}, \Delta_2^{K_2}, \dots, \Delta_n^{K_n})\}$  of all possible configurations and sort them in the increasing order based on the induced bandwidth rate
- (3) Set the current configuration to the next element of the list  $S$ . For the first iteration, we take  $(\Delta_1^1, \Delta_2^1, \dots, \Delta_n^1)$
- (4) Check the schedulability of the current configuration:
  - a : Feasible: SUCCESS
  - b : Non feasible: go to step (3)



## 5 Performance Analysis

In this section, we present schedulability analyses for our case study under (1:1) and FWT gateway strategies. Then, we optimize the induced gateway bandwidth on the AFDX through different CAN loads scenarios.

### 5.1 Case study

Our case study is a representative avionics communication architecture where the traffic circulating on the Backbone network AFDX is described in table 1. As it can be noticed, there are 450 VLs with BAGs in  $\{4, 16, 32\}$ ms and MFSs in  $\{16, 226, 482\}$  bytes. This table represents the VLs distribution according to BAGs and frame lengths values. First, only one CAN sensors bus is integrated to show the impact of the FWT gateway strategy on the communication performances in terms of bandwidth and schedulability. Then, many CAN sensors buses are considered with different loads to show the efficiency of the introduced FWT strategy.

**Table 1. AFDX flows description**

BAG(ms)	Number of VLs	MFS (bytes)	Number of VLs
4	62	16	386
16	100	226	56
32	288	482	8

### 5.2 Schedulability analysis

A first example is considered where one CAN bus is interconnected to AFDX backbone via the specific gateway implementing the FWT strategy. This CAN bus supports the transmission of sensors messages described in table 2. There is 25 data messages to send from sensors to the gateway node, to be then transmitted to a predefined set of AFDX End Systems.

**Table 2. sensors traffic description**

Messages	Number	Payload(bytes)	Period(ms)
$m_1$	3	8	4
$m_2$	2	8	8
$m_3$	16	2	16
$m_4$	4	2	32

In table 3, are reported the induced VLs number and bandwidth rate under a FWT strategy with different waiting times varying from 1ms to 3ms. These values have been chosen less than the smallest period of the set of sensors data messages  $T_1 = 4ms$ , otherwise the configuration is obviously not feasible. The (1:1) strategy is used as a reference to show the possible performance enhancements with the proposed FWT strategy. As it can be noticed, the basic strategy leads to a significant number of VLs on the AFDX network with an inherent induced bandwidth rate. This is essentially due to the overhead when sending each sensor message (less than 8 bytes) in one AFDX frame ( at least 64 Bytes). On the other hand, the FWT strategy offers a noticeable amelioration on the number of allocated VLs in the gateway, but also on the

induced bandwidth rate on the AFDX where a reduction of 50% is obtained with  $\Delta_2$  and  $\Delta_3$ .

These results show a decreasing induced bandwidth rate when the waiting time is increasing. However, from a given value, in this case  $\Delta_3 = 2ms$ , there is a stability on the obtained rate even if the waiting time increases. It is worth to note that a non optimal choice of waiting time can lead to a poor bandwidth utilization rate as the case with  $\Delta_1 = 1ms$ . In fact, this value reduces the number of allocated VLs, but the induced rate still is high compared to the basic strategy. This is mainly due to the overdimensioning of the allocated BAG and MFS for each VL. For  $\Delta_2$  and  $\Delta_3$ , there is an important gain of the bandwidth consumption compared to the basic strategy and this is mainly due to the reduced number of allocated VLs.

**Table 3. Induced VLs characteristics**

Strategy	BAG (ms)	MFS (bytes)	VLs number	rate (Mbps)
(1:1)	4	84	25	1.4
$\Delta_1 = 1ms$	4	167	4	1.3
$\Delta_2 = 2ms$	4	172	2	0.69
$\Delta_3 = 3ms$	4	172	2	0.69

The end to end delay bounds are calculated for each message and then compared to their respective deadlines to analyze the schedulability condition. The obtained delay bounds under the (1:1) basic strategy and the FWT strategy with the considered waiting times are described in tables 4 and 5, respectively. The technical latency in the gateway node is assumed equal to  $\epsilon = 0.05ms$ .

**Table 4. Delay bounds under (1:1) strategy**

Msgs	$d_{AFDX}(ms)$	$d_{CAN}(ms)$	$T(ms)$	$d_{eed}^{1:1}(ms)$
$m_1$	1.12	0.54	4	1.71
$m_2$	1	0.81	8	1.86
$m_3$	1.1	1.95	16	3.1
$m_4$	1.4	2.23	32	3.68

**Table 5. Delay bounds under  $\Delta_1, \Delta_2$  and  $\Delta_3$**

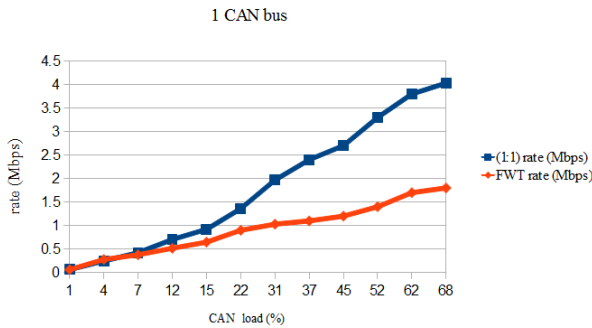
Msgs	$T(ms)$	$d_{eed}^{\Delta_1}(ms)$	$d_{eed}^{\Delta_2}(ms)$	$d_{eed}^{\Delta_3}(ms)$
$m_1$	4	3.3	3.8	5
$m_2$	8	2.8	3.7	4.7
$m_3$	16	4.5	5.1	6.3
$m_4$	32	5.1	5.9	7

