

# Safety Evaluation of Controlled System distributed on TTA Architecture

Jumel Fabrice, Godary Karen and Augé-Blum Isabelle

CITI, INSA de Lyon, Bât L. De Vinci, 21 av Jean Capelle, 69621 Villeurbanne, France

fabrice.jumel,karen.godary,isabelle.auge-blum@insa-lyon.fr

## Abstract :

This paper presents a method to quantify the safety of a critical function in the automotive domain. With the arrival of “*X-by-wire*”, these functions will be made of mechatronic systems (composed by sensors, actuators and calculators), distributed over a communication network. Our study shows the feasibility of the link between the network's behavior (in presence of faults like lost of samples) and the implanted function, and allows to evaluate the probability of a wrong car's behavior.

We illustrate our method with a brake function, implanted over Time Triggered Architecture (TTA) and we quantify, with an initial speed for the car, the probability to stop it in a given duration.

**Keywords :** fault-tolerant, safety modeling, vehicle control system, distributed architecture,

## I. INTRODUCTION

Electronic aspects take a more and more important place in the automotive area. In the future, all the critical functions of the car should be entirely electronically (for instance the brake, direction and suspension functions). To allow this evolution, the developed solution, “*X-by-wire*” architecture, is based on communication network like the Time-Triggered Architecture (TTA) studied here [Ko98]. The inevitable presence of faults on the communication's medium (for instance, due to magnetic perturbations) can corrupt the data exchange. An important problem is to prove the safety of these critical functions mapped on such architecture. It means that replacing the different mechanical parts of the transmission of information with electronic distributed solutions does not lead to an important risk of service's degradation.

To prove this, we need a formalism to model the critical function and to determine how errors on data used by control application can appear for a given architecture. An important characteristic of these faults are to be non-permanent. Therefore, they cannot be modeled with usual representation as AMDEC or HAZOP ([Mc&all95]). In order to take into account this type of transitory faults, different types of models have been proposed ([ShRaGa96] and [GaWiAs03]). Some important studies give a basic way to represent their apparition and their possible propagation

on a complex architecture (for example a distributed one [Cr&all02]).

In a previous studies [JuNaSi03, JuThAu03], we have proposed to integrate this type of model of faults with an accurate model of the critical function. A hybrid representation has been proposed, which take into account the specificity of the controlled system and of the electronic architecture. Combining with the useful formalism of the Markov's chains, it allows to calculate some important safety properties of controlled system. In this paper we use this methodology for the case of automotive application distributed on a communication network. Therefore, we have to take into account the specificities of the communication protocols, particularly in terms of fault-tolerance mechanisms.

The first section presents the context of the study, the second presents a model of the behavior of TTA in presence of faults, which allows to present in the last section an effective way to compute some safety properties for a vehicle's critical function in presence of faults.

## II. CONTEXT OF THE STUDY

The critical functions are typically mechatronic systems composed by sensors, actuators and the control law (algorithm implanted on a calculator). The entire system includes also the physical behavior of the car whose characteristics involve in the response of the function (mass, speed ...).

The quality of this type of functions is generally defined on the physical behavior of

the car. It is for example, “the distance before stop” for a brake function and “the ability to fit a trajectory” for the steer or the suspension functions.

A general solution to model this type of system is to split the system in two parts. The first one models all the continuous views of the system (in particular physical dynamics : hydraulic, mechanical and electrical aspects). The second one models the discrete part of the system, more precisely the algorithms, which control the system. The more general formalism is the state representation. A system is defined by different values, which defined the state of the system (internal or output).

