
Real-Time Systems, 2005

DOI: 10.1007/s11241-005-2743-4

Maximizing the Robustness of TDMA Networks with Applications to TTP/C

BRUNO GAUJAL bruno.gaujal@imag.fr
ID-IMAG Laboratory, 51 avenue Jean Kuntzmann, 38330 Montbonnot, France

NICOLAS NAVET nicolas.navet@loria.fr
LORIA, Campus Scientifique, BP-139, 54506 Vandoeuvre, France

Published online: 16 August 2005

Abstract. In this study we show how one can use Fault-Tolerant Units (FTU) in an optimal way to make a TDMA network robust to bursty random perturbations. We consider two possible objectives. If one wants to minimize the probability of losing all replicas of a given message, then the optimal policy is to spread the replicas over time. This is proved using convexity properties of the loss probability. On the contrary if one wants to minimize the probability of losing at least one replica, then the optimal solution is to group all replicas together. This is proved by using majorization techniques. Finally we show how these ideas can be adapted for the TTP/C protocol.

Keywords: real-time systems, fault-tolerance, TDMA, replica, in-vehicle network, TTP/C

1. Introduction

Context of the study. Multi-access protocols based on TDMA (Time Division Multiple Access) are widely used in communications systems. TDMA based protocols are particularly well suited to real-time applications since they provide deterministic access to the medium and thus bounded response times. Moreover, their regular message transmissions can be used as "heartbeats" for detecting node failures. There exists several variants of the TDMA scheme, in this paper we consider the synchronous TDMA scheme as adopted by the TTP/C protocol (TTTech Computertechnik GmbH, 2003). The stations have access to the bus in a strict deterministic sequential order, each station possesses the bus for a constant period of time called a *slot* during which it has to transmit one frame. The sequence of slots such that all stations have access once to the bus is called a *round*.

The use of TDMA based protocols is considered in high-dependability real-time applications where fault tolerance and guaranteed response times have to be provided. Examples of such applications are "brake-by-wire" and "steer-by-wire" in-vehicle applications (see Dilger et al., 1998 or Wilwert et al., 2004) or avionic applications. In such so called "X-by-wire" applications, mechanical and hydraulic components are replaced by computer control which has to be fault-tolerant. A Fault-Tolerant Unit (FTU) is a set of two or more nodes that performs the same function and thus may tolerate the failure of one or more of its constituent stations. Actually, the role of FTUs is two-fold considering the type of failure of the stations. They make the system resilient in the presence of *transmission errors* (some frames may be still be correct while others are corrupted). They also provide a way to fight

against *measurement and computation errors* occurring before the transmission (some node may send the correct values while others may make errors). In the following we will see that according to which role is the most important, the optimization will lead to very different solutions.

Embedded systems may suffer from strong EMI (electro-magnetic interferences) which may represent a serious threat to the correct behavior of the system. For instance, in automotive applications, the EMI (Noble, 1992; Zanoni and Pavan, 1993) can either be radiated by some in-vehicle electrical devices (e.g., switches or relays) or come from a source outside the vehicle (radio, radars, flashes of lightning, ...). EMI could affect the correct functioning of all the electronic devices but the transmission support is a particularly "weak link" and the use of an all-optical network, which offers very high immunity to EMI, is not generally feasible because of the low-cost requirement imposed by the industry (see Barrenscheen and Otte (1997) for more details on the electro-magnetic sensitivity of different types of transmission support). Even with a redundant transmission support, such as in TTP/C, the network is not immune to transmission errors since a perturbation is likely to affect both channels in quite a similar manner since they are identical and very close one to each other. Unlike CAN (Controller Area Network—(ISO, 1994)), TDMA do not provide automatic retransmission for corrupted frames and their data is actually lost for the application.

Goal of the paper. The problem we address in this study is to find the best allocation of the slot of each station in the round in such as a way as to maximize the robustness of the system against errors. The solution to this slot allocation problem has to take into account the fact that a data will be sent by more than one node in the same round (by all nodes of the FTU) and that it might be sent several times by a same node (in successive rounds) when the production period of the data is greater than the length of a round. We consider two distinct objectives:

Objective 1: Minimize, for each FTU, the probability that all frames of the FTU carrying the same information will be corrupted. In the rest of the paper, this probability will be termed the "loss probability" and denoted by $\mathbb{P}_{\text{all}}$.

Objective 2: Maximize, for each FTU, the probability that at least one frame of each station composing the FTU is successfully transmitted during the production period of a data. For this objective, we will assume that the production period of the data is equal to the length of a round (see Section 2.2 for a justification). Under this assumption, it comes to minimizing, for each FTU, the probability that one (or more) frame of the FTU will be lost during a round. The corresponding probability is denoted by $\mathbb{P}_{\text{one}}$.

As it will be further discussed in Section 2.3, the two objectives correspond to well-defined situations in the field of fault-tolerance that are distinguished with regard to the concept of "fail-silence". It will also be shown that the fulfillment of these two objectives at the same time is incompatible.

Assumptions on the error model. In this study, we will consider an error arrival process where "bursts" of transmission errors may occur. This is very likely in the context of in-vehicle multiplexing applications.

If successive transmission errors are not correlated (i.i.d.), it is clear that the location of each individual slot of an FTU has no influence on the loss probability since each slot has the same probability of being corrupted independently. However, in practice, transmission errors are highly correlated and one observes bursts or errors leading to successive transmission errors. The assumptions made for the error arrival process will thus influence the solution to the problem of locating the FTU slots. We will consider an error model that can take into account both error frequency and error gravity which generalizes a model proposed in Navet et al. (2000). Here are the assumptions on the perturbation errors made in the rest of the paper:

$\langle A_1 \rangle$ Each time an EMI occurs, it will perturb the communications on the bus during a certain duration and each bit transmitted during this perturbation is corrupted with some probability π. If a perturbation overlaps a whole frame, then we assume that the probability that the frames remains uncorrupted is negligible (with $\pi = 0.5$ and a 100 bits frame, this probability is about 10^{-30}).

$\langle A_2 \rangle$ The starting times of the EMI bursts are independent random variables, uniformly distributed over time.

The results achieved for Objective 2 are valid for all possible distributions of the size of the bursts (provided they remain independent of the starting point of the perturbation). Objective 1, however, cannot be tackled without some hypotheses on the distribution of the size of the bursts. In the following, we consider:

$\langle A_3 \rangle$ The size of each EMI burst is exponentially distributed.

Without further knowledge on the considered application and its environment, assumptions $\langle A_1 \rangle$ and $\langle A_2 \rangle$ are rather reasonable. Assumption $\langle A_3 \rangle$ is more technical and will be used in the proofs of Section 3.1 (Objective 1). We would like to point out that the guidelines provided for Objective 1 should be valid for a large class of distributions, not only for the exponential one and we will give the tools to the application designer for checking whether the distribution of the bursts' length, corresponding to its particular context, belongs to this class or not.

Related work. The Time-Triggered Architecture (TTA—see Kopetz, 1997; Kopetz et al., 2001) has been designed for high-dependability real-time systems such as automotive applications. The TTP/C protocol (TTTech Computertechnik GmbH, 2003), which is a central part of the TTA, possesses numerous features and services related to dependability such as the bus guardian (Temple, 1998), the group membership algorithm (Pfeifer, 2000) and support for mode changes (Kopetz et al., 1998). The TTA and the TTP/C protocol have been designed and extensively studied at the Vienna University of Technology. Closely related to our proposal is the work described in Grünsteidl et al. (1991) where the reliability of the

transmission on a TTP network is studied with the taking into account of transmission errors on the bus as well as failures in the TTP nodes. Under the assumption that all failures and transmission errors are statistically independent, a measure of the reliability of the transmission is given in terms of Mean Time To Failure (MTTF) where a communication failure for an FTU is defined as the loss of all messages of an FTU sent in the same round. From the MTTF of each individual FTU, a global measure of the reliability of the system is derived.

There exist two main differences with our work. One concerns the assumptions made on the perturbations and the second the data production. In Grünsteidl et al. (1991) the errors are assumed to be independent, the location of the FTU slots has thus no influence and is not considered. Here on the contrary, we take into account the burstiness of the perturbation process. Hence the time allocations of the FTU replicas will have a big effect on the transmission error probabilities.

As for the data production issue, in Grünsteidl et al. (1991) failure is decided on a per round basis while in this paper this event will be assessed considering the frames sent in a production cycle of a data. Indeed, the same data might be transmitted during successive rounds and the fact that no frame of an FTU has been successfully transmitted in one round does not necessarily imply a communication failure because the same data is also sent in following rounds (see Section 2.2).

The second difference with Grünsteidl et al. (1991) is that we do not merely compute the reliability of a given system but also provide a way to optimize it via time allocation of the replicas. This does not require any modification of the protocol or of the parameters of the system. Just playing with the temporal allocation of replicas provides a substantial gain in resilience (around 80% in many cases) as seen in Section 3.

