Priority Re-assignment for Improving Schedulability and Mixed-Criticality of ARINC 664

Metin Yeniaydın[1,2], Ömer Faruk Gemici[3], M. Selim Demir[3], İbrahim Hökelek[3,4], Sinem Coleri[5], and Ufuk Tureli[6]

[1]Department of Avionics Engineering, Yildiz Technical University, Istanbul, Turkey
[2]Turkish Aerospace Industries Inc., Istanbul, Turkey
[3]Research Center for Advanced Technologies on Informatics and Information Security, TUBITAK BILGEM, Kocaeli, Turkey
[4]Department of Telecommunication Engineering, Istanbul Technical University, Istanbul, Turkey
[5]Department of Electrical and Electronics Engineering, Koc University, Istanbul, Turkey
[6]Electronics and Communication Engineering Department, Yildiz Technical University, Istanbul, Turkey

***Abstract*—ARINC 664, which is a heavily used protocol for modern avionics networks, is preferred due to its simplicity although its mixed-criticality support is limited. Time Triggered Ethernet (TTEthernet), and IEEE Time Sensitive Networking (TSN), which utilize time synchronized schedule, are more suitable for supporting mixed-criticality applications; however, both require a fault tolerant time synchronization that makes the certification process more challenging. In this paper, we propose a novel dynamic priority assignment (DPA) concept together with the burst limiting shaper (BLS) from the IEEE TSN standard to enhance the schedulability and the mixed-criticality support of ARINC 664. The decision of flow re-assignment to a new priority class is done by calculating the high priority (HP) and low priority (LP) class worst-case delays using the network calculus framework. The numerical results show that the class utilization rates can be significantly increased by using the DPA concept with and without the BLS while the deadline constraints for all classes are satisfied. Thus, the DPA can improve the schedulability and mixed-criticality of ARINC 664 without using any time synchronization mechanism.**

***Index Terms*—ARINC 664, deterministic network, mixed-criticality, schedulability**

I. Introduction

Next generation aircrafts have enormous data exchange requirements among avionics applications. Since traditional avionics communication standards such as MIL-STD 1553 and ARINC 429 cannot satisfy these requirements, Ethernet based deterministic network technologies such as ARINC 664 [1], TTEthernet, and IEEE TSN have been proposed [2]. ARINC 664 is a profiled network, where the flows are shaped based on leaky bucket algorithm and policed with token bucket algorithm whose parameters are determined using offline network planning tool. TTEthernet and IEEE TSN implement time-triggered communication mechanisms that establish and maintain a global time through time synchronization, which helps to establish temporal partitioning and ensures isolation of the synchronous time-critical data flows from other traffic [3]. Time-triggered communication is highly suitable for periodic command and control tasks or synchronous data delivery with low latency and jitter requirements. TTEthernet has been specifically developed for the avionics systems while the IEEE TSN standards are for time-synchronized low latency streaming services [4]. In addition to its extensive usage in automotive and industrial control applications, there is a growing interest to utilize IEEE TSN standards for avionics networks [2]; however, to the best of our knowledge, there is no certifiable IEEE TSN solution up to this date for avionics applications. For both TTEthernet and IEEE TSN, the performance will be significantly degraded if time synchronization is broken due to failures [3].

The mixed-criticality is an important concept for the modern avionics systems such that distributed applications with mixed time-criticality requirements can be integrated and co-exist on a single physical network [5] [6]. TTEthernet and IEEE TSN are more suitable for the mixed-criticality since they utilize time synchronized schedule. However, both require implementing a complex fault tolerant time synchronization that makes the certification process more challenging. In [5], deficit-round robin, credit-based shaper, peristaltic shaper, and BLS are studied in detail from the mixed-criticality perspective. They demonstrate the advantage of using the BLS together with the strict priority scheduler in terms of complexity, modularity, fairness, and predictability. The traffic class that runs the BLS has two different priority levels and the priority level at a particular time is determined according to the consumed credit of the BLS. Although their mechanism decreases the worst-case delay of the LP class, the HP class may have higher worst-case delay than the LP class. Our priority assignment algorithm with and without using the BLS can guarantee that the HP class has lower or the same delay compared to the LP class and the bandwidth utilization rate that can meet the deadline constraints of both classes can be significantly increased.