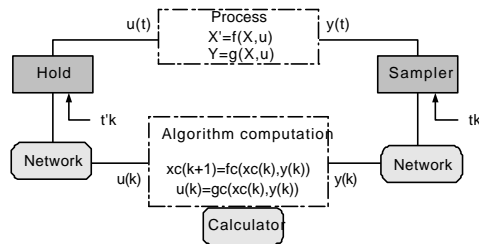


Figure 1 : States representation of a sampled control

Figure 1 represents such a decomposition. The physical part is defined by the state  $X$ . The dynamic of the system corresponds to the different differential equations (due to electrical or mechanical laws for examples). The output of the process  $y(t)$  is sampled at each  $t_k$  ( $y(k)$ ). This sample is used by the algorithm of the control law in order to produce the control signal  $u(k)$  (which is hold in the different actuators and become a continuous input for the physical process ( $u(t)$ )).

The algorithm corresponds to a discrete system, which produces control samples with the data samples.

The critical functions are mapped on a distributed architecture based on a communication medium. Faults on this medium can lead to errors at the application level. Different types of errors can appear on the data samples used by critical functions. These errors can be bad values, delays or eventually a lost of the data samples depending on the chosen network protocol.. In the case of TTA, as presented in the following section, the lost of data is the more important and is the only one error discussed in this paper.

### III. TTA’S BEHAVIOR IN PRESENCE OF FAULTS

### GENERAL PRESENTATION

TTA is a time triggered architecture composed of a set of nodes connected through a communication network based on TTP/C (Time Triggered Protocol class C) protocol [KoGr94], [TTP02].

A node is composed of three independent levels (figure 2) :

- The application level, which executes the control algorithms,
- The FTCom layer, compatible with FTCom (Fault Tolerant Communication) standard [FTC01]. This level is in charge of the redundancy, which is the base of fault-tolerance problems.
- And the TTP/C protocol controller.

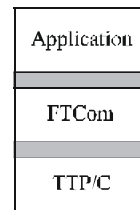


Figure 2: The different Layers of a TTA node

The TTP/C protocol implements a broadcast communication using the TDMA (Time Division Multiple Access) strategy. It is based on a static and predefined scheduling stored in a static table. All the nodes have the global knowledge of the whole behavior of the system : global time, emission and reception instants, state of the other nodes.

A slot is the smallest unit of the communication phase : each node is assigned to a slot in which it can send its message. The slots are grouped together in a TDMA round. The usual system phase is a cyclic execution of all TDMA rounds.

Fault tolerance in TTA is carried out with some detection mechanisms [Rus01], (for instance local clocks synchronization [Ko&all97] or membership service [BaPa00]) and with a hardware redundancy: the architecture is based on two buses, and the management of this redundancy is included in TTP/C. Other possible redundancy can be done by the FTCom layer.

### CONSEQUENCE OF A FAULT

Faults on the communication medium lead to faults on the messages transmissions. In TTA, a fault on a message is detected. The static scheduling of messages implies that there is no possible delays on the transmissions and just losses are possible.

From an application point of view, a data is lost in two cases:

1) The data has been corrupted by a fault during its transmission. With the mechanisms of detection, this message is not acknowledge and consider as lost for the application. The sending node would be see as faulty and would have to reintegrate.

2) A node is in passive state and no messages can be send before to be reintegrated. This state is implied by the fault tolerance mechanism associated to the detection of a previous fault.

During the necessary time to reintegrate, some data can be lost. We need to characterize the worst duration (bound), in order to model the number of possible losses linked with a initial transmission fault.

TTP/C services have been validated in the context of faults. Nevertheless, this validation is not enough here. We need to evaluate the temporal properties of their behaviors, particularly the worst execution case associated to a reintegration.

In previous work [GoAuMi04], we present a methodology to extract parametric temporal bounds on TTP/C services using a timed automata model of the protocol and the verification tool UPPAAL ([Am&all01]). In some hypothesis and fault model, the behavior of the protocol is verified and above all some temporal bounds could be extracted on the behavior of the different services. The only case considered here is the detection of a faulty node and its reintegration.