Finally another novelty with respect to previous work comes from the proof techniques. They are based on *multimodularity* and *bracket sequences* for Section 3 and on *majorization* and *Schur convexity* for Section 5. To the best of our knowledge, these notions have never been applied in this framework and they may prove to be useful for several other related problems.

2. Framework of the Study

In this section, we first describe the Medium Access Control (MAC) protocol, namely the synchronous TDMA scheme, then the model of the application and the notations used. Then, we justify the two distinct objectives that were identified with regard to the concept of "fail-silence".

2.1. MAC Protocol Description

Throughout this paper, we will consider the synchronous TDMA protocol. The number of *stations*, S, is static and the stations have access to the bus in a strict deterministic sequential order. Each station possesses the bus for a constant period of time called a *slot* during which it has to transmit one *frame*. The size of the slots is not necessarily identical for all stations but successive slots belonging to the same station are of the same size. The sequence of slots such that all stations have access once to the bus is called a *round*, as shown in Figure 1.

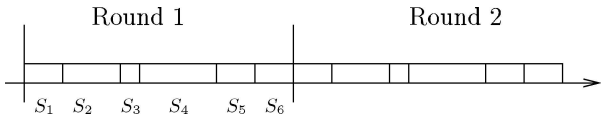

Figure 1. A round is made of S slots (here $S = 6$), one slot per station.

The time needed to transmit one bit over the bus is taken as the time unit. In the following all time quantities are given using this time-bit as unit.

2.2. Application Model

2.2.1. Fault-Tolerant Units

To achieve fault-tolerance, that is the capacity of a system to deliver its service even in the presence of faults, some nodes are *replicated* and are clustered into *Fault-Tolerant Units* (FTUs). An FTU is a set of several stations that perform the same function and each node of an FTU possesses its own slot in the round so that the failure of one or more stations in the same FTU might be tolerated. The stations forming an FTU are called *replicas* in the following. For the sake of simplicity, a non-replicated station will also be termed an FTU (of cardinality one).

2.2.2. Construction of the Round

One denotes by $\mathcal{F}$ the set of FTUs : $\mathcal{F} = \{A, B, C...\}$ and C_A is the cardinality of FTU A, i.e. the number of stations forming FTU A. The size (in bits) of the slots of all the stations in A is the same and is denoted by h_A. By definition, the total number of bits in a round, denoted R, is equal to:

$$R = \sum_{A \in \mathcal{F}} C_A h_A.$$

The whole problem consists in choosing the position of the slots of all stations forming an FTU in a round. This is done under the form of a binary vector x^A of size R (called an allocation for A) defined by

$$\forall 1 \leq i \leq R, \quad x_i^A = \begin{cases} 1 & \text{if some station in } A \text{ transmits at time-bit } i \\ 0 & \text{otherwise.} \end{cases}$$

Note that the construction of x^A must respect several constraints. First the binary vector x^A must be made of C_A "blocks" of ones, each of size h_A to correspond to an allocation of all the slots of A. Second, the allocations of all the FTUs must be *compatible*, meaning that

the same bit cannot be allocated to two different FTUs. Finally all bits in a round must be allocated to some FTU.

2.2.3. Data Production

Each frame contains some data whose value is periodically updated as it is generally the case in control applications. For instance, in a typical in-vehicle application, a frame sent by the engine controller may contain the number of revolutions per minute value plus the engine temperature. Since they are replicas, all nodes of an FTU update their data with the same period denoted by T_A and called a *production cycle*. The data sent during one production cycle is also called a *message* in the following. It is also assumed that all nodes of a FTU are synchronized, using the global time service requested by the communication protocol, so that at each point in time each node of an FTU sends the data corresponding to the same production cycle.

The length of the TDMA round R is a function of the number of nodes, of the maximal size of the message sent in each slot, and on some characteristics of the network and of the communication controllers. Theoritically, the value of R is thus not correlated with the production period of the data. If $\exists A \in \mathcal{F}$ s.t. $T_A < R$ then some data may not be transmitted which is generally unacceptable. If $\forall A \in \mathcal{F},\ T_A > R$ then the same data is transmitted in more than one round. Also, if the beginning of the production cycle does not correspond to the beginning of a round, then data corresponding to different production cycles may be transmitted in the same round as it is the case in the first and third round of the example drawn on Figure 2.

In practice, it is very convenient for the application designer to set the production period of an information equal to the length of a round or a multiple of the length of the round (see, for instance, the steer-by-wire case study in Wilwert et al. (2004)). For instance, it guarantees that all successive informations that are produced are transmitted in exactly the same number of frames. For a single FTU, this is made possible by inserting idle time after the transmission of a frame so that the duration of a slot or a round can take an application related value.

2.3. Which Objective with Respect to Fail-Silence?

The number of replicas per FTU which is required to tolerate k faults heavily depends on the behavior of the individual components (Dilger et al., 1998). For instance, if the failure

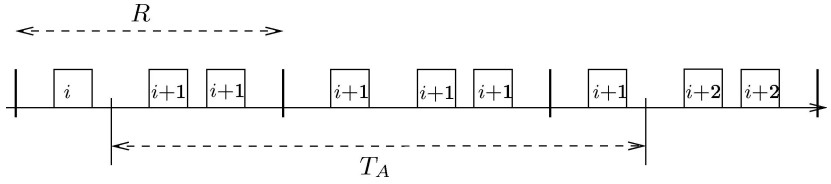

Figure 2. Three successive rounds. Only the slots allocated to the FTU A of cardinality 3 are shown.The message corresponding to the $(i+1)$th production cycle is sent over 3 rounds.

of k nodes must be tolerated, the least necessary number of replicated nodes is $k + 1$ when all nodes are *fail-silent*. A node is said fail silent if

1. (a) it sends frames at the correct point of time (correctness in the time domain) and (b) the correct value is transmitted (correctness in the value domain),

2. or it sends detectably incorrect frames (e.g., wrong CRC) in its own slot or no frame at all.

TTP/C provides very good support for the requirements 1(a) and 2 (whose fulfillment provide the so-called "fail-silence in the temporal domain") especially through the bus guardian concept, while the value domain is mainly the responsibility of the application. The reader is referred to Brasileiro et al. (1996), Dilger et al. (1998), Temple (1998), and Poledna et al. (2000) for good starting points on the problem of ensuring fail-silence. For FTUs composed of a set of fail-silent nodes, the successful transmission of one single frame for the whole set of replicas is sufficient since the value carried by the frame is necessarily correct (i.e., one can safely consume it). In this case, the objective to achieve with regard to the robustness against transmission errors is the minimizing of $\mathbb{P}_{\text{all}}$, that is the probability that all frames of the FTU (carrying data corresponding to the same production cycle) will be corrupted.

In practice, it is generally impossible to guarantee that nodes are fail-silent with probability one; this can be due to possible measurement errors, possible calibration problems or simply sensors can disagree because they are physically distributed (see Poledna, 1996; Brasileiro et al., 1996 for the problem of ensuring fail-silence). Two types of faults are identified: faults in the value domain (e.g., measurement problems) and faults in the time domain (e.g., transmission problem). A fault in the value domain corresponds to the case where the value of an information received is wrong (the sender node is thus non fail-silent). When an information is not received or not on time (e.g., a frame has been corrupted by an EMI), one talks of a failure in the time domain.

When conceiving a system that has to be fault-tolerant, it is crucial to carefully define the fault-hypothesis. Precisely, one has to state what has to be tolerated. For many industrial systems, in particular in the context of automotive systems, due to the constraint of energy, weight, size and cost, one can reasonably not expect a fault-hypothesis stating that more than one failure (either a fault in the value domain or a fault the time domain) has to be tolerated, even for X-by-Wire systems (see, for example, the case studies in Wilwert et al. (2004) or X-by-Wire Consortium (1998) on page 12 and the fault-hypothesis of TTP/C in TTTech Computertechnik GmbH (2003) on page 27). For instance, tolerating two faults in the value domain would necessitate an FTU of cardinality 5 for performing a majority vote!

Faults that are not covered by the fault-hypothesis are generally treated with some predefined procedure (default procedure or so called Never Give Up procedure, see Rushby, 2003) but the probability to be outside the fault hypothesis has to be minimized. To this end, since there is no way to avoid faults in the value domain for most FTUs, one has to minimize the probability that a fault in the temporal domain occurs for an FTU. This comes

to maximize the probability that all replicas are received, namely $\mathbb{P}_{\text{one}}$, which is the second objective of our study.

3. Minimising the Loss Probability: The General TDMA Case

In this section, we investigate the problem of minimizing the loss probability $\mathbb{P}_{\text{all}}$, the probability that all frames of a FTU carrying the same information is corrupted. In Section 3.1, we focus on the optimal policy for one FTU. In Section 3.2, we consider all FTU combined. Some cases can be treated analytically in an optimal way (see Section 3.2.1). For the other cases, an heuristic is proposed in Section 3.2.2 and its performances are assessed by simulation.

