SAE TECHNICAL PAPER SERIES

2016-36-0293 E

A Discussion on Time Synchronization and their Effects in Distributed Cyber-Physical Control Systems

ELOY MARTINS OLIVEIRA JUNIOR
MARCELO LOPES DE OLIVEIRA E SOUZA

AFFILIATED TO

25th SAE BRASIL International Congress and Display
São Paulo, Brasil
October, 25th to 27th

AV. PAULISTA, 2073 - HORSA II - CJ. 1003 - CEP 01311-940 - SÃO PAULO – SP

2016-36-0293

A Discussion on Time Synchronization and their Effects in Distributed Cyber-Physical Control Systems

Eloy Martins de Oliveira Junior
Marcelo Lopes de Oliveira e Souza
National Institute for Space Research – INPE – Brazil
Division of Space Mechanics and Control-DMC

Abstract

Cyber-physical systems are joint instances of growing complexity and high integration of elements in the information and physical domains reaching high levels of difficulty to engineer an operate them. This happens with satellites, aircraft, automobiles, smart grids and others. Current technologies as computation, communication and control integrate those domains to communicate, synchronize and operate together. However, the integration of different domains brings new challenges and adds new issues, mainly in real time distributed control systems, beginning with time synchronization. In this paper, we present a discussion on time synchronization and their effects in distributed cyber-physical control systems. To do that, we review the literature, discuss some time synchronization techniques used in cyber-physical systems, and illustrate them via model and simulation of a system representative of the aerospace area. We also show how the time synchronization can affect the transient and stability of distributed cyber-physical control systems and we propose a way to alleviate those issues. The results obtained so far suggest that the proposed way is effective in reducing the effects of time synchronization in distributed cyber-physical control systems and we hope it will contribute to their development in the near future.

Introduction

Cyber-physical systems (CPS) consists in an integration of different instances as computation, communication and control operating together, reaching high levels of difficulty to design the system. This happens with satellites, aircraft, automobiles, smart grids and others. Such systems demand high performance, precision, accuracy, modularity, integration, dependability and others, requiring predictability and determinism of logical domain and temporal domain.

The temporal order is required to achieve a proper coordination among different tasks, distributed or not. To achieve it, local clocks have been used to produce a local time representation and, time synchronization techniques have been used to correct the local time representation produced by clocks. All together to achieve a common sense of time among nodes of distributed systems. At this point, the reader might ask if my system does not need a common sense of time and time synchronization, why can it requires the temporal order and produce a local time representation? There is no easy answer; all depends of requirements of your system. However, many systems abstract the system in different layers, and most of them represents the time in a low layer levels, not available, or disregarded, by the designer. According to Lee (2008) [1], timing is not part of the semantics of C language, however, it is relevant to determine whether system has performed correctly.

This paper shows the mainly challenges and issues of CPS related to time synchronization of real time distributed systems. To understand how the time is represented and, hence, synchronized among different nodes and domains, first it is necessary a good understand about the time representation and synchronization techniques used at each domain. Before, the time synchronization issues due to the integration of different domains is presented. This paper also show how the time synchronization can affect the transient and stability of distributed cyber-physical control systems and proposes a way to alleviate those issues. All to understand and mitigate the effects of time synchronization in a realization of CPS.

Time Representation and Synchronization

Passage of time is an important issue in CPS. This problem is not new and has been studied for years, as could be observe in [2,3,4,5]. However, to do a good design, it is necessary understand how the time is represented and synchronized in each domain. Therefore, it is possible to understand the issues and effects of synchronization that come with the integration of different domain.

According to Kopetz *et. al* (1996) [5] and Kopetz (1997) [3], the representation of time must be independent of individual characteristics of each oscillator. That is, the representation of time requires to be on a common nominal frequency to the set. Furthermore, the representation should meet the following criteria:

- Should be understandable by humans;
- Must be independent of details of implementation and speed of communication;
- Must be controllable and / or observable by the computer;

Time is represented by clocks, and clocks must be produced by local oscillator that represents clocks in different timescales, as showed by Figure 1.