In this paper, we propose a novel dynamic priority assignment (DPA) concept together with the burst limiting shaper (BLS) from the IEEE TSN standard to enhance the schedulability and the mixed-criticality support of ARINC 664. In the

ISBN 978-3-903176-39-3

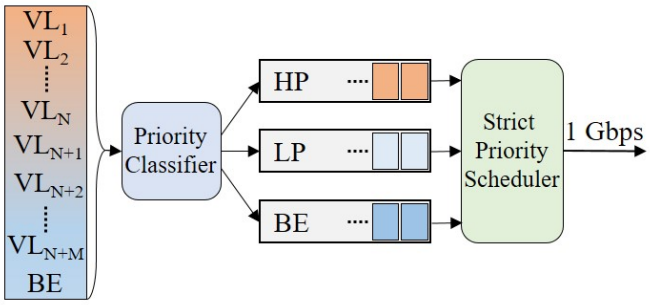

Fig. 1. System model

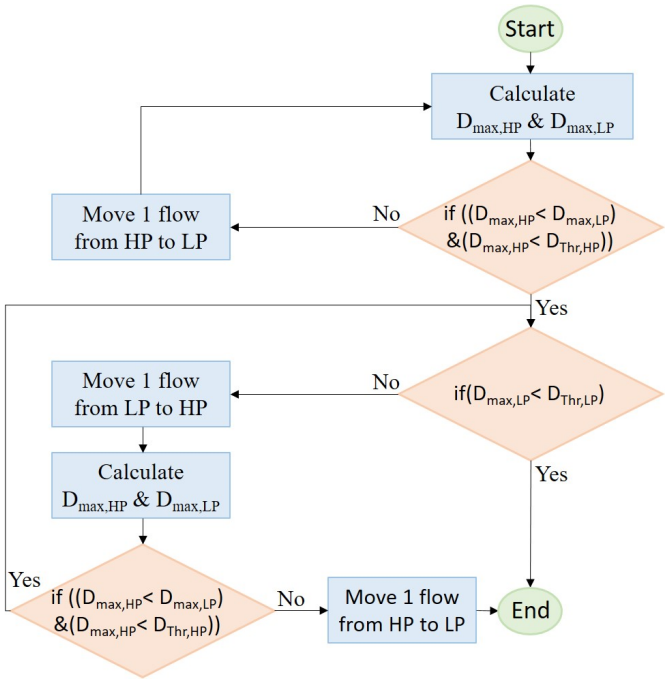

Fig. 2. Algorithm Flow Chart

proposed model, there are three different classes, namely HP for safety critical traffic (SCT), LP for rate constrained (RC) traffic, and best effort (BE) traffic. The class-based forwarding behavior at each node provides the latency and jitter bounds per class and these bounds can significantly change if the amount of traffic for each class is time-varying. This study utilizes the DPA concept to modify the initial assignment of the virtual link (VL) flows to the priority classes as the class utilization rates change. The priority re-assignment is performed carefully such that the deadline constraints of the HP and LP flows are simultaneously considered. The decision of flow re-assignment to a new priority class is done by calculating the HP and LP class worst-case delays using the network calculus framework. Numerical results show that the class utilization rates can be significantly increased by using the DPA concept with and without the BLS while the deadline constraints for all classes are satisfied. This indicates that the DPA can improve the schedulability and mixed-criticality of ARINC 664 without using any time synchronization mechanism.

The rest of the paper is organized as follows. The system model and the proposed algorithm are described in Section II and Section III, respectively. The delay analysis for both strict priority scheduler and BLS using network calculus is presented in Section IV. In Section V, numerical results are shown. Finally, we conclude the paper with Section VI.

II. SYSTEM MODEL

The proposed system model, where *N* HP flows, *M* LP flows, and BE traffic share the same output port of a node (i.e., ARINC 664 switch or end system), is depicted in Fig. 1. Here, each VL corresponds to an application flow in the ARINC 664 system.

For simplicity, there are three different priority classes in the system, including HP, LP, and BE. However, the proposed model can be extended if there are more than three classes. We assume that, within the same priority class, a flow with lower index number has a higher priority than another flow with higher index number. For example, VL_1 has a higher priority than VL_2, VL_2 has a higher priority than VL_3, and so on. In traditional ARINC 664 networks, the assignment of VL flows to priority classes are done at the offline network planning stage, and there is no concept of priority re-assignment of a VL flow to a different priority class. However, the bandwidth utilization rates of priority classes can vary significantly from time to time within the same flight duration due to the dynamic reconfiguration concepts in the integrated modular avionics (IMA) systems [7] [8]. This study suggests that the priority re-assignment can potentially prevent undesired delay performances especially for the lower priority flows. Note that the decision of flow re-assignment to a new priority class is done by calculating the HP and LP class worst-case delays using the network calculus framework as described in Section IV.

