Towards a Time Redundancy Mechanism for Critical Frames in Time-Sensitive Networking

Inés Álvarez*, Julián Proenza†, Manuel Barranco†, Mladen Knezic*
*Faculty of Electrical Engineering, University of Banja Luka, Bosnia and Herzegovina,
ines.alvarez@uib.es, mladen.knezic@etf.unibl.org
†Departament de Matemàtiques i Informàtica, Universitat de les Illes Balears, Spain,
julian.proenza@uib.es, manuel.barranco@uib.es

***Abstract*—Time-Sensitive Networking (TSN) is a set of technical standards that is being developed to provide Ethernet with hard real-time, reliability and flexibility services. In the last years, there has been a growing interest in increasing the connectivity of all kind of devices. This trend has reached industrial environments, where the demanding timing and reliability constraints imposed the use of specialised networks with specific features to support these requirements. Moreover, the industry has shown interest in using Ethernet as the network technology in industrial environments, due to its low cost, high bandwidth and extensive use. The ability of TSN to support both, data-oriented and traditional control traffic over the same network makes it an appealing technology to implement the next generation of industrial networks with high connectivity. Nevertheless, TSN does not cover some reliability aspects important for its deployment in critical systems. In this work we propose the implementation of time redundancy of frames in order to tolerate temporary faults in the channel and, therefore, increase the reliability of the network.**

I. Introduction

Time-Sensitive Networking (TSN) is a set of technical standards that is being developed by the Institute of Electrical and Electronics Engineers (IEEE). These standards aim at providing Ethernet with hard real-rime, reliability and flexibility services. For a long time, Ethernet was considered by the industry as an appealing technology to substitute field buses [1] for its low cost high bandwidth, spread use and Internet compatibility. Nevertheless, Ethernet lacked support for the demanding timing and reliability needs of traditional control networks. Even though there have been previous efforts to provide Ethernet with these services, such as Time-Triggered Ethernet [2] or Flexible Time-Triggered Ethernet [3], there was not a standard solution for the arising needs in the industry.

Moreover, in the last years there has been a growing interest in increasing the connectivity of all types of devices. This trend has showed up in industry in the form of Industrial Internet of Things (IIoT) and Industry 4.0. TSN represents a promising technology for the integration of data-oriented and traditional control communications over the same network since it allows to adapt the Quality of Service (QoS) provided by the network to the type of traffic that needs to be transmitted.

Even though the Time-Sensitive Networking task group is currently working in increasing the reliability level of Ethernet, the available information at the time of writing this paper shows that some important reliability aspects to meet the demanding needs of critical systems are not to be covered by the standards. More precisely, TSN provides mechanisms for spatial redundancy of the channel and the transmission of replicated frames through multiple channels to tolerate permanent and temporary faults of the channel, a.k.a transient faults.

Nevertheless, the communication channel is especially vulnerable to transient faults due to electromagnetic interference. Therefore, if spatial redundancy has been designed to provide a certain level of reliability against permanent faults, using it to tolerate transient faults will decrease the capability of the network to tolerate permanent faults, particularly when they are coincident with transient ones. Additionally, once the spatial redundancy is not available due to the permanent failure of some of the channels, the system is not able to tolerate transient faults any more.

Therefore, in this work we propose the usage of time redundancy of streams to tolerate transient faults of the channel. This mechanism consists in sending multiple copies of each frame over the same link. We provide an overview of the advantages and disadvantages of different types of replication schemes, different approaches to select which frames should be replicated, what elements of the network should carry out the replication, how to decide the number of copies to be transmitted and finally, what are the different steps needed to effectively replicate the frames and identify the replicas.

The remainder of the document is organised as follows. In section II we introduce the TSN concepts related to our proposal, paying special attention to the reliability aspects of the standards. In section III we describe different design options. Finally, in section IV we conclude the work.

II. TSN basic concepts

As mentioned before, TSN aims at providing hard real-time, flexibility and reliability to Ethernet-based networks to enable their use in critical applications [4]. To this, the TSN standardisation group is working in ten different standards and amendments. To provide timing guarantees and increase flexibility TSN relies, among others, on the Stream Reservation Protocol (SRP), originally standardised in IEEE 802.1Qat-2010 [5], and currently under revision.