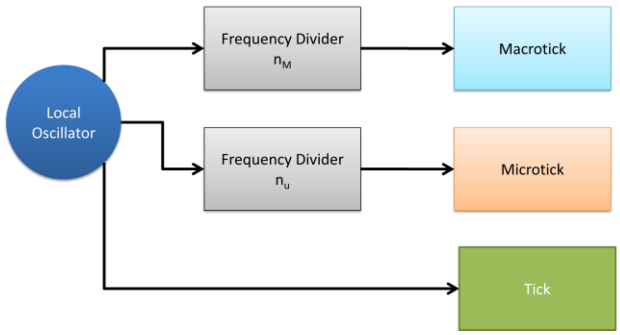

Figure 1. Time signals of a local node. Adapted Source: [4].

A local physical clock as a counter and a physical oscillation mechanism that periodically generates events that will add counts to the counter. This recurring event is set to ticks of the clock;

A local physical time as microtick which is defined as a periodic event generated by the frequency divider with a different frequency of local physical clock;

A local virtual time as macrotick which is defined as a periodic event generated by the frequency divider.

Real time systems use these time representation and synchronization to achieve a temporal order and hence, achieves determinism and predictability. However, in cyber physical systems, it is necessary to understand how the time representation and synchronization arises due to integration of different domains. This paper will be take care the computational, communication and control domains. Figure 2 shows an example of a simple cyber physical system with these domains.

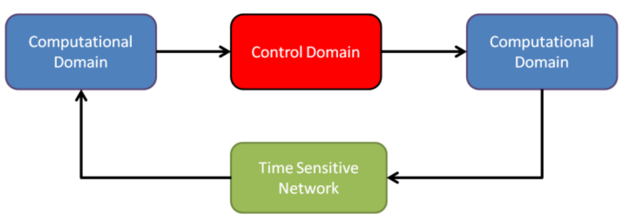

Figure 2. Simple cyber physical system example.

Computational domain

The computational domain has been ruled by a sequential logical actions that produces desire output, most of them, disregarding the timing, or not included in it semantics. In fact, it is not a problem when the computational domain does not interacts with physical domain and does not operates in a real time system. However, in CPS the computational domain interacts with physical domain, and most of them in real time.

According to Lee (2008) [1], the lack of timing in computational domain has been exploited heavily in such computer science disciplines as architecture, programming languages, operating systems, and networking. In fact, the architecture, operating systems, programming languages and networking have been more elaborate to avoid the non-deterministic behavior of the computational domain. In fact, according to Lee (2015) [6] computational domain is a non-deterministic environmental.

Therefore, full timing and determinism in computation domain could be achieved only in a low-level of programming, taking care time into low-levels of programming. However, it is hard to produced these solutions for integrated and complex systems. So, the solution are abstract the timing lack of computation into a time sensitive networking; and encapsulated the solutions into a formal rules. All of them loosing capability.

Communication domain

Time sensitive networks (TSN) have been developed to provide a more deterministic behavior. Time synchronization is a critical part of a TSN. In fact, TSN provides a deterministic interface among computational domain and control domain. Therefore, these solutions use time synchronization techniques to achieve a common time sense among nodes interconnected to a network. This solution fill partially the timing lack due to the computation. In fact, this solutions guarantees the worst case delay.

Control domain

The control domain is part of physical domain. Therefore, the passage of time is continuous and all actions are concurrent. The time is a independent variable and in many mathematical models is used to calculate the control laws. Here, the time is a natural phenomena and all domains must be follow these phenomena. However, the models and controller have designed under some suppositions and imperfect models. So, the issues about time come when the plant must be controlled by a computer using a network to exchange data and hence, their nodes must be synchronized to establish a common time sense.