3.1. Optimal Allocation for a Single FTU

One focus here on a given FTU, say A, made of $K := C_A$ replicas per round, all of size $h := h_A$. The problem is to find an allocation x of the K replicas over one production period that minimizes the probability $\mathbb{P}_{\text{all}}$ that all replicas carrying the same message are lost, regardless of the other FTUs. The proof technique uses two notions: multimodularity and "bracket" sequences.

3.1.1. Optimization using Multimodularity and Bracket Sequences

Let x be a binary vector of size R. Its *density* is $(1/R)\sum_{i=1}^{R} x_i$. A binary vector is a *block-vector* with blocks of size h if $x_i = 1$ only in intervals of h consecutive values. A *block shift* is a vector δ_i such that $\delta_i(n) = 0$ for all n except $\delta_i(i) = +1$ and $\delta_i(i+h) = -1$. Basically if x is a block-vector, then $x + \delta_i$ is also a block-vector similar to x with one of its blocks shifted to the left by one unit as in the following example with blocks of size 3.

$$\begin{aligned} x &= (0,0,0,0,1,1,1,0,1,1,1,0,0,1,1,1,0)\\ x+\delta_4 &= (0,0,0,1,1,1,0,0,1,1,1,0,0,1,1,1,0). \end{aligned}$$

A *global shift of size* j , s_j is an operation on vectors that shifts all values to the left by j (modulo the size of the vector) as in the following example:

$$\begin{aligned} x &= (0,0,0,0,1,1,1,0,1,1,1,0,0,1,1,1,0)\\ s_2(x) &= (0,0,1,1,1,0,1,1,1,0,0,1,1,1,0,0,0). \end{aligned}$$

A real function $F(x)$ is *block-multimodular* with blocks of size h if the following inequality holds for all block-vectors x.

$$\forall i \neq j \quad F(x+\delta_i) + F(x+\delta_j) \geq F(x) + F(x+\delta_i+\delta_j) \tag{1}$$

as soon as $x+\delta_i, x+\delta_j, x+\delta_i+\delta_j$ are all block vectors.

A *bracket sequence* v with density a/b is a binary vector of size b such that

$$v_n = \lfloor na/b \rfloor - \lfloor (n-1)a/b \rfloor. \tag{2}$$

For example, the bracket sequence with density $3/8$ is

$$v_{3/8} = (0, 0, 1, 0, 0, 1, 0, 1).$$

A *block bracket vector* x with density $ha/(b+(h-1)a)$ with blocks of size h is constructed from v in the following way.

- Start with x empty.
- If $v_i = 1$, then $x := x.1 \cdots 1$, (with h ones concatenated at the end of x).
- If $v_i = 0$, then $x := x.0$.

Continuing the example, the block bracket vector with blocks of size 3 with density $9/14$ is derived from $v_{3/8}$ using the procedure above:

$$x = (0, 0, 1, 1, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1).$$

Note that x is not equal to $v_{9/14}$, the bracket sequence with density $9/14$, since $v_{9/14}$ does not contain blocks of size 3:

$$v_{9/14} = (0, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1).$$

One can apply a general optimization theorem given in Altman et al. (2000b) to the block case. This theorem relates the minimizing of multimodular functions with bracket vectors.

Theorem 1 (Altman et al., 2000b)**.** *Let F be a block-multimodular function, then consider the average value over all possible shifts (also called the shift invariant version of F), namely the function $G(x) := 1/R \sum_{i=0}^{R-1} F(s_i(x))$. Then G is minimized over all block vectors with density d by the block bracket vector of density d.*

Basically, multimodularity is the counterpart of convexity for discrete function ($f : \mathbb{Z}^m \to \mathbb{R}$). For more details on multimodularity and bracket sequence, the reader might refer to Hajek (1985) and Altman et al. (2000b). The next step is to prove that the loss probability is a block multimodular function.

3.1.2. Multimodularity of the loss probability

Here, we prove that the loss probability $\mathbb{P}_{\text{all}}$ is block-multimodular. In addition to assumptions $\langle A_1 \rangle$, $\langle A_2 \rangle$ and $\langle A_3 \rangle$ that concerns the error model, the following assumptions are made on the production of data:

$\langle A_4 \rangle$ The production period of an information is not necessarily equal to one round but is assumed to be a multiple of the round length.

$\langle A_5 \rangle$ Furthermore, it is assumed that there is no synchronisation between production and transmission: in the initialization phase, the very first information is made available at a random point in time in the first round.

In a first step, we consider a single error burst, then the result will be extended to the case where several perturbations may occur.

Lemma 1. *Under the foregoing assumptions, the probability $\mathbb{P}_{\text{all}}$ of losing all replicas of FTU A with a single perturbation is block multimodular, with blocks of size h_A.*

The proof of Lemma 1 is given in Appendix A.

We believe that the multimodularity property holds for more general distributions of the error size. By mimicking the proof of lemma 1, the application designer can check whether the distribution of the bursts' length, corresponding to its particular context, induces this property. However, we checked that it does not hold for Pareto distributions ("heavy tailed" distribution). The exponential distribution assumption is also crucial in the proof the next theorem.

Theorem 2. *Under the assumptions $<A_1>, <A_2>, <A_3>$, $<A_4>$ and $<A_5>$, the probability $\mathbb{P}_{\text{all}}$ of losing all replicas of FTU A forming the same message is minimized if the replicas are allocated over each round according to a block bracket sequence.*

Proof: By considering only bursts between time 0 and round C, we can assume using $\langle A_2 \rangle$ that all the bursts start at independent random times, uniformly distributed. The fact that each individual burst is of exponential size ($\langle A_3 \rangle$), makes is possible to discard overlaps.

The second step of the proof consists in noticing that $\mathbb{P}_{\text{all}}$ does not depend on shifts of the allocation sequence x. This means it is equal to its shift invariant version. Finally, Theorem 1 together with Lemma 1, which is true for each burst independently, show that the function $\mathbb{P}_{\text{all}}$ is block-multimodular and is minimized if the allocation of the replicas forms a block bracket sequence. □

The memoryless property of the exponential distribution allows to discard overlaps. This will not be possible with other distributions. However, if perturbation overlaps are so unlikely that they can be neglected, block bracket sequences still provide optimal allocations for all distributions such that the loss probability is multimodular (see Lemma 1).

3.2. Slot Allocation for Several FTUs

In this section, we consider several FTUs together and try to find an allocation for all of them simultaneously. An optimal allocation for each FTU constructed using Theorem 2 is not always feasible since the allocations may be conflicting with each other (if two allocations have at least one bit in common). In the following, we distinguish the case where it is possible to allocate all FTUs optimally and the case where this is not possible and where compromises have to be found.

For example, consider three FTUs A, B, C of cardinalities $1, 2, 3$ respectively over a round of size 6. The optimal allocations of C are $\cdot C \cdot C \cdot C$ or $C \cdot C \cdot C \cdot$ and the optimal allocation of B are $\cdot B \cdot \cdot B \cdot$ or $B \cdot \cdot B \cdot \cdot$ or $\cdot \cdot B \cdot \cdot B$ while any allocation of a single A over a round is optimal. However note that B and C cannot be optimally scheduled together since all six combinations between their individual optimal allocations have conflicts. Now consider the case where the three FTUs A, BC, D have cardinalities $2, 2, 4, 3$ respectively over a round of size 11. The global allocation $CBDCACDBCAD$ is optimal for all FTUs.

In the following we give some conditions under which a global allocation is optimal and how to construct it. When this is not possible, we give some heuristics that provide "good" allocations.

3.2.1. Some optimal cases

Here, we give some conditions under which it is possible to allocate each FTU optimally with no conflicts and provide an algorithm to construct such allocations.

Condition (C): the set of replicas can be split into two subsets such that each subset induces a sub-sequence which is an exact covering sequence (i.e., a sequence in which each replica of a FTU appears periodically—see Altman et al. (2003) for more details).

The algorithm to construct an optimal allocation is the following:

1. Find a partition of the replicas into two subsets such that (C) is verified.
2. Construct an exact covering sub-sequence for each sub-set independently.
3. Merge the two sequences using a bracket sequence with the appropriate rate (number of slots in subset 1/overall number of slots).

For example, consider our second example above (A, B, C, D of cardinalities $2, 2, 4, 3$ respectively over a round of size 11).

1. Split the set of replicas into $\{A, B, C\}$ and $\{D\}$.
2. Build the exact covering sequence $CACBCACB$ for the first set. The sub-sequence corresponding to $\{D\}$ is DDD which is also an exact covering sequence.

3. Merge the two sequences according to the bracket word 00100010001 ('0' positions are for FTUs of subset 1 while '1' belongs to subset 2) which yields $CBDCACDBCAD$.

In addition, there exist some cases where the cardinalities take more than two values and a bracket allocation is still possible for all FTUs. If the cardinalities are of the form $1, 2, 4, 8, 16, \ldots, 2^k$ (all powers of 2) then, it is possible to find bracket allocations for all FTUs. The Fraenkel conjecture (see Altman et al., 2000a) says that these are essentially the only cases where the superposition of several bracket allocations is possible without conflicts.