SRP enables the reservation of resources along the path between two nodes that want to communicate to guarantee availability and bounded transmission times. The communication is done through virtual communication channels called streams to which nodes attach as *talker* (transmitter) or *listeners* (receivers). The resource reservation is done in a per-stream manner and determines the QoS of all messages transmitted through that

©2017 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

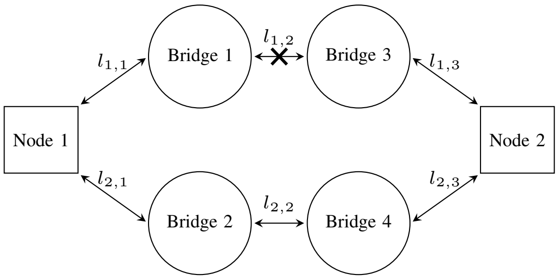

Fig. 1: Network with two disjunctive paths between Node 1 and Node 2, and one permanent error in link $l_{1,2}$.

stream. Since SRP is in charge of creating, managing and deleting streams it will have to be taken into account by our design. SRP considers different classes of traffic with different QoS guarantees, namely hard real-time *critical* traffic and soft real-time *class A* and *class B* traffic. Furthermore, SRP considers an additional class of traffic, *alarm* traffic, which corresponds to critical traffic that has higher priority than any other, so resources will be assigned to this type of traffic even if other streams must be removed from the network.

One of the main objectives of TSN is to increase the reliability of the communications based on Ethernet by means of spatial and information redundancy. To that, two new standards were proposed, IEEE 802.1Qca which allows for the creation and management of multiple paths between any pair of nodes in the network and IEEE 802.1CB which manages the creation and elimination of frame replicas to be transmitted through the existing multiple paths.

IEEE 802.Qca describes new services, to allow for the creation of multiple and non-shortest paths between nodes and the further reservation of resources through those paths. On the other hand, IEEE 802.1CB manages the replication of streams so one frame will be transmitted through each one of the multiple paths. It defines how to identify streams that must be replicated, how frames should be replicated at transmission and identified at reception. Every element in the replicated paths, bridges and nodes, implement the replication and elimination mechanisms.

Nevertheless, as has been mentioned before, TSN does not include time redundancy of critical frames, since this issue is considered to be out of the scope of the standards. In the next section we will describe the possible negative effects derived from not having time redundancy and we will discuss about different design options for including in our proposal.

III. Time Redundancy of Frames

In this section we further describe the problem that motivates this work, we discuss different approaches to implement the redundancy mechanism, we propose several methods to decide the number of replicas to be transmitted and finally we make a high level description of the phases to implement in transmission and reception.

A. Description of the problem

As has been already mentioned, TSN considers spatial redundancy of the communication channel and the transmission of frames through those channels. This way, TSN aims at increasing the reliability level of the network, using the redundancy to tolerate permanent and temporary faults in the channel. Nevertheless, while spatial redundancy is especially suitable to tolerate permanent faults it is not a good solution to tolerate temporary faults [6]. The communication channel is particularly vulnerable to transient faults due to electromagnetic interference. Therefore, if spatial redundancy has been designed to provide a certain level of reliability against permanent faults, using it to tolerate transient faults will reduce the effectiveness against permanent faults, especially when they are coincident with transient ones. Additionally, once the spatial redundancy is not available due to the permanent failure of some of the channels, the system is not able to tolerate transient faults any more.

Figure 1 shows a network with two redundant paths connecting Node 1 and Node 2. Link $l_{1,2}$ is suffering a permanent fault, thus if link $l_{2,1}$, $l_{2,2}$ or $l_{2,3}$ suffers a transient fault during the transmission of a message it will be lost, jeopardizing the system operation. The most reasonable approach to deal with temporary faults in the channel is to replicate frames in time. In this work we focus on time redundancy of critical frames.

We consider to be critical those frames transmitted by critical systems. A system is critical when its failure can cause the loss of big amounts of money, machinery or human life [7]. Moreover, these systems usually have to interact with other systems or external elements, often performing a control action over them. This real-world interaction imposes time constraints, since the system should produce an adequate output at the appropriate moment and the time available to produce the output is usually shorter than in multimedia or data-oriented applications.

