Optimal replica allocation for TTP/C based systems

Bruno Gaujal, Nicolas Navet

OPTIMAL REPLICA ALLOCATION FOR TTP/C BASED SYSTEMS [1]

Bruno Gaujal [1] **Nicolas Navet** [2]

[1] ENS Lyon - LIP
46, Allée d'Italie
69007 Lyon - France
bgaujal@ens-lyon.fr

[2] LORIA - INPL
ENSEM - 2, Avenue de la forêt de Haye
54516 Vandoeuvre-lès-Nancy - France
nnavet@loria.fr

Abstract: In this study we show how one can use Fault-Tolerant Units (FTU) in an optimal way to make a TTP/C network robust to bursty random perturbations. We consider two possible objectives corresponding to well defined situations of the field of fault-tolerance. 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. 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.

Keywords: Real-Time Systems, Vehicles, Fault Tolerance, Noisy Channels, Network Reliability.

1. INTRODUCTION

Multi-access protocols based on TDMA (Time Division Multiple Access) 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 station failures. There exists several variants of the TDMA scheme. In this paper, we study the TTP/C protocol (Time Triggered Protocol - see TTTech Computertechnik GmbH (2002)) which implements the synchronous TDMA scheme: the stations (or nodes) have access to the bus in a strict deterministic sequential order and 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 TTP/C is currently considered in high-dependability real-time applications where fault tolerance and guaranteed response times has to be provided. Examples of such applications are "brake-by-wire" and "steer-by-wire" in-vehicle applications (see Dilger et al. (1998)) 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 of the FTU may still be correct while others are corrupted). They also provide a way to fight against *measurement and computation errors* occurring before the transmission (some nodes 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 repre-

[1] An extended version of this paper is available as INRIA Research Report RR-4614.

sent a serious threat to the correct behavior of the system. For instance, in automotive applications, the EMI (see Noble (1992); Zanoni and Pavan (1993)) can either be radiated by some in-vehicle electrical devices (switches, 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 because they are identical and very close one to each other. Unlike CAN (Controller Area Network - ISO (1994)), TTP/C do not provide automatic retransmission for corrupted frames and their data are actually lost for the application.

Goal of the paper. The system under study is an application with redundant nodes distributed over a TTP/C network. The problem we address here 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 transmission errors. We consider two distinct objectives :

(1) **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 $\mathbb{P}_{all}$.
(2) **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. 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 $\mathbb{P}_{one}$.

As it will be further discussed in Subsection 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".

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 the slot of each station 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 as proposed in Navet et al. (2000) with the following assumptions:

(A_1) 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.
(A_2) the starting times of the EMI bursts are independent random variables uniformly distributed over time.
(A_3) the distribution of the size of the bursts is arbitrary provided that it is independent of the starting point of the burst.
(A_4) if a perturbation overlaps a whole slot, then we assume that the probability that the frames remains uncorrupted is neglectable (with $\pi = 0.5$ and a 100 bits frame, this probability is about 10^{-30}).

Without further knowledge on the considered application and its environment, assumptions (A_2), (A_3) and (A_4) are rather reasonable.

Related work. The TTP/C protocol, which is defined in TTTech Computertechnik GmbH (2002), is a central part of the Time-Triggered Architecture (TTA - see Kopetz (1997); Kopetz et al. (2001)) and it possesses numerous features and services related to dependability such as the bus guardian (see Temple (1998)), the group membership algorithm (see Pfeifer (2000)) and support for mode changes (see 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/C network is studied with the taking into account of transmission errors on the bus as well as failures in the TTP/C 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 influence on the transmission error probabilities.

The second difference with Grünsteidl et al. (1991) is that we do not compute the reliability of a given system but 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.

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. We then justify the two distinct objectives that were identified with regard to the concept of "fail-silence". Finally, we describe the TTP error handling mechanisms that have to be taken into account.

2.1 MAC Protocol description

Throughout this paper, we will considered the synchronous TDMA protocol. 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.

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*

2.2 Application model

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).

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 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 follow 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. In mathematical terms these compatibility constraints can be written

$$\sum_{A \in \mathcal{F}} x^A = (1, \cdots, 1).$$

Finally, the frame sent by a node contains some data whose value is periodically updated as it is generally the case in distributed control applications. For instance, in a typical car environment, a frame sent by the engine controller may contain the RPM value plus the engine temperature and a new frame is built every 10ms.

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. The value of R is thus not generally correlated with the production cycle of the data [2]. 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 and if FTU A has more than one replica, 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 1.

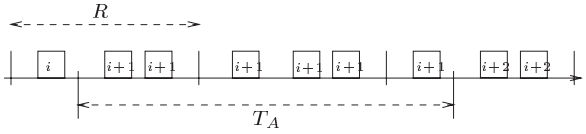

Fig. 1. Three successive rounds. Only the slots allocated to the FTU A of cardinality 3 are shown. The message corresponding to the $(i+1)^{\text{th}}$ production cycle is sent over 3 rounds.