At this point, we should point out that the case with up to two different cardinalities, thus verifying condition (C), should fulfill most of the needs. In a system where only a subset of nodes are critical from a the point of view of the dependability, FTUs will generally be of cardinality one (non-critical nodes) and two (critical nodes). In the context of X-by-Wire applications where dependability constraints are stringent, two different cardinalities should also generally be sufficient. For instance the prototype designed in the Brite Euram III project "Safety related Fault Tolerant Systems In Vehicle" (see Dilger et al., 1998) is composed of nodes of cardinalities two (the steering wheel actuator and the steering control unit) and three (steering actuators).

3.2.2. General case

As mentioned before, the cases where condition (C) is verified are rather common in practice. Nevertheless, it could happen that a more difficult configuration arises. In general, it is not possible to allocate the slots of all FTUs according to bracket sequences without getting conflicts. Two possible strategies can be considered:

1. One can deliberately favor a subset S of particularly critical FTUs having all the same cardinality K and the same size h. In this case, the slots of those FTUs are allocated optimally (regarding the loss probability) while the slots of the others FTUs are fit in the remaining free places. The allocation is given by any block bracket sequence (see Eq. 2) of density $\alpha = \#(S)Kh/R$ as done in the previous paragraphs.

2. No FTUs are of special importance and a solution minimizing the loss probability for the set of all messages of the system has to be found.

In the rest of this section, we will consider the latter objective and provide a low-complexity heuristic algorithm whose performance are evaluated against random allocation and optimal allocations.

3.2.2.1. Description of the heuristic. As for a bracket sequence, the basic idea of this heuristic is to spread the replicas of a same FTU as evenly as possible over time.

For each FTU A with cardinality C_A and frames of size h_A, we define the density of frames per bit: $u^A := C_A h_A/R$. Intuitively, u^A is the number of frames belonging to FTU

A that should be transmitted per bit. The sum of the densities up to bit k for FTU A is $U_k^A := ku^A$. We denote by n_i^A the number of bits FTU A has already been allocated up to step i (including step i). At each step, an FTU will be allocated the number of bits necessary to send its frame. In the following, $s(i)$ indicates the FTU chosen at step i while $b(i)$ is the total number of bits already allocated at step i.

1. Initialization step: $n_0^A := 0$, $b(0) := 0$ and $i := 1$.

2. At step i the FTU for which the difference between the number of "due" bits and the previous allocation is maximum is selected:

$$s(i) := \text{argmax}_{A \in F}\left(U^A_{b(i-1)+1} - n^A_{i-1}\right).$$

3. The next $h_{s(i)}$ bits are allocated for FTU $s(i)$.

4. Perform the updates $b(i) := b(i-1) + h_{s(i)}$, $n_i^{s(i)} := n_{i-1}^{s(i)} + h_{s(i)}$ and $n_i^A := n_{i-1}^A$ if $A \neq s(i)$.

5. if $b(i) = R$ stop else $i := i + 1$, go to item 2.

The algorithmic complexity of the heuristic allocation is linear in the number of bits of a round. Note that a similar construction based on density has been successfully used for defining a policy that shapes real-time traffic in Gaujal and Navet (1999).

3.2.2.2. Performance evaluation. To assess the robustness of the allocations given by the heuristic, simulations were performed against random allocations and optimal allocations with $\mathbb{P}_{\text{all}}$ being the performance metric.

A configuration is defined by a number of FTU and the cardinality of each FTU. We distinguish two classes of problem according to the number of FTUs on the network: for a "medium size problem" there are at least 3 FTUs and at most 6 FTUs while in a "large size problem" there are up to 12 FTUs. Two hundreds configurations were randomly generated with FTUs having a cardinality between 2 and 4. For each configuration, we randomly pick up 100 slots (in the 1000 first rounds) where a data is transmitted for the first time. The duration of the production cycle of the data is equal to 3 rounds and is denoted by T. Then for each selected start of transmission, 500 bursts of errors are generated with $\pi = 1$ and a size exponentially distributed of mean $c \cdot T$ with $c \in \{0.5, 1, 1.5, 2\}$. If the burst of errors starts before the end of transmission of the first replica and finishes after the start of transmission of the last replica, the data is lost. The results of the experiments with random allocations and the proposed heuristics are shown on Figure 3.

The use of the proposed heuristics greatly diminishes the total number of lost data (up to 79%) knowing that there are cases where the size of the burst is such that the data cannot be

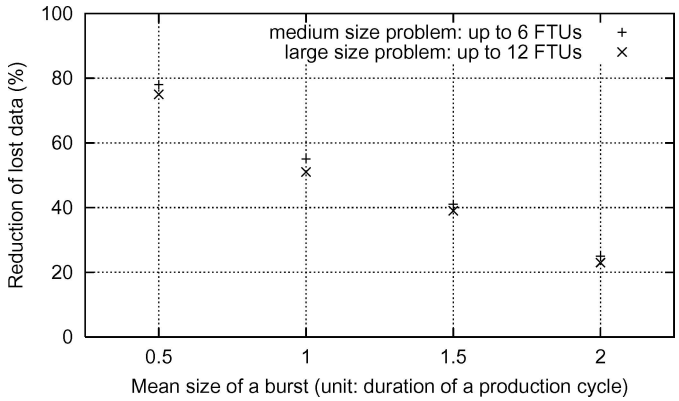

Figure 3. Reduction of the number of lost data when the heuristic is used instead of a random allocation. The mean burst size ranges from 0.5 to 2 times the length of a production cycle which is chosen equal to 3 TDMA rounds.

transmitted whatever the allocation. This fact explains why the efficiency of the heuristic tends to be lower when the size of the burst is becoming larger.

We now evaluate the behavior of the heuristic with regard to the optimal bracket allocation. We consider a case with only two replica cardinalities. Using the previous section, we know that we can construct on optimal allocation. The heuristic allocation will not necessarily find this optimal allocation and we want to measure how well it performs compared to the optimal.

One considers 200 random configurations of the medium size problem for which the optimal allocation is known (i.e., number of FTUs cardinalities is less than 3). The conditions of the experiment are the same as in Section 5.3 except that the number of first transmission slots that are selected is equal to 1000 (in the first 2000 rounds) and that 5000 bursts of errors are randomly generated. The loss of performance against the optimal solution is shown on Figure 4.

The average loss of performances with regard to the optimal solutions is small (less than 11% on this set of experiments) and it logically decreases when the size of the bursts becomes larger. That good behavior of the heuristic on configurations with less than 3 different cardinalities is a positive element with regard to its performance on arbitrary configurations.

4. Minimising the Loss Probability: The TTP/C Case

In this section, we investigate the problem of minimizing the loss probability $\mathbb{P}_{\text{all}}$ on TTP/C. The problem has been studied the previous section for the general synchronous TDMA case but, as it will be discussed below, some features of TTP/C changes the solution with respect to the general TDMA case. In fact, it makes it easier to reach optimal allocation for all FTUs together compared to the pure synchronous TDMA network.

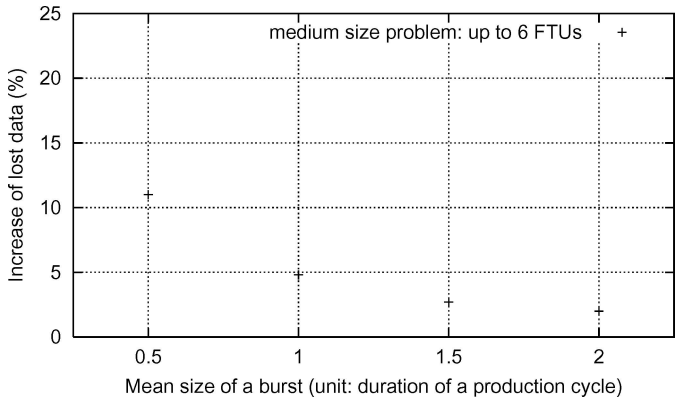

Figure 4. Increase of the lost data when the heuristic is used instead of the optimal allocation.

4.1. TTP/C Error Handling Mechanisms

The TTP/C protocol includes powerful but complex algorithms such as the clique avoidance and membership algorithms. In this paragraph, we give a simplified description of the functioning schemes of TTP/C version 1.0 that are related with transmission error handling and that might a priori interfere with our analysis. For instance, TTP/C defines the concept of "shadow" node. A shadow node replaces a defective node but does not possesses its own slot in the round. This redundancy scheme does not protect against transmission errors and we won't consider them in the rest of the paragraph.

A TTP/C controller is always in one of the nine states defined by the protocol (see TTTech Computertechnik GmbH, 2003). Three are of particular importance in our context:

1. the "active" state which is the normal functioning state,

2. the "passive" state: the controller is synchronized and can receive frames but no transmission is allowed,

3. the "freeze" state: the execution of the protocol is halted and the reintegration process will not be started before the controller is turned on by the application software.