As it can be noticed from table 4, the messages configuration is schedulable under the (1:1) strategy since all the messages delays respect their associated deadlines. Under FWT strategy, as reported in table 5,  $\Delta_1$  and  $\Delta_2$  lead to a schedulable configurations. However, when  $\Delta$  increases the schedulability condition is compromised as it can be seen with  $\Delta_3$ . Hence, increasing the waiting time inside the gateway is not always the best solution to enhance the performances with satisfying the temporal constraints.

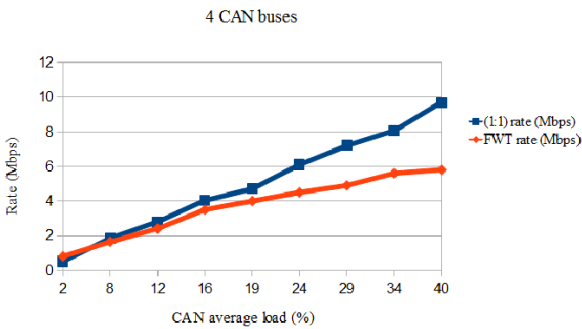
### 5.3 Optimization Results

We consider the general case of many CAN sensors buses connected to AFDX backbone. We keep the 450

AFDX flows of table 1 and we vary CAN buses loads. To simplify the analysis, for each scenario, sensors messages are generated with the following characteristics: periods randomly chosen in  $\{2^k, k \in \{2, 3, \dots, 7\}\}$ , sizes randomly chosen between 1 and 8 bytes and destinations number chosen randomly between 2 and 5. The selected scenarios respect the necessary condition of scheduling feasibility on CAN buses. We first consider the case of one CAN bus, then the case of 4 CAN buses in order to analyze the efficiency of FWT strategy under high loads. The optimal FWT waiting time is computed for each scenario using the algorithm introduced in section 4.



**Figure 7.** Impact of one CAN load in the induced bandwidth on the AFDX



**Figure 8.** Impact of 4 CAN loads in the induced bandwidth on the AFDX

Obtained results are presented in figures 7 and 8. As it can be noticed, FWT strategy leads to more efficient bandwidth consumption than the basic strategy under high loads, where a reduction of 55% is achieved for an average CAN load of 68%. This is mainly due to the overhead reduction thanks to this optimized strategy. However, under low loads, the over-dimensioning of allocated VLs under FWT strategy compared to (1:1) strategy can lead to the same bandwidth utilization. The same observation still is true with 4 CAN buses. However, the maximal CAN load leading to a feasible scenario is about 40% and this is mainly due to the important contention between different CAN buses.

Hence, the proposed FWT strategy within the gateway can inherently reduce the induced bandwidth on the AFDX compared to the (1:1) strategy, especially under high load scenarios.

## 6 Conclusion and Future Work

A multi-cluster avionics architecture is presented in this paper to enhance the modularity and the performances for new generation aircrafts. Since an efficient bandwidth consumption is inherently important to guarantee an easy incremental design process, an optimized gateway design with a new frame packing strategy, named Fixed Waiting Time strategy is proposed. The obtained results through our case study show the efficiency of this proposal to enhance the system's performances and to satisfy the different system constraints.

The next step will be to minimize the over-dimensioning of the VLs allocation induced by this approach to obtain further enhancements.

## References

- [1] Airlines Electronic Engineering Committee. Aircraft Data Network Part 7, AFDX NETWORK, ARINC Specification 664. Annapolis, Maryland, 2002. Aeronautical Radio.
- [2] Avionic Systems Standardisation Committee. Guide to avionics data buses. 1995.
- [3] H. Ayed and A. Mifdaoui. WOPANets: a tool for Worst Case Performance Analysis of Embedded Networks. IEEE International Workshop on Computer-Aided Modeling Analysis and Design of Communication Links and Networks (CAMAD), 2010.
- [4] Cheddar home page. <http://beru.univ-brest.fr/singhoff/cheddar/index-fr.html>.
- [5] J. Grieu. *Analyse et evaluation de techniques de commutation Ethernet pour l'interconnexion de systemes avioniques*. PhD thesis, INP, Toulouse, 2004.
- [6] P. Pop, P. Eles, and Z. Peng. Schedulability-driven frame packing for multicluster distributed embedded systems. *ACM Transactions on Embedded Computing Systems (TECS)*, 4, February 2005.
- [7] Robert Bosch GmbH. CAN specification Version 2.0. 1991.
- [8] R. Saket and N. Navet. Frame packing algorithms for automotive applications. *Journal of Embedded Computing*, 2:93–102, 2006.
- [9] J.-L. Scharbarg, M. Boyer, and C. Fraboul. CAN-Ethernet architectures for real-time applications. IEEE ETFA, 2005.
- [10] A. A. Sheikh, O. Brun, M. Chramy, and P. Hladik. Optimal Design of Virtual Links in AFDX Networks. 2012.