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 behaviour of the individual components (see Dilger et al. (1998)). For instance, if the failure 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 (eg. 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 Dilger et al. (1998); Temple (1998); 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. In this case, the main objective to achieve with regard to the robustness against transmission errors is the minimizing of $\mathbb{P}_{all}$, that is the probability that all frames of the FTU (carrying data corresponding to the same production cycle) will be corrupted.

In practice replicated sensors may return slightly different observations and, without extra communication for an agreement, replicated nodes of a same FTU may transmit different data. If a decision, such as a majority vote, is taken by a consumer node with regard to the value of the transmitted data, the objective is to maximize the probability that at least one frame of each FTU is successfully transmitted. If the production cycle is equal to one round then it comes to minimizing $\mathbb{P}_{one}$, the probability that one or more frames of an FTU become corrupted.

2.4 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 (2002) page 101). Three are of particular importance in our context :

- the "active" state which is the normal functioning state,
- the "passive" state : the controller is synchronized and can receive frames but no transmission is allowed,
- 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 :

[2] The version 1.0 of the TTP/C specification TTTech Computertechnik GmbH (2002) enables the designer to insert an idle time after the transmission of a frame so that the duration of a round can take an application related value. In particular it could be equal to the length of the production cycle of a data but the problem remains with data having different production cycles.

(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".

3. MINIMIZING $\mathbb{P}_{ALL}$

In this section, we investigate the problem of minimizing the loss probability $\mathbb{P}_{all}$ on TTP/C. The problem has been studied in Gaujal and Navet (2002) for the general synchronous TDMA case. The TTP/C rules 1,2 and 3 actually affect the value of $\mathbb{P}_{all}$ but not which allocation scheme is optimal. However, the majority rule of TTP/C (item 4 above) 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.

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 2.

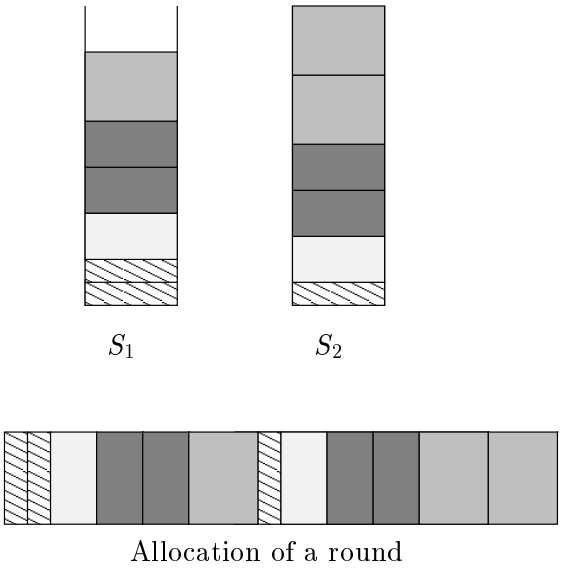

Fig. 2. Construction of the optimal allocation x_{stack}.

Theorem 1. Using TTP/C and under the foregoing assumptions, the allocation x_{stack} minimizes $\mathbb{P}_{all}$.

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 (A_2)), 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 network. It is thus useless to consider a distance between replicas larger than $\lfloor S/2 \rfloor$. This means that x_{stack} is optimal. □

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.

4. MINIMIZING $\mathbb{P}_{ONE}$

Unlike the previous case, the technique used to find the optimal allocation of the replicas of one FTU is based on majorization and Schur convexity.

4.1 Schur convexity and majorization

Let $u = (u_1, \cdots, u_n)$ and $v = (v_1, \cdots, v_n)$ be two real vectors of size n. We denote by $(u_{[1]} \cdots, u_{[n]})$ and $(v_{[1]}, \cdots, v_{[n]})$ 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:

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

$$\sum_{i=1}^{k} u_{[i]} \leq \sum_{i=1}^{k} v_{[i]} \quad k \leq n. \tag{2}$$

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).

4.2 Schur concavity of $\mathbb{P}_{one}$

In this section, we will show that the probability that an error burst corrupts at least one replica within a production cycle ($\mathbb{P}_{one}$) 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 $\mathbb{P}_{one}$. Note that the result will be proven for arbitrary production cycles although, in our context, $\mathbb{P}_{one}$ 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 $(I_1, \cdots I_t)$ and by $|I(x)|$ the vector of the length of the intervals, $|I(x)| = (|I_1|, \cdots, |I_t|)$. Note that $|I_1(x)| + \cdots + |I_t(x)| = N(R - Kh)$ does not depend on the allocation x.

Lemma 1. 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 $\mathbb{P}_{one}(x) \geq \mathbb{P}_{one}(x')$.

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 $I_{[1]}, \cdots, I_{[j]}$ and $I'_{[1]}, \cdots, I'_{[j]}$ such that the probability $\mathbb{P}'_j$ that an error starting in $I'_{[1]}, \cdots, I'_{[j]}$ and corrupting at least one replica is smaller than the corresponding probability $\mathbb{P}_j$ in $I_{[1]}, \cdots, I_{[j]}$. For $j = 1$, the coupling is done according to Figure 3.