The system model can utilize the BLS together with the strict priority scheduler to shape the HP class. The priority level for the HP class at a particular time is determined by comparing the consumed credit with the maximum credit threshold (L_M) and the minimum credit threshold (L_R). The credit increases with a rate of I_{send} when the packets in the HP class are sent, otherwise it decreases with the rate of I_{idle}. If the credit reaches the L_M, the priority of the HP class becomes lower than the LP class. The priority returns back to the same level only if the credit equals to the minimum threshold L_R.

III. DYNAMIC PRIORITY ASSIGNMENT ALGORITHM

In this study, we propose a dynamic priority assignment concept such that the initial assignment of the flows to priority classes changes according to time varying class utilization rates. Note that this priority re-assignment is performed carefully such that the highest priority VL flow is moved from the LP to the HP class if the delay constraint of the HP class will not be violated. This is achieved by moving a flow with the

lowest index number in the LP class to the HP class since it has the highest priority. Hence, the flow that needs lower delay is assigned to the HP class assuming that the strict priority scheduler always provides the lowest delay for the HP class. Similarly, the lowest priority VL flow (highest index number) is moved from the HP to the LP class. Hence, the HP class delay can be decreased by sacrificing the delay performance of the lowest priority VL flow. This potentially provides better overall network performance since the sacrificed VL flow may still meet its deadline constraint in the LP class assuming that the priority levels are determined according to the delay constraints of the VL flows.

Fig. 2 shows the flowchart of the proposed algorithm to dynamically change the assignment of the VL flows to the priority classes if needed. Assuming that there are initially N HP and M LP flows, first of all, the maximum worst-case delays of HP and LP classes ($D_{max,HP}$, $D_{max,LP}$) are calculated using the network calculus framework. If $D_{max,HP}$ is higher than $D_{max,LP}$ or maximum delay threshold ($D_{max,Thr}$), the VL flows (starting from the lowest priority, the second lowest priority, and so on) are moved from the HP class to the LP class until the HP class delay constraints are satisfied (i.e., $D_{max,HP}$ is lower than $D_{max,LP}$ and $D_{max,Thr}$). At the second stage, if $D_{max,LP}$ is higher than $D_{max,Thr}$, then the algorithm moves a VL flow from the LP to the HP if the HP class delay constraints are not violated. If one of these constraints is violated, then the algorithm moves this flow back to the LP and terminates. Under the condition that $D_{max,LP}$ is higher than $D_{max,Thr}$, flows can be moved from the LP to the HP until $D_{max,LP}$ is lower than $D_{max,Thr}$ as long as the HP class delay constraints are not violated.

The method without using the algorithm described in the flow chart is called fixed priority assignment (FPA), where the strict priority scheduler with the HP and LP classes and the leaky bucket traffic shaper are used as described in the ARINC 664 standard. The method with the algorithm in the flow chart is called dynamic priority assignment (DPA), where the leaky bucket traffic shaper is used for VLs but the initial assignment of the VL flows to the priority classes can be changed if there is an assignment that can simultaneously satisfy the objectives of all classes. These two approaches can be applied when the BLS defined in the IEEE TSN standard is used for the HP class in addition to the leaky bucket shaper for VL flows. Then, these two methods (FPA and DPA) are named BLS-FPA and BLS-DPA, respectively. The following section presents how the network calculus can be applied to calculate the worst-case delay bounds for these four methods.

IV. NETWORK CALCULUS DELAY ANALYSIS

Network calculus that lies in a theoretical background of Min-plus algebra allows us to calculate the upper bound delay [9]. Therefore, it is widely used to analyze the worst-case delay bounds for deterministic networks including ARINC 664 or IEEE TSN [10] [11]. In the following two subsections, we present the worst-case delay analysis for the FPA, DPA, BLS-FPA, and BLS-DPA methods.

A. Network Calculus for FPA and DPA

We assume that there is one-to-one mapping from virtual link n (VL$_n$) to flow n (f_n) which can be defined by two parameters, namely bandwidth allocation gap (BAG_n) and maximum frame size (s_n). The arrival curve of f_n, which is regulated by the leaky bucket algorithm, can be formulated as $\alpha_n(t) = r_n t + s_n$, where $r_n = s_n/BAG_n$ represents the arrival rate. The service curve of an output port for the ARINC 664 switch can be expressed as $\beta(t) = R[t-T]^+$, where R is the service capacity and T is the technological latency. Sharing the same output port reveals the concept of residual service curve, which allows us to find the guaranteed individual remaining service capacity for each flow [9]. The residual service curve needs to be calculated based on the service type of the service element. In this study, we utilize first in first out (FIFO) service type for the flows having the same priority level, whereas the strict service type assumption is utilized for the flows having different priority levels. The preemption mechanism is not considered in this study. Let $p(i)$ represents the priority level of flow i, and $p(i) < p(j)$ denotes that flow i has higher priority than flow j. The residual strict service curve offered by strict priority to flow f_j having the priority level $p(j) \in \{HP, LP, BE\}$ can be expressed as [12]:

$$\beta_j^{sp}(t) = \left[\beta(t) - \sum_{\forall i, p(i)<p(j)} \alpha_i(t) - \max_{\forall i, p(i)>p(j)} s_i\right]^+ . \quad (1)$$

The worst-case delay bound of f_j can be calculated by obtaining the maximum horizontal distance between the arrival and service curves as [13]:

$$D_j^{sp}(t) = \left(\frac{\sum_{\forall i, p(i)=p(j)} s_i}{R - \sum_{\forall i, p(i)<p(j)} r_i}\right) + \left(\frac{\sum_{\forall i, p(i)<p(j)} s_i + \max_{\forall i, p(i)>p(j)} s_i}{R - \sum_{\forall i, p(i)<p(j)} r_i}\right). \quad (2)$$

Note that the above equations can be directly applicable for both FPA and DPA methods since the flow re-assignment to the priority classes affects only the number of flows for HP and LP classes in these equations.

B. Network Calculus for BLS-FPA and BLS-DPA

In ARINC 664, the worst-case delay bounds for LP flows can significantly increase as the bandwidth utilization rate of HP flows increases due to the strict priority scheduler at the switch output port. The BLS concept is applied for ARINC 664 to enhance the worst-case delay bounds by dynamically changing the priority level of HP class [12]. In other words, the priority level of the HP class can be lower than the LP class for a certain period of time. Window-based approach (WbA) is utilized to determine the priority level of the HP class with respect to the LP class. In this study, in addition to applying the BLS for the HP class, we also utilize the DPA concept presented in Section III.

For the proposed system model in Fig. 1, the minimum service curve for the HP class, where the BLS is utilized, can be expressed as [12]:

$$\beta_{HP}^{bls}(t) = \left(\frac{\Delta_{send}^{min}}{\Delta_{send}^{min} + \Delta_{idle}^{max}} \right) \cdot R \cdot (t - \Delta_{idle}^{max})^{+} \ , \quad (3)$$

where $\Delta_{send}^{min} = \frac{L_M - L_R}{I_{send}}$ represents the minimum duration of the HP class when it has the highest priority while $\Delta_{idle}^{max} = \frac{L_M - L_R}{I_{idle}} + \frac{s_{LP}}{R}$ represents the maximum duration for the priority level of the HP class when its priority is lower than the LP class. Here, L_M, L_R, I_{send}, and I_{idle} represent the upper credit threshold, the lower credit threshold, the increase rate of the credit, and the decrease rate of the credit for the BLS algorithm, respectively. Note that the BE traffic has always the lowest priority.

Since the service curve of the HP class can take two different values depending on the priority level of the HP class, namely the non-preemptive strict priority service curve and the BLS service curve, the multiplexed service curve can be calculated as:

$$\beta_{HP}^{mux}(t) = max\Big(\beta_{HP,2}^{sp}, \beta_{HP}^{bls} \otimes \beta_{HP,0}^{sp} \Big)(t) \ , \quad (4)$$

where $\beta_{HP,2}^{sp}(t)$ and $\beta_{HP,0}^{sp}(t)$ represent the service curves of the HP class when it has lower and higher priority than the LP class, respectively, while β_{HP}^{bls} is the service curve of the BLS. Note that $\otimes$ denotes the min-plus convolution operator and β_{HP}^{bls} is given in Eq. (3). $\beta_{HP,2}^{sp}(t)$ and $\beta_{HP,0}^{sp}(t)$ can be expressed as follows:

$$\beta_{HP,2}^{sp}(t) = \big(R \cdot t - \alpha_{LP}(t) - max_{i \in BE} s_i \big)^{+} \ , \quad (5)$$

$$\beta_{HP,0}^{sp}(t) = \big(R \cdot t - max_{i \in (LP,BE)} s_i \big)^{+} \ . \quad (6)$$