#### TEMPORAL BOUND EXTRACTION : REINTEGRATION SERVICE

**Detection and reintegration service:** After sending a message, the sender node try to acknowledge this message. If it fails (“membership loss”), the node detects that it is faulty and then transits to passive state. This mechanism is based on the membership service of TTP/C, and on the acknowledgment algorithm of the protocol [TTP02]. A node in passive state remains synchronized with other nodes and still received messages. To reintegrate, the node must receive 2 correct frames. Then, it can transits in active state and therefore would be able to send frames. For more details, please refer to the protocol specification [TTP02].

The time necessary for reintegration (including the detection of the fault leading to

it) depends on the type of faults. We present in the following the bounds in the hypothesis that there is at the maximum one fault during a TDMA round. The faults can be classified in different types according to their effects on the system [Rus01]. There are two types: the non-byzantine ones, which have the same effect on all the nodes, and the byzantine ones, for which the effects can be different on different nodes. These two cases are presented in scenarios 1 and 2.

**Scenario 1**, figure 3 : In the case of a simple emission fault, i.e. a non-Byzantine, the faulty node detects the membership loss in two slots. It then transits to the passive state and then can reintegrate after two slots. In the case there is 5 slots in the TDMA round the faulty node can reintegrate before its next sending slot. Thus, the bound is  $\Delta_{\text{round}}$  (duration of a round).

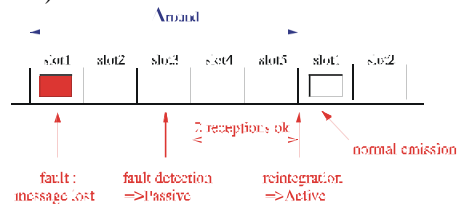


Figure 3 : Best case of reintegration

**Scenario 2**, figure 4: In case the fault is a Byzantine one, the faulty node can detects the membership loss in one round. It then transits to the passive state and waits for two correct receptions. In this case, the faulty node loses its next sending slot and has to wait for two rounds since the fault instant to send a new frame. The bound here is  $2 * \Delta_{\text{round}}$ .

This bound is the same if there is a non-byzantine fault during the reintegration phase.

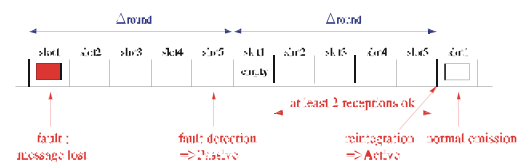


Figure 4 : Worst case of reintegration

#### IV EXAMPLE, STUDY OF BRAKING FUNCTION

The previous results can be used to evaluate the safety of critical function under network faults. We present the method on a brake function.

#### MODELING OF A BRAKE FUNCTION

We consider here a brake function on only one wheel. The output of the process (the braking car) is defined by the actual speed of the car.

During the braking, the speed is influenced by the slip friction

$$\text{slip} = 1 - \frac{w_w}{w_v}$$

where  $w_v$  is the vehicle speed in terms of wheel angular velocity and  $w_w$  is the real wheel angular velocity.

The process can be described as state representation as presented in introduction. The brake data input controls the brake torque, which is the input of the process.

An effective control is obtained for a relative slip of 0.2, which is known to minimize the time to brake.

A possible input data  $Y$  of the control law is the relative slip.

The control algorithm could be based on a direct control with a bang-bang control (simple example of the Anti Lock Braking system ABS) :

$$u(k) = g_c(y(t))$$

With  $g_c = \text{maxtorque}$  if  $y(t) < 0.2$  and  $g_c = \text{mintorque}$  if  $y(t) > 0.2$

We need to model the behavior of the brake function in presence of errors on the data. To simplify this behavior, we consider only one type of degradation: the vehicle's speed is decreased from  $\Delta_{\text{max}}$  in absence of error and  $\Delta_{\text{min}}$  in presence of error. The brake function under errors can therefore be represented as:

$$v_i = v_{i-1} - \epsilon_i \Delta_{\text{Max}} - (1 - \epsilon_i) \Delta_{\text{Min}}$$

where  $v_i$  is the vehicle's speed (internal state of the process) and where  $\epsilon_i = 0$  corresponds to an error and  $\epsilon_i = 1$  to none.