The protocol distinguishes frames with and without "C-State". The C-State is a collection of control data that describes the state of the network as seen by the sending node: current time, current operating mode, membership of the stations (i.e., their operational state) ... The most important TTP/C functioning schemes related to transmission error handling are listed below:

1. Lost of membership due to a incorrect transmission: if a frame is corrupted during its transmission the sender loses its membership and enters the passive state. It waits in the passive state until it can re-acquire its slot. To re-acquire a slot the controller must have received the "minimum integration count" (MIC) correct frames (the first correct frame must contain an explicit C-state). The value of the MIC should be set at least to two.

2. Maximum Membership Failure Count (MMFC) check: if a node do not possess its membership in MMFC successive sending slots, then the controller terminates its operation by entering the "freeze state". It is an optional feature since MMFC can be set to zero which means no verification.

3. Re-integration of a node (transit from freeze state to passive state): a "frozen" node must wait until the application sets the Controller On (CO) field to the value "on". Then it must listen to a valid frame containing explicit C-state before entering the passive state. Then the node has to re-acquire its slot as described in point 1.

4. Clique avoidance algorithm: before starting to send a frame, a node must verify whether the number of frames that have been successfully sent in the last S slots (where S is the number of slots in the round so that it includes its own last transmission) is greater than the number of incorrect frames. In the latter case, the node enters the "freeze state" otherwise it transmits its frame and reset its counters. This rule will be termed the "majority rule".

4.2. Minimizing $\mathbb{P}_{\text{all}}$ on TTP/C

The TTP/C rules 1, 2 and 3 actually affect the value of $\mathbb{P}_{\text{all}}$ but not which allocation scheme is optimal. However, the majority rule of TTP/C (item 4 above) simplifies the solution with respect to the general TDMA case.

Let us consider the following algorithm: one constructs two stacks S_1 and S_2 of slots. For each FTU i with C_i replicas, push $\lfloor C_i/2 \rfloor$ slots in the largest stack and $\lceil C_i/2 \rceil$ slots in the smallest stack. The allocation x_{stack} is constructed by concatenating S_1 and S_2. The construction is illustrated by Figure 5.

Theorem 3. *On TTP/C, under assumptions $\langle A_1 \rangle$ and $\langle A_2 \rangle$, the x_{stack} allocation minimizes $\mathbb{P}_{\text{all}}$.*

Proof: The replicas of an FTU can be corrupted by several perturbations each touching exactly one frame. Since starting points of EMI bursts are uniformly distributed over time (assumption $\langle A_2 \rangle$), this probability is equal under all allocations. Several replicas can also be corrupted by a same perturbation with a probability decreasing when the distance between the replicas inside the round becomes larger. The allocation x_{stack} has the following property: each FTU with more than two replicas has two replicas separated by at least $\lfloor S/2 \rfloor$ slots. Now, as soon as two replicas of the same message are allocated more that $\lfloor S/2 \rfloor$ slots apart, no single perturbation can destroy both of them without freezing all the nodes of the

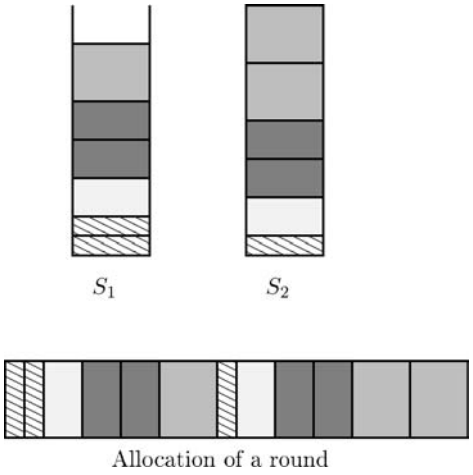

Figure 5. Construction of the optimal allocation x_{stack}.

network. It is thus useless to consider a distance between replicas larger than $\lfloor S/2 \rfloor$. This implies that x_{stack} is optimal. □

The following Corollary of practical interest can be deduced from Theorem 3.

Corollary 1. *If the probability to have more than one perturbation in the same round is sufficiently low, and because of the TTP/C majority rule, it is useless to have more than two replicas per FTU if the objective is to minimize the corruption of all the replicas.*

5. Minimizing the Probability that at Least One Replica is Corrupted

The objective here is to minimize the probability that one or more replicas of a FTU become corrupted. The results of this section hold for general error model since only assumption $\langle A_1 \rangle$ are $\langle A_2 \rangle$ are needed. In the following, the technique used to find the optimal allocation is based on majorization and Schur convexity.

5.1. Schur convexity and majorization

Let $u = (u_1, \ldots, u_n)$ and $v = (v_1, \ldots, v_n)$ be two real vectors of size n. We denote by $(u_{[1]}, \ldots, u_{[n]})$ and $(v_{[1]}, \ldots, v_{[n]})$ the permutations of u and v such that $u_{[1]} \leq \cdots \leq u_{[n]}$ and $v_{[1]} \leq \cdots \leq v_{[n]}$. The vector *u majorizes* v ($u \succ v$) if the following conditions hold:

$$\sum_{i=1}^{n} u_i = \sum_{i=1}^{n} v_i, \tag{3}$$

$$\sum_{i=1}^{k} u_{[i]} \leq \sum_{i=1}^{k} v_{[i]} \quad k \leq n. \tag{4}$$

For example, one has $(1, 3, 5, 10) \succ (2, 4, 4, 9)$.

A function f from $\mathbb{R}^n$ to $\mathbb{R}$ is *Schur convex* (resp. *Schur concave*) if $u \succ v$ implies $f(u) \geq f(v)$ (resp. $f(u) \leq f(v)$). For more details on these notions, the reader can refer to Marshall and Olkin (1979).

5.2. *Schur Concavity of* $\mathbb{P}_{\text{one}}$

In this section, we will show that the probability that an error burst corrupts at least one replica within a production cycle ($\mathbb{P}_{\text{one}}$) is a Schur concave function with respect to the allocation of the replicas. Using the definition of Schur concavity, this will provide directly the best allocation minimizing $\mathbb{P}_{\text{one}}$. Note that the result will be proven for arbitrary production cycles although, in our context, $\mathbb{P}_{\text{one}}$ is only meaningful for a production cycle equal to one TTP/C round.

Let x be an allocation of the K replicas forming FTU A. We denote by $t = NK$ the number of frames (of size h) composing a message for FTU A.

The quantity $I_i(x)$ denotes the interval between the end of replica r_{i-1} and the beginning of replica r_i. We denote by $I(x)$ the sequence of intervals $(I_1, \ldots, I_t)$ and by $|I(x)|$ the vector of the length of the intervals, $|I(x)| = (|I_1|, \ldots, |I_t|)$. Note that $|I_1(x)| + \cdots + |I_t(x)| = N(R - Kh)$ does not depend on the allocation x.

Lemma 2. *Let us consider a single error burst starting at a random time uniformly distributed over one round. Let x and x' be two allocations of A. If $|I(x)| \prec |I(x')|$ then the probabilities of losing at least one frame satisfy $\mathbb{P}_{\text{one}}(x) \geq \mathbb{P}_{\text{one}}(x')$.*

Proof: A replica can either be corrupted by a perturbation that starts between two replicas of the FTU or by a perturbation that starts during the transmission of a replica of the FTU. Both cases are independent and can be studied separately.

Let us first consider the first case. Note that if $t = 1$ then $|I(x)| = |I_1(x)| = N(R - Kh) = |I_1(x')| = |I(x')|$ and all allocations are equivalent since the error model is time homogeneous.

If $t \geq 2$, we renumber the intervals of x and x' such that$|I_{[1]}| \leq \cdots \leq |I_{[t]}|$ and $|I'_{[1]}| \leq \cdots \leq |I'_{[t]}|$. Using the majorization condition, one gets for all j, $\sum_{i=1}^{j} |I_{[i]}| \geq \sum_{i=1}^{j} |I'_{[i]}|$.

We now prove by induction that for all $1 \leq j \leq t$ one can construct a coupling between $I_{[1]}, \ldots, I_{[j]}$ and $I'_{[1]}, \ldots, I'_{[j]}$ such that the probability $\mathbb{P}'_j$ that an error starting in $I'_{[1]}, \ldots, I'_{[j]}$ and corrupting at least one replica is smaller than the corresponding probability $\mathbb{P}_j$ in $I_{[1]}, \ldots, I_{[j]}$. For $j = 1$, the coupling is done according to Figure 6.

After the coupling, the interval$I_{[1]}$ is split into two intervals,Z_1 and J_1 such that $I_{[1]} = Z_1 \cup J_1$ and $|I'_{[1]}| = |J_1|$. A burst starting in J_1 has the same probability of corruption that a burst starting in $I'_{[1]}$ because

– both intervals are of the same size and both are contiguous to replicas having the same length,

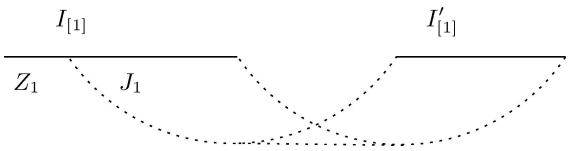

Figure 6. Coupling for the smallest interval.