The service curve $\beta_{HP,2}^{sp}(t)$ can be expressed as $R_{HP,2}^{sp} \left(t - T_{HP}^{sp} \right)^{+}$, where $R_{HP,2}^{sp}$ represents the residual service capacity while T_{HP}^{sp} includes the technological latency for the case that the HP class has lower priority than the LP class. Similarly, $\beta_{conv}(t)$ can be expressed as $R_{conv} \left(t - T_{conv} \right)^{+}$ where R_{conv} denote the residual service capacity while T_{conv} includes the technological latency for the service curve of $\beta_{HP}^{bls} \otimes \beta_{HP,0}^{sp}$. Therefore, the worst-case delay bounds for the HP class can be expressed as:

$$D_{HP}^{mux} = min(D_{HP,2}^{sp}, D_{conv}) \ . \quad (7)$$

Substituting $D_{conv} = \sum_{i \in HP} \frac{s_i}{R_{conv}} + T_{conv}$ and $D_{HP,2}^{sp} = \sum_{i \in HP} \frac{s_i}{R_{HP,2}^{sp}} + T_{HP,2}^{sp}$ into Eq. (7), the worst-case delay bound for the HP class can be calculated.

The service curve of the LP class is also affected when the BLS is applied to the HP class. The multiplexed minimum service curve offered to the LP class can be expressed as:

$$\beta_{LP}^{mux}(t) = max\Big(\beta_{LP}^{sp}, \beta_{LP}^{bls} \Big)(t) \ , \quad (8)$$

where $\beta_{LP}^{sp}(t)$ and $\beta_{LP}^{bls}(t)$ represent the minimum service curve offered to the LP class when it has lower priority than the HP class and when the BLS is applied to the HP class, respectively. These service curves can be expressed as:

$$\beta_{LP}^{sp}(t) = \big(R \cdot t - \alpha_{HP}(t) \oslash \beta_{HP}^{bls}(t) - max_{i \in (HP,LP,BE)} s_i \big)^{+} \ , \quad (9)$$

$$\beta_{LP}^{bls}(t) = \big(R \cdot t - \Upsilon_{HP}^{bls} - max_{i \in (HP,LP,BE)} s_i \big)^{+} \ , \quad (10)$$

where $\oslash$ denotes the min-plus deconvolution operator and Υ_{HP}^{bls} represents the maximum service curve offered by the BLS. It becomes $\Upsilon_{HP}^{bls} = R \cdot t$ when there is no backlogged LP traffic, otherwise:

$$\Upsilon_{HP}^{bls} = \frac{R}{\Delta_{\Upsilon_{SCT}}} \cdot \Delta_{send}^{max} \cdot t + \frac{R}{\Delta_{\Upsilon_{SCT}}} \cdot \Delta_{send,0}^{max} \cdot \Delta_{idle}^{min} \ , \quad (11)$$

where $\Delta_{send}^{max} = \frac{L_M - L_R}{I_{send}} + \frac{s_{HP}}{R} + min(\frac{s_{LP}}{R} \cdot \frac{I_{idle}}{I_{send}}, \frac{L_R}{I_{send}})$ represents the maximum duration of the HP class having the highest priority. The initial value for Δ_{send}^{max} is represented as $\Delta_{send,0}^{max} = \frac{L_M}{I_{send}} + \frac{s_{HP}}{R}$. In addition, $\Delta_{idle}^{min} = \frac{L_M - L_R}{I_{idle}}$ represents the minimum duration for the HP class when its priority level is lower than the LP class. Here, $\Delta_{\Upsilon_{SCT}} = \Delta_{send}^{max} + \Delta_{idle}^{min}$ denotes the period of the maximum service curve analysis.

The service curve $\beta_{LP}^{\mu}(t)$ can be expressed as $R_{LP}^{\mu} \left(t - T_{LP}^{\mu} \right)$, where $\mu \in \{sp, bls\}$. R_{LP}^{μ} represents the residual service capacity, while T_{LP}^{μ} includes the technological latency for the case that the LP class has lower priority than the HP class. Therefore, the worst-case delay can be expressed as:

$$D_{LP}^{mux} = min(D_{LP}^{sp}, D_{LP}^{bls}) \ . \quad (12)$$

Substituting $D_{LP}^{\mu} = \sum_{i \in LP} \frac{s_i}{R_{LP}^{\mu}} + T_{LP}^{\mu}$ into Eq. (12), the worst-case delay bound for the LP class can be calculated.

Note that the above equations can be directly applicable for both BLS-FPA and BLS-DPA methods since the flow re-assignment to the priority classes affects only the number of flows for HP and LP classes in these equations.