**SAFETY OF A BRAKE FUNCTION BASED ON TTA**

We consider a simple implantation of the brake control, with two nodes, one for the control law and one for the actuator (figure 6). The TDMA round is composed of 5 slots (one for the brake data, the others are for other applications).

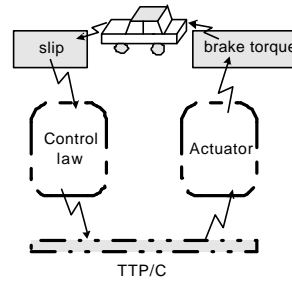


Figure 6 : Simple implantation with only two nodes

In order to present the methodology we have amplified the consequence of a fault by reducing considerably the maximal number of steps necessary for a brake (we consider about 10 steps, but in reality we would have to consider them in hundreds).

So, for the study, we set the parameters of the brake function under errors :

- The initial state corresponds to a speed of  $v_0 = 12$  Units.
- The final state is obtained when the vehicle is stopped ( $v_{13} = 0$ )
- $\Delta_{\text{max}} = 3$  and  $\Delta_{\text{min}} = 1$

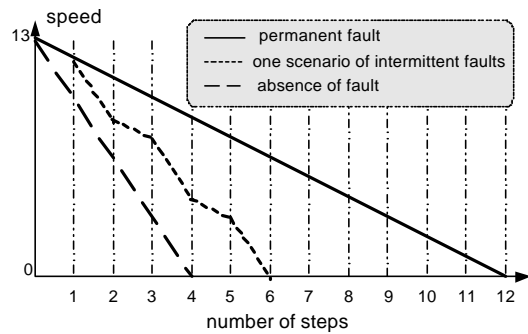


Figure 7 : Different behavior in presence of faults

Figure 7 shows the result for three different scenarios. One can see that the brake can take three times more steps before a complete stop in presence of errors. To study the whole behavior of the function, we need to take into account all the error scenarios. The presence of error is linked with the presence of fault on

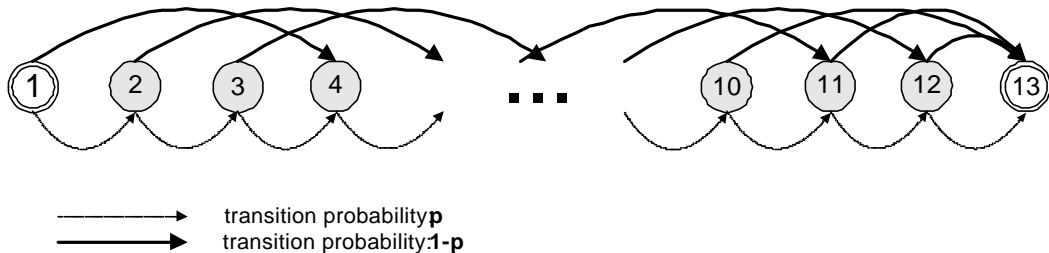


Figure 5 : Markov's Chain of a braking function : reintegration in one round



simplification used in this study, the probability obtained is not realistic for an existent vehicle. But this methodology is still applicable with more accurate models.

## VI. CONCLUSION

In this article, we have presented a methodology to study the safety of a mechatronic function distributed on a network. We have illustrated this methodology on a brake function mapped on the Time Triggered Architecture (TTA). Such a study needs accurate models of the physical systems and of the electronic devices and protocols, in presence of faults. The integration of these different models, thanks to Markov's Chains, allows computing some important properties of safety.

We have demonstrated here the feasibility of the methodology on a simple case. A future work is to apply this methodology to real industrial cases in order to prove the safety of future "X-by-wire" vehicles.