These time constraints make traditional schemes for frame retransmission used in Ethernet, such as Automatic Repeat Request (ARQ) [8], not suitable for control applications, since it relies in time-outs or NACKs to perform the retransmissions, and the time needed for the retransmissions could exceed the short deadlines of control applications. Therefore, we propose the proactive transmission of several copies of one frame that can meet the demanding time constraints of control applications. Moreover, processing the error closer to the source prevents it from propagating to other parts of the system, making it easier to solve it and preventing unexpected behaviours.

Proactive frame redundancy consists in sending *k* copies of a message to ensure that at least one copy is delivered to the receiver even in the presence of transient faults. Each copy of the message is called a *replica*. The number of replicas that will be transmitted strongly depends on the loss probability, target reliability and the selected approach to implement redundancy. Next we will describe two different approaches.

B. Proposed proactive frame replication schemes

The first approach consists in using an end-to-end worst case estimation to decide on a single number of replicas to be sent in all the links of the path. From now on, we will refer to this approach as *approach A*. The number of replicas is decided for the path as a whole and the transmitter is responsible for replicating and sending all frames, whereas bridges only forward each one of the replicas they receive. Nevertheless, this approach can be highly inefficient in terms of bandwidth, since a larger number of replicas than strictly needed will be transmitted, as we will discuss later on.

The second option is to use a link-based approach, in which *k* replicas are transmitted by every bridge as long as one replica reaches it. We will refer to this approach as *approach B*. This replication mechanism should be implemented in all bridges in the network and in end nodes. When receiving a correct replica, the bridge will discard all other replicas and will transmit *k'* new replicas through the corresponding ports. It is important to note that, implementing this mechanism in the bridges would allow to use nodes not implementing it while still providing a high reliability in the rest of the network. This would allow the use of legacy nodes.

Next we want to study the advantages and disadvantages of the two approaches aforementioned. To that, we decided to start by carrying out a probability analysis to compare the number of replicas that would be needed in each case to provide the same level of reliability. To that, we calculate the probability that at least one replica will be successfully delivered to the receiver. First, we need to calculate the probability of a replica being successfully transmitted through one link. We base our calculations on the bit error rate of the link, called λ. We assume that the number of links that the replicas will traverse is *L*. We also assume that the occurrence of bit errors follows a Poisson distribution. Therefore, the probability of a replica suffering a bit error only depends on the duration of the replica, called *d*, and λ. Finally, for the sake of simplicity, we assume that λ is the same in all the links.

The probability that a replica *r* is successfully transmitted through a given link in both approaches is:

$$e^{-\lambda d} \tag{1}$$

In approach A we first calculate the probability that a replica is successfully transmitted throughout the whole path, that is $e^{-\lambda dL}$. Therefore, the probability that the replica is lost in one of the links is $1 - e^{-\lambda dL}$. We now calculate the probability that all *k* replicas are lost in some of the links as $(1-e^{-\lambda dL})^k$. Finally, the probability that at least one replica is successfully transmitted through the path, is the probability that not all replicas are lost and the result is $1 - (1 - e^{-\lambda dL})^k$.

In the case of approach B, we first calculate the probability that all *k'* replicas are lost in one link, that is $(1 - e^{-\lambda d})^{k'}$. Now, we use this to calculate the probability that at least one replica is successfully transmitted through the link: $1 - (1 - e^{-\lambda d})^{k'}$. Finally, we calculate the probability that at least one replica is successfully transmitted through the complete path: $(1 - (1 - e^{-\lambda d})^{k'})^L$.

Now, in order to know which approach requires a higher number of replicas to achieve the same reliability we compare both expressions: $1 - (1 - e^{-\lambda dL})^k = (1 - (1 - e^{-\lambda d})^{k'})^L$. We express *k* as a function of *k'*.

$$k = \frac{\ln(1 - (1 - (1 - e^{-\lambda d})^{k'})^L)}{\ln(1 - e^{-\lambda dL})} \tag{2}$$