V. NUMERICAL EXPERIMENTS

In this section, we provide numerical experiments to demonstrate the delay performance of four methods described in Section III, and IV, namely FPA, DPA, BLS-FPA, and BLS-DPA. Note that our proposed dynamic priority assignment based methods are DPA and BLS-DPA, while the fixed priority assignment based methods FPA and BLS-FPA are used as the benchmark.

For all experiments, one node with three priority classes including HP, LP, and BE is considered, as shown in Fig. 1. The same experiment parameters in [10] are used in this study. The maximum frame size (MFS) and the bandwidth allocation gap (BAG) are set to 64 bytes and 2 ms for SCT and RC flows, respectively, while MFS for BE is 1024 bytes and BAG is 8 ms. In the first set of the experiments, the utilization rate of the HP class (UR_{HP}) is increased from 0.1% to 78% while the utilization rate of the LP class (UR_{LP}) is kept constant as 20%. In the second set of the experiments, UR_{LP} is varied from 0.5% to 72% while UR_{HP} is kept constant

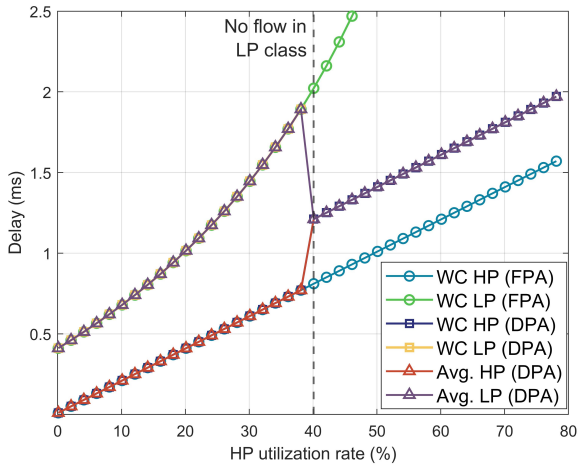

Fig. 3. The delay performance of FPA and DPA as UR_{HP} varies

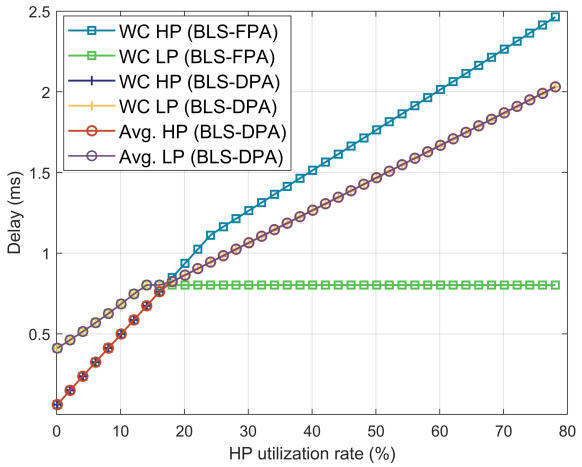

Fig. 5. The delay performance of BLS-FPA and BLS-DPA as UR_{HP} varies

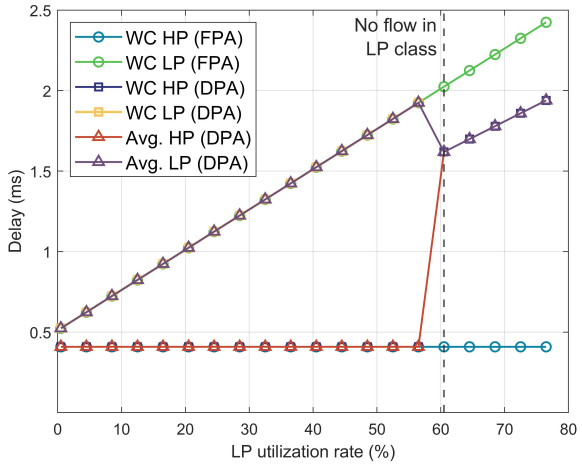

Fig. 4. The delay performance of FPA and DPA as UR_{LP} varies

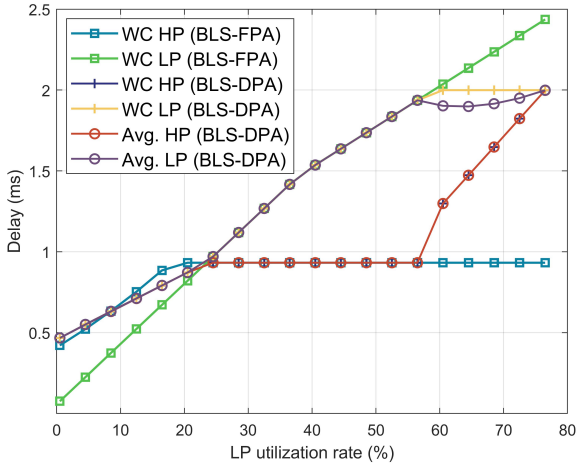

Fig. 6. The delay performance of BLS-FPA and BLS-DPA as UR_{LP} varies

as 20%. Note that the utilization rate is increased by adding new flows to the corresponding class and the BE is used to load the system up to the 100% utilization rate. The delay analysis of the BE is not presented since there is no deadline constraint. Without loss of generality, the delay requirements for both HP and LP traffic classes are set to 2 ms. The L_M, L_R, and BW parameters required for the BLS-FPA and BLS-DPA experiments are set to 22118, 0 and 0.46, respectively, as in [10]. The BW parameter of 0.46 means that the bandwidth of the HP class is set to the 46% of the total outgoing capacity. We report the worst-case (WC) and average (Avg.) delays for all four methods. Since there is no priority re-assignment for FPA and BLS-FPA, the WC and Avg. delay results will be the same. Therefore, only the WC delays are shown for FPA and BLS-FPA in the figures. To quantify the improvement due to the priority re-assignment, the average of the worst-case delays for the flows, that are initially at the same class, is calculated after the priority re-assignment.

Fig. 3 illustrates the effect of the HP utilization rate on the delay results for the FPA and DPA methods. The same delay results are observed for both methods until UR_{HP} reaches 38.1%. Beyond this point, the delay requirements of the LP class cannot be satisfied using the FPA while the worst-case and average delays of the LP class significantly decrease by using the DPA method thanks to the priority re-assignment of the initially LP flows to the HP class. Note that all LP flows have to be moved to the HP class for this particular experiment, otherwise some of the LP flows cannot satisfy the LP delay constraint since all LP flows have the same delay constraint of 2 ms. Therefore, the worst-case and average delays for both HP and LP flows are the same when UR_{HP} is beyond 38.1%. As a result, the DPA method can satisfy the delay constraints of HP and LP flows for all UR_{HP} values while the FPA method cannot satisfy the delay constraints of LP flows when UR_{HP} is beyond 38.1%. This improvement for the LP flows is achieved by increasing the delay results of the HP class in an acceptable