## REFERENCES

- [Am&all01] T. Amnell, G. Behrmann, J. Bengtsson, P. R. D'Argenio, A. David, A. Fehnker, T. Hune, B. Jeannot, Kim G. Larsen, M. O. Möller, P. Pettersson, C. Weise, and W. Yi, Uppaal - Now, Next, and Future, In Proceedings of Modelling and Verification of Parallel Processes (MOVEP'2k), Nantes, France, June 19 to 23, 2000. LNCS Tutorial 2067, pages 100-125, F. Cassez, C. Jard, B. Rozoy, and M. Ryan (Eds.), 2001.
- [BaPa00] G. Bauer, M. Paulitsch, An Investigation of Membership and Clique Avoidance in TTP/C, 19th IEEE Symposium on Reliable Distributed Systems, October 2000, Nürnberg, Germany.
- [Bh84] U.N Bhat. Elements of Applied Stochastic Processes. John Wiley & Sons 1984.
- [Cr&all02] L. Cristaldi, A. Ferrero, C. Muscas, S. Salicone, R. Tinarelli - The effect of net latency on the uncertainty in distributed measurement system - 18th IEEE/IMTC Instrumentation and Measurement Technology Conference, Anchorage, Alaska, USA, 21-23 May 2002.
- [FTC01] OSEK/VDX Fault-Tolerant Communication, Specification document, Edition 1.0, July 2001, [www.osek-vdx.org](http://www.osek-vdx.org)
- [GaWiAs03] M. Gäfvert, B. Wittenmark and O. Askerdal. On the effect of transient data errors in controller implementation, In Proceedings of American Control Conference 2003, Denver, US, June 2003.
- [GoAuMi04] K. Godary, I. Augé-Blum, A. Mignotte, Temporal Bounds for TTA : Validation, RR, INSA Lyon, January 2004.
- [JuNaSi03] F. Jumel, N. Navet, F. Simonot "Influence des performances d'une architecture informatique sur la fiabilité des systèmes échantillonnés. (in french) " in Real-Time Systems and Embedded Systems 2003 - RTS, Paris, France, April 2003.
- [JuThAu03] F. Jumel, J-M. Thiriet, J.-F. Aubry, O. Malasse "Towards an Information-Based Approach for the Dependability Evaluation of Distributed Control Systems" in IEEE Instrumentation and Measurement Technology Conference (IMTC 2003), Vail, USA, May 2003.
- [KoGr94] H. Kopetz, G.Grünsteidl, TTP - A Protocol for Fault Tolerant Real-Time systems, IEEE Computer, pages 14-23, January 1994.
- [Ko98] H. Kopetz, The Time-Triggered Architecture, IEEE International Symposium on Object-Oriented Real-Time Distributed Computing (ISORC'98), April 1998, in Kyoto, Japan
- [Ko&all97] H. Kopetz R. Hexel, A. Krüger, D. Millinger, and S. Schedl, A Synchronization Strategy for a TTP/C Controller, In SAE Congress and Exhibition, number 960120 in SAE paper, Detroit, MI, USA, Feb 1997
- [Mc&all95] J.A. McDermid, M. Nicholson, D.J. Pumfrey, P. Fenelon - Experience with the application of HAZOP to computer-based systems, 10th Annual IEEE Conference on COMPUTER ASSurance (COMPASS '95).
- [Rus01] J. Rushby, A Comparison of Bus Architectures for Safety-Critical Embedded Systems, SRI International Computer Science Laboratory (CSL) Technical Report, 2001.
- [ShRaGa96] L. Sha, R. Rajkumar, and M. Gagliardi. Evolving dependable real-time systems. In IEEE Aerospace Applications Conference, Aspen, US, February 96
- [TTP02] Time-Triggered Technology, TTTech computertechnik AG, Vienna, Austria, Specification of the TTP/C protocol, edition 1.0.0 , july 2002.