– if a perturbation overlaps the whole replica then the corruption occurs with probability 1 (assumption $\langle A_1 \rangle$) under x and x' otherwise the corruption probability is also identical under x and x'.

The remaining zone (Z_1) is such that an error starting in Z_1 corrupts one replica with a non-negative probability. Therefore, $\mathbb{P}_1 \geq \mathbb{P}'_1$.

The proof continues by induction on j. The induction property is that for a given j one can construct a splitting of $I_{[1]}, \ldots, I_{[j]}$ into $(J_1, Z_1), \ldots, (J_j, Z_j)$ such that the probability that a burst starting in $J_1 \cup \cdots \cup J_j$ is larger or equal than in $I'_{[1]} \cup \cdots \cup I'_{[j]}$ and the zone $Z_1 \cup \cdots \cup Z_j$, has a non-negative total probability of corrupting a replica.

We now add $I_{[j+1]}$ and $I'_{[j+1]}$. Two cases can occur.

1. If $I_{[j+1]} \geq I'_{[j+1]}$ then one splits $I_{[j+1]}$ as it has been done for $I_{[1]}$ and $I'_{[1]}$ in Figure 6. We get new intervals Z_{j+1} and J_{j+1} and the induction remains true by using the argument given for $j = 1$.

2. If $I_{[j+1]} \leq I'_{[j+1]}$, we couple according to the following procedure. The interval $I'_{[j+1]}$ is split into two intervals U and V such that $|V| = |I_{[j+1]}|$, which are coupled together.

Note that by the majorization property, $|U| = |I'_{[j+1]}| - |I_{[j+1]}| \leq |Z_1| + \cdots + |Z_j|$. Let $k := \min\{k : |Z_1| + \cdots + |Z_k| \geq |U|\}$. We split the interval Z_k into two intervals R_k, W_k such that $|W_k| = |U| - (|Z_1| + \cdots + |Z_{k-1}|)$. The coupling is illustrated in Figure 7.

– An error starting in V has the same probability to corrupt a frame than an error starting in $I_{[j+1]}$.

– An error starting in U has a smaller probability of corruption than an error starting in $Z_1 \cup \ldots Z_{k-1} \cup W_k$ because $|V| > |J_i|$ for all $i \leq k$.

– An error starting in $I'_{[1]} \cup \cdots \cup I'_{[j]}$ has a probability of corruption smaller or equal than an error starting in$J_1 \cup \cdots \cup J_j$ by the induction hypothesis.

– An error starting in $R_k \cup Z_{k+1} \cup \cdots \cup Z_j$ has a non-negative probability of corruption.

In total, $\mathbb{P}_{j+1} \geq \mathbb{P}'_{j+1}$.

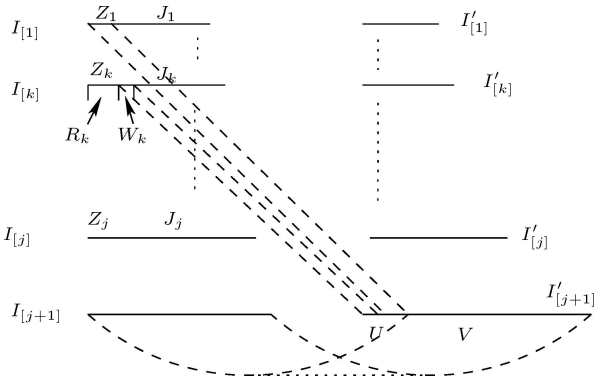

Figure 7. Coupling when $I_{[j+1]} \leq I'_{[j+1]}$.

Finally, the induction assumption is carried one more step by using the new splitting of $I_{[1]}, \ldots, I_{[j+1]}$ into

$$((J_1, \emptyset), \ldots (J_{k-1}, \emptyset), (J_k, R_k), (J_{k+1}, Z_{k+1}), \ldots, \\ (J_j, Z_j), (I_{[j+1]} \cup Z_1 \ldots \cup Z_{k-1} \cup W_k, \emptyset)).$$

We will now consider the case where a replica is corrupted by a perturbation starting during the transmission of a replica. The perturbation might corrupt either the replica during which it occurred, with probability $\mathbb{P}_a$ under allocation x and $\mathbb{P}'_a$ under x', or the next replica (using assumption $\langle A_1 \rangle$) respectively with probability $\mathbb{P}_b$ or $\mathbb{P}'_b$. Since perturbation starting points are uniformly distributed over time and slots have the same size under all allocations, $\mathbb{P}_a = \mathbb{P}'_a$. The same proof based on the length of the intervals between replicas used for $\mathbb{P}_t$ shows that $\mathbb{P}_b \geq \mathbb{P}'_b$ since $|I(x)| \prec |I(x')|$.

The proof is concluded by noticing that $\mathbb{P}_{\text{one}}(x) = \mathbb{P}_t + \mathbb{P}_a + \mathbb{P}_b \geq \mathbb{P}_{\text{one}}(x') = \mathbb{P}'_t + \mathbb{P}'_a + \mathbb{P}'_b$. □

Theorem 4. *Under assumptions $\langle A_1 \rangle$ and $\langle A_2 \rangle$, for each FTU A, the optimal allocation x_{one} minimizing $\mathbb{P}_{\text{one}}$ is to group together all replicas of A.*

Proof: Under (A_1) and (A_2), each burst may corrupt a same replica independently. Therefore, $\mathbb{P}_{\text{one}}$ is a function of the probability that one burst corrupts one replica (denoted by q). By conditioning on the number of bursts, say K, one gets

$$\mathbb{P}_{\text{one}} = \sum_{i=0}^{K-1} q(1-q)^i = 1 - (1-q)^K.$$

This is an increasing function of q for all K. Therefore, minimizing q (i.e., minimizing the impact of one burst) also minimizes the combined effect of all bursts.

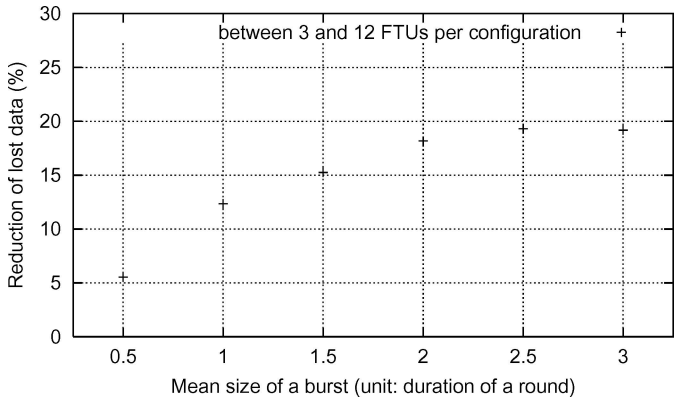

Figure 8. Reduction of the number of lost data when the optimal allocation is used instead of a random allocation. The data being lost when at least one replica of a same FTU is corrupted. The mean burst size ranges from 0.5 to 3 times the length of a round.

Finally, let x be an arbitrary allocation. The restrictions over one round R of x and x_{one} are denoted $x|_R$ and $x_{\text{one}}|_R$ respectively. They obviously satisfy $I(x|_R) \prec I(x_{\text{one}}|_R)$. By periodicity, one has $I(x) = (I(x|_R), I(x|_R), \ldots, I(x|_R))$ (repeated N times). This implies $I(x) \prec I(x_{\text{one}})$. Finally, applying Lemma 2 concludes the proof. □

5.3. Performance Evaluation

To assess the robustness improvement brought by the optimal allocation for $\mathbb{P}_{\text{one}}$, simulations were performed against random allocations. A configuration is defined by a number of FTU and the cardinality of each FTU. In these experiments, the number of FTUs ranges from 3 to 12. Two hundreds configurations were randomly generated with FTUs having a cardinality between 2 and 4. For each configuration, we randomly pick up 500 hundred slots (in the 2000 first rounds) where a data is transmitted for the first time. The duration of the production cycle of the data is chosen equal to one round which length is R. Then for each selected start of transmission, 10000 bursts of errors are generated with a size exponentially distributed of mean $c \cdot R$ with $c \in \{0.5,\ 1,\ 1.5,\ 2,\ 2.5,\ 3\}$. The starting point of each burst is randomly chosen in the first 2000 rounds. The event that has to be avoided is the corruption of one or more frames of the FTU by a perturbation. The results of these experiments are shown on Figure 8.

One observes that the clustering of the replica significantly diminishes the total number of lost data (around 18.5% for $c \in \{2,\ 2.5,\ 3\}$) knowing that there are cases where the start of the burst and its size are such that at least one replica will be corrupted whatever the allocation. The loss of robustness with a random allocation tends to be more important when the size of the burst is becoming bigger.