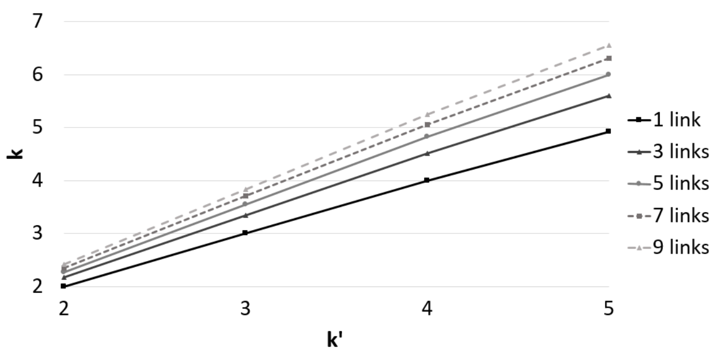

Fig. 2: k vs. k' for different number of links.

Let us assume that the speed of the link is 100 Mbps, and let also assume that the length of the messages is the minimum length allowed in Ethernet, that is 72 bytes. Therefore, the duration of the replicas is $d = (72 \times 8)/100 = 5.76\mu s$. We also assume a bit error rate of 10^{-4} errors/μs.

Figure 2 shows the evolution of k, for different values of k' and different lengths of the network, that is, different values of L. This way we calculate the number of replicas k needed by approach A in order to achieve the same reliability than using a certain k' in approach B. Note that since it is not possible to send a fraction of a message, non-integer values of K will have to be rounded to the next larger integer for their use in the actual system. We can see that the number of replicas needed by approach A is always higher when there are more than one link in the path, and that the difference between k and k' increases when the number of links increases. Nevertheless, this difference may not be enough to decide which approach should be selected.

However, approach B makes it easier to implement and handle dynamic replication schemes, where the *k'* of each link varies in different moments depending on the number of errors reported by adjacent bridges. Details about dynamic redundancy and how it will be integrated with SRP are out of the scope of this paper and are left as future work.

Furthermore, approach B allows to simplify the integration of time redundancy with the mechanisms for spatial redundancy described in IEEE802.1Qca and IEEE802.1CB. As mentioned before, these standards provide support for spatial redundancy of the network. Since the paths created by IEEE802.1CB may be different in length and reliability, using link-based replication simplifies the identification and elimination of the replicas by the bridges and prevents the replica radiation problem [9].

C. Selection of streams

The natural next step is to decide which frames should be replicated. As mentioned before, TSN relies on streams to classify the traffic and perform the QoS management. Thus, a first approach would consist in taking advantage of the existing classes of traffic to decide which streams should implement frame replication. In this sense, streams tagged as critical or alarm would be replicated since they correspond to hard real-time traffic.

Another possible approach would be to use the stream's priority to decide whether it should be replicated or not. Since traffic classes correspond to one or more priorities it would be possible to replicate just a subset of critical, class A and class B streams. Finally, another possibility would be to do per-stream management of the replication. Nevertheless, this approach requires the addition of new parameters to the stream definition to allow the identification of frames that should be replicated. Since we want this mechanism to be as orthogonal as possible to other TSN standards, we discarded the last option and we propose to use a per-priority identification scheme, since it provides higher flexibility with respect to the first scheme.

D. Phases of the time redundancy mechanism

The proposed mechanism can be divided in several phases. These phases will be different in transmission and reception. More precisely, in the transmission process we can distinguish the next two phases: (a) Identification of the stream. In this phase the bridge or node will identify the stream to which the frame belongs and will check its priority in order to see if it should be replicated. If it should not be replicated, the frame will be forwarded following the normal process, otherwise phase b will be executed. (b) Frame replication. The frame will be copied and tagged with a sequence number that will allow to identify it as a replica. The sequence number value will go from 1 to k. Thus, the sequence number can also allow to identify errors in the network since the receiver can know which and how many replicas were lost by checking the sequence number of the replicas received. This will allow to adapt k when the receiver detects a variance in the number of losses.

Regarding the reception process two phases can be identified: (a) Identification of replicas. When receiving a frame, the bridge or node will check if the frame is a replica, by using the sequence number tag. If it is a replica phase b will be executed. (b) Replica elimination and error control. Using approach A all frames will be forwarded by bridges, while with approach B only the first received replica is forwarded, replicas received later on will be discarded.