manner since the delay requirements of the HP class are still met.

Similarly, Fig. 4 demonstrates how the change in UR_{LP} affects the worst-case and average delay results of the FPA and DPA methods. Both the FPA and DPA methods satisfy the delay requirements for HP and LP flows when UR_{LP} is lower than 56.5%. Beyond this point, the DPA method performs the priority re-assignment of initially LP flows to the HP class to meet the delay requirement for the LP class. Although this priority re-assignment increases the delay of the HP class, it does not pose a problem since the delay requirements are met for both HP and LP flows. Thanks to the DPA method, a significant improvement in terms of supporting higher LP utilization rates is achieved by fulfilling the delay requirements that cannot be satisfied by the FPA method.

Fig. 5 shows the impact of UR_{HP} on the delay results when the BLS-FPA and BLS-DPA methods are utilized. Until UR_{HP} reaches 18.1%, the delay results of the HP class are the same for both BLS-FPA and BLS-DPA and the delay results of the LP class are the same for both methods as well since there is no need for dynamic priority assignment. However, after this critical point, BLS-FPA exhibits an undesirable behavior such that the worst-case delay is higher for the HP compared to the LP. BLS-DPA avoids this problem by priority re-assignment in such a way that some of HP flows (starting from the lowest priority one) are moved to the LP class until the worst-case delay for the HP class is lower than the LP class. Therefore, the worst-case delays of both classes can be seen at midpoint in Fig. 5. The results show that the average delays of HP and LP are also close to each other when UR_{HP} is beyond 18.1%. BLS-DPA outperforms BLS-FPA in terms of schedulability since the maximum UR_{HP} values, for which the deadline requirements of both HP and LP classes are met, are 60.1% and 78.1% for BLS-FPA and BLS-DPA, respectively.

The impact of UR_{LP} on the delay results for both BLS-FPA and BLS-DPA is demonstrated in Fig. 6. For low UR_{LP} values, BLS-FPA suffers from the above mentioned undesired problem that the worst-case delay for the HP class is higher than the LP class. However, BLS-DPA resolves this issue by re-assigning the HP flows into the LP class. Both methods provide the same results when UR_{LP} is between 24.5% and 56.5%. After this critical point, the worst-case delay for the LP class exceeds the deadline constraint for BLS-FPA. In contrast, BLS-DPA moves some of the HP flows into the LP class until the delay constraint is satisfied. It is crucial to note that the average delay for the LP class is below the worst-case delay, which means that the priority level is changed only for a few flows. Besides, BLS-FPA schedulability expires when UR_{LP} is equal to 56.5%. On the other hand, there is no restriction for BLS-DPA in terms of schedulability concern.