6. Concluding Remarks

This study shows that for TDMA-based systems with bursty perturbations choosing the position of the replicas inside the round has a very important impact on the efficiency of the replication.

The first result of this study is to give an optimal way to spread the replicas in order to minimize the probability to lose all replicas. This result is valid for most of the needs (see Section 3.2.1). For the other cases, we provide a low-complexity heuristic which proves to be very efficient on the simulations that were performed.

In a second part, it was shown that clustering together all replicas minimizes the probability to lose one or more replicas under a more general bursty perturbation model (the length of the bursts are not necessarily exponentially distributed). This result holds when the production cycle of a data is equal to the length of a TTP/C round. A first extension of this study is to consider arbitrary data production cycles but, then, the objective would be to receive at least one frame per station that belongs to the FTU during one production cycle.

As suggested by a reviewer, another interesting objective would be to maximize the expected number of replicas that are successfully received in order to maximize the confidence in the information. In a future work, one may also consider the case where a subset of FTUs requires the minimization of the loss probability while the rest of the FTUs need to minimize the probability that at least one replica is lost. This may be a situation arising on systems made of fail-silent and non fail-silent nodes. Another future work is to consider the use of Forward Error Correction techniques (such as Reed-Salomon codes) instead of replicas in order to make the system even more robust to transmission errors. Finally, we intend to study the robustness against transmission errors of an hybrid event-triggered/time-triggered network such as FlexRay which is also considered for use in X-by-Wire automotive applications.

Appendix A: Proof of Lemma 1

Let us consider an arbitrary allocation x for A. We look at the probability that an error corrupts all replicas carrying a given message m for allocation x. The same message (m) is emitted by a number of replicas which can be written as NK where N is an integer, and $K := C_A$ is the number of replicas per round. For notation simplicity, we also set $h := h_A$.

In the following, one denotes by C the round where message m begins. One also denotes by P_k the position of the last bit of the k-th replica for the FTU A in x and by d_k^i the "distance" between replica k and replica $k+i$: $d_k^i = P_{k+i} - P_k$.

One denotes by $\mathbb{P}_{\text{all}}(x)$ the loss probability of m under allocation x; by $\mathbb{P}_0(x)$ the loss probability under allocation x given that the perturbation starts in a round preceding round C and by $\mathbb{P}_1(x)$ the loss probability under allocation x given that the perturbation starts in the same round (C) . When the EMI burst covers the whole message, there is a relation between the random variables corresponding to the beginning of the message m (called B) and the beginning of the error burst (called S) respectively. Basically, the error must start before the end of the first replica carrying message m. See Figures 9 and 10 for an illustration of cases $\mathbb{P}_0$ and $\mathbb{P}_1$ respectively.

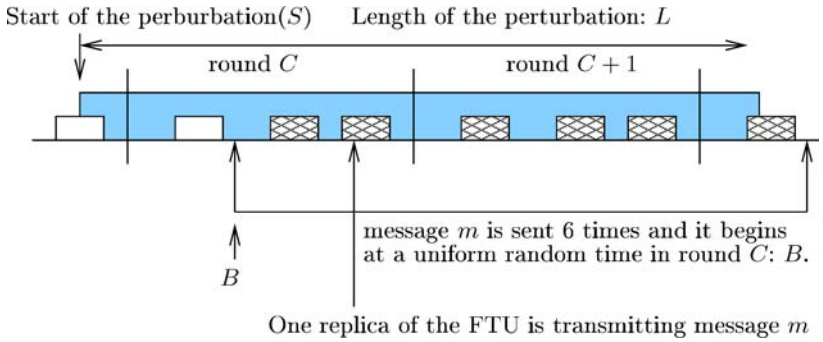

Figure 9. A perturbation burst which begins in a round preceding the start of a message covers the whole message (case $\mathbb{P}_0$).

By conditioning over the values of B and L (which are independent variables), we obtain:

$$\begin{aligned}\mathbb{P}_1(x) &= \sum_{k=CK+1}^{CK+K} \Big[(1-(1-\pi)^h)^{NK} \Pr(P_{k-1} - h < B < P_k - h) \Pr(CR \le S \le P_k) \\ &\quad \times \Pr\big(L + S \ge RN - d_{k-1}^1 - h + P_k\big)\Big] \\ &= \frac{(1-(1-\pi)^h)^{NK}}{R^2} \sum_{k=CK+1}^{CK+K} d_{k-1}^1 \int_{CR}^{P_k} \exp\big(\lambda\big(-RN + d_{k-1}^1 + h - P_k + S\big)\big) dS \\ &= \frac{(1-(1-\pi)^h)^{NK}}{\lambda R^2} \sum_{k=1}^{K} d_{k-1}^1 \exp\big(\lambda\big(-RN + d_{k-1}^1 + h\big)\big)(1 - \exp(-\lambda P_k)),\end{aligned}$$

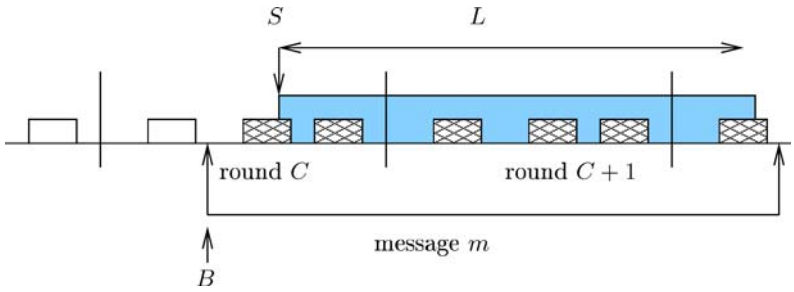

Figure 10. A perturbation burst, beginning in the same round as the message, covers the whole message (case $\mathbb{P}_1$).

and

$$\begin{aligned}
\mathbb{P}_0(x) &= \sum_{k=CK+1}^{CK+K} (1-(1-\pi)^h)^{NK} \Pr(P_{k-1}-h < B < P_k - h) \Pr\big(L+S \\
&\quad \geq RN + P_k - d^1_{k-1} - h\big) \\
&= \frac{(1-(1-\pi)^h)^{NK}}{R} \sum_{k=CK+1}^{CK+K} d^1_{k-1} \Pr(L+S \geq RN + P_k - d^1_{k-1}) \\
&= \frac{(1-(1-\pi)^h)^{NK}}{R} \sum_{k=CK+1}^{CK+K} d^1_{k-1} \int_0^{CR} \exp\big(\lambda\big(-RN - P_k + S + d^1_{k-1}\big)\big) dS/CR \\
&= \frac{\pi^{NK}(1-(1-\pi)^h)^{NK}}{\lambda C R^2} \sum_{k=1}^{K} d^1_{k-1} \exp\big(-\lambda RN - \lambda P_k + \lambda d^1_{k-1}\big)(1-\exp(-\lambda CR)).
\end{aligned}$$

Finally,

$$\begin{aligned}
\mathbb{P}_{\text{all}}(x) &= 1/(C+1)\mathbb{P}_1(x) + C/(C+1)\mathbb{P}_0(x) \\
&= M \sum_{k=1}^{K} d^1_{k-1}((1-\exp(-\lambda P_k)) + \exp(-\lambda P_k)(1-\exp(-\lambda CR))) \exp\big(\lambda d^1_{k-1}\big) \\
&= M \sum_{k=1}^{K} d^1_{k-1}(1-\exp(-\lambda P_k - \lambda CR)) \exp\big(\lambda d^1_{k-1}\big),
\end{aligned}$$

where

$$M = \frac{\exp(-\lambda RN + \lambda h)(1-(1-\pi)^h)^{NK}}{(C+1)\lambda R^2}.$$

We consider shifts to the left of the normalization $P(x) := \mathbb{P}_{\text{all}}(x)/M$.

$$\begin{aligned}
P(x+\delta_i) = &\sum_{k=1, k\notin\{a,a+1\}}^{K} d^1_{k-1}(1-\exp(-\lambda P_k - \lambda CR)) \exp\big(\lambda d^1_{k-1}\big) \\
&+ \big(d^1_{a-1}-1\big)(1-\exp(-\lambda P_a + \lambda - \lambda CR)) \exp\big(\lambda d^1_{a-1} - \lambda\big) \\
&+ \big(d^1_a + 1\big)(1-\exp(-\lambda P_{a+1} - \lambda CR)) \exp\big(\lambda d^1_a + \lambda\big).
\end{aligned}$$