As we said, the sequence number of replicated messages can be used for error detection. To do so, nodes, and bridges if using approach B, will have a counter per port, that will be increased every time a replica is received. When a replica from a new stream is received, the receiver will compare the counter to the expected number of replicas k and the counter will be reset. If the selected approach uses static k, all elements in the network will know its value, nevertheless if we use a dynamic k, its value must be added to the message in transmission, so the receiver can know the expected number of replicas.

IV. Conclusions

TSN is a set of technical standards that is currently being developed to provide Ethernet with hard real-time, reliability and flexibility services. In the last years there has been a growing interest in increasing the connectivity of all type of devices. This trend showed up in industry in the form of IIoT and Industry 4.0. TSN represents an appealing technology to lead the integration of data-oriented and control applications for its capability to support the traffic transmitted by both types of applications over the same network.

Nevertheless, TSN does not cover some reliability aspects important for its deployment in critical systems. More precisely, TSN provides spatial redundancy and support for the transmission of replicated frames through multiple channels, to tolerate permanent and transient faults of the network. if spatial redundancy has been designed to provide a certain level of reliability against permanent faults, using it to tolerate transient faults will decrease the capability of the network to tolerate permanent faults and, once the spatial redundancy is not available due to the permanent failure of some of the channels, the system is not able to tolerate transient faults any more.

Therefore, in this work we propose to use proactive retransmission of frames to tolerate transient faults in the channel, including a discussion of different approaches to implement the redundancy mechanism and a comparison by means of a reliability analysis; a proposal of several methods to decide the number of replicas that should be transmitted; a description of several ways to decide which streams should be replicated; a high level description of the phases to implement in transmission and reception and finally a description of an error control mechanism implemented using the replication.

As future work, we plan to complete the comparison of the different approaches taking into account transient faults that could affect more than one replica; carry out a performance analysis that would help to predict the impact of replicating critical, class A and B traffic and propose an architecture of the final solution.

Acknowledgements

This work is supported in part by the Spanish Agencia Estatal de Investigacin (AEI) and in part by FEDER funding through grant TEC2015-70313-R (AEI/FEDER, UE). Ines Alvarez was supported by a scholarship of the EUROWEB+ Project, which is funded by the Erasmus Mundus Action II programme of the European Commission.

References

[1] B. Galloway and G. P. Hancke, "Introduction to industrial control networks," *IEEE Communications Surveys Tutorials*, vol. 15, no. 2, pp. 860–880, Second 2013.

[2] H. Kopetz, A. Ademaj, P. Grillinger, and K. Steinhammer, "The Time-Triggered Ethernet (TTE) Design," in *8th IEEE International Symposium on Object-oriented Real-time distributed Computing (Seattle, Washington: TU Wien)*, May 2005, p. 2233.

[3] P. Pedreiras and L. Almeida, "The Flexible Time-Triggered (FTT) paradigm: an approach to QoS management in distributed real-time systems," in *Proc. Int. Parallel and Distributed Processing Symposium*. IEEE Computer Society, 2001.

[4] W. Steiner, P. G. Pen, M. Gutirrez, A. Mehmed, G. Rodriguez-Navas, E. Lisova, and F. Pozo, "Next generation real-time networks based on it technologies," in *2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA)*, Sept 2016, pp. 1–8.

[5] *IEEE Standard for Local and metropolitan area networks, Virtual Bridged Local Area Networks, Amendment 14: Stream Reservation Protocol*, IEEE Std. 802.1Qat, 2010.

[6] Aviziens, "Fault-tolerant systems," *IEEE Transactions on Computers*, vol. C-25, no. 12, pp. 1304–1312, Dec 1976.

[7] J. C. Laprie, "Dependable computing: Concepts, limits, challenges," in *FTCS-25, the 25th IEEE International Symposium on Fault-Tolerant Computing*, June 1995, pp. 42–54.

[8] L. Peterson and B. Davie, "Getting connected," in *Computer Networks, Fifth Edition: A Systems Approach*, 2012.

[9] D. Gessner, J. Proenza, and M. Barranco, "A proposal for managing the redundancy provided by the flexible time-triggered replicated star for ethernet," in *2014 10th IEEE Workshop on Factory Communication Systems (WFCS 2014)*, May 2014, pp. 1–4.