Finally, it is worth mentioning that DPA yields lower delay results for the HP class compared to BLS-DPA until UR_{LP} reaches 60.1%. However, as UR_{LP} increases beyond this point, BLS-DPA provides the delay results that are more convenient for the HP class. Similarly, as UR_{HP} increases, DPA outperforms BLS-DPA when UR_{HP} is lower than 40.1% while the difference is only a few microseconds for higher UR_{HP} values.

VI. CONCLUSION

This paper presented a DPA concept to decide the mapping of ARINC 664 VL flows to the priority classes according to time-varying bandwidth utilization rates. The DPA concept is applied with and without the IEEE TSN BLS by calculating the HP and LP class worst-case delays using the network calculus framework. The numerical results show that the DPA method with the IEEE TSN BLS maximizes the bandwidth utilization rate for the HP class first and the remaining capacity is used to meet the deadline constraints for the LP class. This indicates that the DPA with the IEEE TSN BLS can improve the schedulability and mixed-criticality of ARINC 664 without using any complex time synchronization mechanism. Future work will investigate a dynamic learning based flow re-assignment algorithm and more realistic multi-node scenario for avionics networks by specifying different delay constraints for VLs.

REFERENCES

[1] *ARINC 664, P7: Avionics Full Duplex Switched Ethernet (AFDX) Network*. INC Aronautical Radio, 2005.

[2] L. Zhao, F. He, E. Li, and J. Lu, "Comparison of Time Sensitive Networking (TSN) and TTEthernet," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*, 2018, pp. 1–7.

[3] E. Kyriakakis, M. Lund, L. Pezzarossa, J. Sparsø, and M. Schoeberl, "A time-predictable open-source TTEthernet end-system," *Journal of Systems Architecture*, vol. 108, p. 101744, 2020.

[4] L. L. Bello and W. Steiner, "A perspective on IEEE time-sensitive networking for industrial communication and automation systems," *Proceedings of the IEEE*, vol. 107, no. 6, pp. 1094–1120, 2019.

[5] A. Finzi, A. Mifdaoui, F. Frances, and E. Lochin, "Incorporating TSN/BLS in AFDX for mixed-criticality applications: Model and timing analysis," in *2018 14th IEEE International Workshop on Factory Communication Systems (WFCS)*. IEEE, 2018, pp. 1–10.

[6] J. Yao, J. Wu, Q. Liu, Z. Xiong, and G. Zhu, "System-level scheduling of mixed-criticality traffics in avionics networks," *IEEE Access*, vol. 4, pp. 5880–5888, 2016.

[7] M. N. Durceylan, Ö. F. Gemici, İ. Hökelek, and H. Aşmer, "Reconfigurable ARINC 664 end system implementation on FPGA," in *2020 28th Signal Processing and Communications Applications Conference (SIU)*, 2020, pp. 1–4.

[8] S. Cevher, A. Mumcu, A. Caglan, E. Kurt, M. K. Peker, I. Hokelek, and S. Altun, "A fault tolerant software defined networking architecture for integrated modular avionics," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*, 2018, pp. 1–9.

[9] A. Bouillard, M. Boyer, and E. Le Corronc, *Deterministic Network Calculus: From Theory to Practical Implementation*. John Wiley & Sons, 2018.

[10] A. Finzi and A. Mifdaoui, "Worst-case timing analysis of AFDX networks with multiple TSN/BLS shapers," *IEEE Access*, vol. 8, pp. 106 765–106 784, 2020.

[11] F. He, L. Zhao, and E. Li, "Impact analysis of flow shaping in Ethernet-AVB/TSN and AFDX from network calculus and simulation perspective," *Sensors*, vol. 17, no. 5, p. 1181, 2017.

[12] A. Finzi, "Specification and analysis of an extended AFDX with TSN/BLS shapers for mixed-criticality avionics applications," Ph.D. dissertation, Toulouse, ISAE, 2018.

[13] H. Bauer, J.-L. Scharbarg, and C. Fraboul, "Improving the worst-case delay analysis of an AFDX network using an optimized trajectory approach," *IEEE Transactions on Industrial informatics*, vol. 6, no. 4, pp. 521–533, 2010.