We need to distinguish the cases where $|b-a| = 1$. We focus on the case where $b = a+1$ (the case $a = b+1$ is symmetrical by exchanging the roles of a and b). If $b > a+1$,

$$\begin{aligned}
P(x+\delta_i+\delta_j) = &\sum_{k=1, k\notin\{a,a+1,b,b+1\}}^{K} d^1_{k-1}(1-\exp(-\lambda P_k - \lambda CR)) \exp\big(\lambda d^1_{k-1}\big) \\
&+ \big(d^1_{a-1}-1\big)(1-\exp(-\lambda P_a + \lambda - \lambda CR)) \exp\big(\lambda d^1_{a-1} - \lambda\big)
\end{aligned}$$

$$
\begin{aligned}
&+ \left(d_a^1 + 1\right)(1 - \exp(-\lambda P_{a+1} - \lambda CR)) \exp\left(\lambda d_a^1 + \lambda\right) \\
&+ \left(d_{b-1}^1 - 1\right)(1 - \exp(-\lambda P_b + \lambda - \lambda CR)) \exp\left(\lambda d_{b-1}^1 - \lambda\right) \\
&+ \left(d_b^1 + 1\right)(1 - \exp(-\lambda P_{b+1} - \lambda CR)) \exp\left(\lambda d_b^1 + \lambda\right).
\end{aligned}
$$

If $b = a + 1$, on the other hand, we get

$$
\begin{aligned}
P(x + \delta_i + \delta_j) = &\sum_{k=1, k \notin \{a, a+1, a+2\}}^{K} d_{k-1}^1 (1 - \exp(-\lambda P_k - \lambda CR))) \exp\left(\lambda d_{k-1}^1\right) \\
&+ \left(d_{a-1}^1 - 1\right)(1 - \exp(-\lambda P_a + \lambda - \lambda CR)) \exp\left(\lambda d_{a-1}^1 - \lambda\right) \\
&+ \left(d_a^1\right)(1 - \exp(-\lambda P_{a+1} + \lambda - \lambda CR)) \exp\left(\lambda d_a^1\right) \\
&+ \left(d_{a+1}^1 + 1\right)(1 - \exp(-\lambda P_{a+2} - \lambda CR)) \exp\left(\lambda d_{a-1}^1 + \lambda\right).
\end{aligned}
$$

If we compute $Q := P(x + \delta_i) + P(x + \delta_j) - P(x + \delta_i + \delta_j) - P(x)$, we get 0 when $b > a + 1$ and when $b = a + 1$, we get

$$
\begin{aligned}
Q = &\left(d_a^1 + 1\right)(1 - \exp(-\lambda P_{a+1} - \lambda CR)) \exp\left(\lambda d_a^1 + \lambda\right) \\
&- \left(d_a^1\right)(1 - \exp(-\lambda P_{a+1} + \lambda - \lambda CR)) \exp\left(\lambda d_a^1\right) \\
&+ \left(d_a^1 - 1\right)(1 - \exp(-\lambda P_{a+1} + \lambda - \lambda CR)) \exp\left(\lambda d_a^1 - \lambda\right) \\
&- d_a^1 (1 - \exp(-\lambda P_{a+1} - \lambda CR)) \exp\left(\lambda d_a^1\right).
\end{aligned}
$$

After some simplifications, we obtain

$$
\begin{aligned}
Q = &\exp\left(\lambda\left(d_a^1 + 1\right)\right) d_a^1 + d_a^1 \exp\left(\lambda\left(d_a^1 - 1\right)\right) - 2 d_a^1 \exp\left(\lambda d_a^1\right) \\
&+ \exp\left(\lambda\left(d_a^1 + 1\right)\right) - \exp\left(\lambda\left(d_a^1 - 1\right)\right) \\
&+ \exp\left(\lambda\left(d_a^1 - P_{a+1} - RC\right)\right) - \exp\left(\lambda\left(d_a^1 - P_{a+1} - RC + 1\right)\right) \\
\geq\ &0.
\end{aligned}
$$

The first line is non-negative by convexity of the function $z \mapsto \exp(\lambda z)$. The sum of the second and third lines is also non-negative by convexity of the function $z \mapsto \exp(\lambda z)$.

Acknowledgments

The authors would like to thank the anonymous referees for their helpful comments and suggestions. They would also like to thank Dinard van der Laan (`dalaan@feweb.vu.nl`) for pointing out some results used in Section 3.2.1.

References

Altman, E., Gaujal, B., and Hordijk, A. 2000a. Admission control in stochastic event graphs. *IEEE Transaction on Automatic Control* 45(5): 854–868.

Altman, E., Gaujal, B., and Hordijk, A. 2000b. Multimodularity, convexity and optimization properties. *Mathematics of Operations Research* 25(2): 324–347.

Altman, E., Gaujal, B., and Hordijk, A. 2003. *Discrete-Event Control of Stochastic Networks: Multimodularity and Regularity*, No. 1829 in Lecture Notes in Mathematics. Springer Verlag.

Barrenscheen, J. and Otte, G. 1997. Analysis of the physical CAN bus layer. In *4th international CAN Conference, ICC'97*. pp. 06.02–06.08.

Brasileiro, F., Ezhilchelvan, P., Shrivastava, S., Speirs, N., and Tao, S. 1996. Implementing fail-silent nodes for distributed systems. *IEEE Transactions on Computers* 45(11): 1226–1238.

Dilger, E., Führer, T., Müller, B., and Poledna, S. 1998. The X-by-wire concept: Time-triggered information exchange and fail silence support by new system services. Technical Report 7/1998. Technische Universität Wien, Institut für Technische Informatik. also available as SAE Technical Paper 98055.

Gaujal, B. and Navet, N. 1999. Traffic shaping in real-time distributed systems: a low-complexity approach. *Computer Communications* 22(17): 1562–1573.

Grünsteidl, G., Kantz, H. and Kopetz, H. 1991. Communication reliability in distributed real-time systems. In *10th Workshop on Distributed Computer Control Systems*.

Hajek, B. 1985. Extremal splittings of point processes. *Mathematics of Operation Research* 10(4): 543–556.

ISO, I. 1994. *Road Vehicles—Low Speed serial data communication—Part 2: Low Speed Controller Area Network*. ISO. ISO 11519–2.

Kopetz, H. 1997. *Real-Time Sytems: Design Principles for Distributed Embedded Applications*. Boston: Kluwer Academic Publishers.

Kopetz, H., Bauer, G. and Poledna, S. 2001. Tolerating arbitrary node failures in the time-triggered architecture. In *SAE 2001 World Congress*, March 2001, Detroit, MI, USA.

Kopetz, H., Nossal, R., Hexel, R., Krüger, A., Millinger, D., Pallierer, R., Temple, C., and Krug, M. 1998. Mode handling in the time-triggered architecture. *Control Engineering Practice* 6: 61–66.

Marshall, A. W. and Olkin, I. 1979. *Inequalities: Theory of Majorization and its Applications*, Vol. 143 of *Mathematics in Science and Engineering*. Academic Press.

Navet, N., Song, Y.-Q., and Simonot, F. 2000. Worst-case deadline failure probability in real-time applications distributed over CAN (Controller Area Network). *Journal of Systems Architecture* 46(7): 607–618.

Noble, I. 1992. EMC and the automotive industry. *Electronics & Communication Engineering Journal* 263–271.

Pfeifer, H. 2000. Formal verification of the TTP group membership algorithm. In *FORTE/PSTV 2000*.

Poledna, S. 1996. *Fault-Tolerant Real-Time Systems: The Problem of Replica Determinism*. Kluwer Academic Publishers.

Poledna, S., Barrett, P., Burns, A., and Wellings, A. 2000. Replica determinism and flexible scheduling in hard real-time dependable systems. *IEEE Transactions on Computers* 49(2): 100–111.

Rushby, J. 2003. A comparison of bus architecture for safety-critical embedded systems. Technical report, NASA/CR.

Temple, C. 1998. Avoiding the babbling-idiot failure in a time-triggered communication system. In *International Symposium on Fault-Tolerant Computing (FTCS)*, pp. 218–227.

TTTech Computertechnik GmbH 2003. Time-triggered protocol TTP/C, high-level specification document, Protocol Version 1.1.

Wilwert, C., Navet, N., Song, Y.-Q., and Simonot-Lion, F. 2004. Design of automotive X-by-wire systems. In R. Zurawski (ed.): *The Industrial Communication Technology Handbook*. CRC Press.

X-by-Wire Consortium 1998. X-by-wire – Safety related fault tolerant systems in vehicles – final report. Project BE95/1329, Contract BRPR-CT95-0032.

Zanoni, E. and Pavan P. 1993. Improving the reliability and safety of automotive electronics. *IEEE Micro* 13(1): 30–48.

Bruno Gaujal is a research director at INRIA Rhone-Alpes since 2003 where he is the leader of the group on large scale networks. He has held several positions in INRIA Sophia-Antipolis, Loria and ENS-Lyon before. His main interest are performance evaluation and control of discrete event dynamic systems.

Nicolas Navet received the M.S. in Computer Science from the University of Berlin (Germany) in 1994 and the PhD in Computer Science from the University of Nancy in 1999. Before joining the INRIA (LORIA Lab.) in November 2000, he was research scientist at Gemplus Software. His research interests include scheduling theory, the design of communication protocols for real-time and fault-tolerant data transmission and probabilistic risk evaluation when transient faults may occur (for instance, due to electromagnetic interferences). More information on his work can be found at url http://www.loria.fr/∼nnavet.