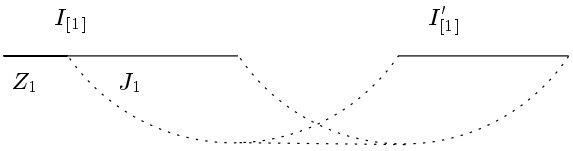

Fig. 3. Coupling for the smallest interval.

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,
- if a perturbation overlaps the whole replica then the corruption occurs with probability 1 (assumption (A_4)) 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 $I_{[1]}, \cdots, I_{[j]}$ into $(J_1, Z_1), \cdots, (J_j, Z_j)$ 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 3. 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.

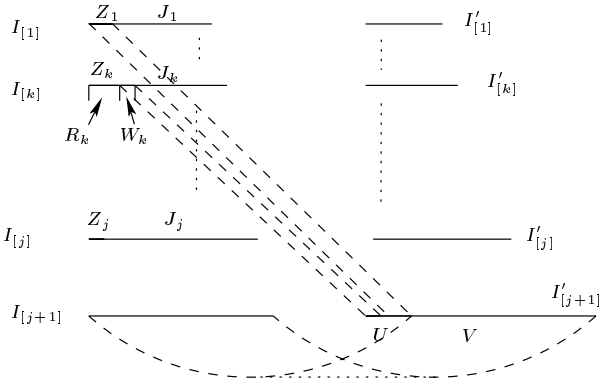

Fig. 4. Coupling when $I_{[j+1]} \leq I'_{[j+1]}$.

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 4.

- 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 $Z_1 \cup \cdots Z_{k-1} \cup W_k$ 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, $\mathbb{P}_{j+1} \geq \mathbb{P}'_{j+1}$.

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

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

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 (A_4)) 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 $\mathbb{P}_{one}(x) = \mathbb{P}_t + \mathbb{P}_a + \mathbb{P}_b \geq \mathbb{P}_{one}(x') = \mathbb{P}'_t + \mathbb{P}'_a + \mathbb{P}'_b$.

Theorem 2. Under assumptions (A_1), (A_2) and (A_4) for each FTU A, the optimal allocation x_{one} minimizing $\mathbb{P}_{one}$ is to group together all replicas of A.

PROOF. Under (A_1), (A_2) and (A_4) each burst may corrupt a same replica independently. Therefore, $\mathbb{P}_{one}$ 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

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

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.

Finally, let x be an arbitrary allocation. The restrictions over one round R of x and x_{one} are denoted $x|_R$ and $x_{one}|_R$ respectively. They obviously satisfy $I(x|_R) \prec I(x_{one}|_R)$. By periodicity $I(x) = (I(x|_R), I(x|_R), \cdots, I(x|_R))$ (repeated N times). This implies $I(x) \prec I(x_{one})$. Finally, applying Lemma 1 concludes the proof. □

The combined minimization of $\mathbb{P}_{one}$ for all FTUs is not a problem since the optimal solution is to group all replicas of each FTU together.

4.2.1. Performance comparison against random allocations To assess the robustness improvement brought by the optimal allocation for $\mathbb{P}_{one}$, 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 5. 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

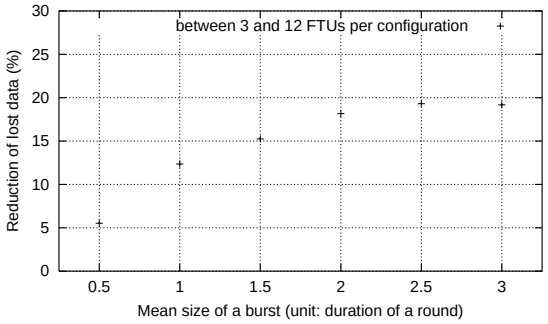

Fig. 5. Reduction of the number of lost data when the optimal allocation is used instead of a random allocation. The data being lost when at least one replica of a same FTU is corrupted. The mean burst size ranges from 0.5 to 3 times the length of a round.

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.

5. CONCLUDING REMARKS

The position of the replicas inside a TTP/C round has an impact on the robustness of the system to transmission errors when bursts of errors may occur. The first result of this study is to give an optimal way to spread the replicas in order to minimize the loss probability of all replicas. In a second part, it has been proven that clustering together all replicas minimizes the probability to lose one or more replicas when the production cycle of a data is equal to the length TTP/C round. A first extension of this study is to consider arbitrary data production cycles.

In a future work, one may consider the case where a subset of FTUs requires the minimization of the loss probability while the rest of the FTUs need to maximize the probability that at least one replica of each FTU is successfully transmitted. 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.

REFERENCES

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

E. Dilger, T. Führer, B. Müller, and S. Poledna. 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, 1998. also available as SAE Technical Paper 98055.

B. Gaujal and N. Navet. Maximizing the robustness of tdma networks with applications to TTP/C. Technical Report RR-4614, INRIA, 2002.

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

International Standard Organization ISO. *Road Vehicles - Low Speed serial data communication - Part 2: Low Speed Controller Area Network.* ISO, 1994. ISO 11519-2.

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

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

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

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

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

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

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

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

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

TTTech Computertechnik GmbH. *Specification of the TTP/C Protocol - version 1.0*, July 2002.

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