Time synchronization techniques

Time synchronization problem appears in many domains. Clocks, that produce a time representation, have imperfections caused by the environmental fluctuations, aging, the non-linear dynamics and/or long lifetimes. These imperfections may cause a fluctuation beyond a tolerance in clock synchronization (called as clock de-synchronization) which can cause time faults or failures. There are many techniques related to minimize the clock de-synchronization in different levels. An algorithm has been the main method used to perform clock synchronization. In order, clock synchronization algorithms establish a global sparse time model that supports the consistent ordering of events Kopetz *et. al* (1996) [5] of a safety-critical distributed system. Below, this paper present two algorithms that have been used in many distributed real time systems as aerospace and automotive systems to achieve clock synchronization and establish a common time sense.

FTM - Fault - Tolerant Mid-Point

The FTM (Fault-Tolerant Mid-Point) algorithm, also known as Welch-Lynch algorithm [2] provides fault tolerance for byzantine clock synchronization to distributed systems. Based on Welch and Lynch [2], to ensure that all nodes have a consistent view of time, the system requires regularly (periodically) re-synchronization of clocks. The FTM algorithm is distributed/democratic approach, with deterministic policy. It fault tolerance are in sense of byzantine errors.

So, to achieve clock synchronization, the algorithm establishes a global time (virtual master clock). The global time is a set of mathematical equation (convergence function) that involves a subset of the set of clocks. The clocks, individually, follow a logical sequence described in Figure 3 and converge it local time to a global time.

Each node applies this algorithm with the goal to reach a correction term. The correction term adjusts the skew caused by the clock drift within a certain precision.

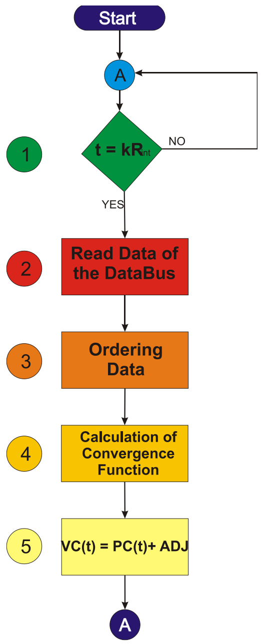

Figure 3. Flowchart of FTM Algorithm.

PTP

IEEE 1588-2002 and now 2008 standard [7,8] defines Precision Time Protocol (PTP) and other protocols and rules to establish clock synchronization in a distributed system over a network. According to National Instruments White Paper (2013) [8], the IEEE 1588 specification does not include any standard implementation for adjusting a clock; it merely provides a standard protocol for exchanging these messages. This standard is mostly applied in systems that make use of clock synchronization based on a real master clock. PTP uses a master/slave approach, *i.e.*, the slave clocks receive periodically a time from real master clock and then compare and adjust their times. The IEEE 1588 is a good candidate to be used in cyber physical systems as standard to establish common time sense among nodes in a network, as showed in Broman *et. al* [11].

In most cases the algorithms and techniques to establish a clock synchronization follow Figure 4 shows the PTP scheme.

According to National Instruments White Paper (2013) [8], in the PTP, the master clock periodically sends a sync packet containing a timestamp of the time when the packet has left. The master may also, optionally, sends a follow up packet containing the timestamp for the sync packet. The follow up packet allows the master to accurately timestamp the sync packet on networks where the departure time of a packet cannot be known accurately beforehand. A slave clock receives the master's sync packet and using its own clock timestamps in the packet's arrival time. The difference of the sync packet departure timestamp and the sync packet the arrival timestamp is the slave clock's offset from the master with the network propagation delay. By adjusting its clock by the offset measured at this point, the offset between the master and slave can be reduced to the network propagation delay only. Assuming the symmetrical delay, the slave can discover, and compensate the propagation delay. It accomplishes this by issuing a delay request packet which is timestamped on departure from the slave. The delay request message is received and time stamped by the master clock, and the arrival timestamp is sent back to the slave clock in a delay response packet. The difference in these two timestamps is the network propagation delay.

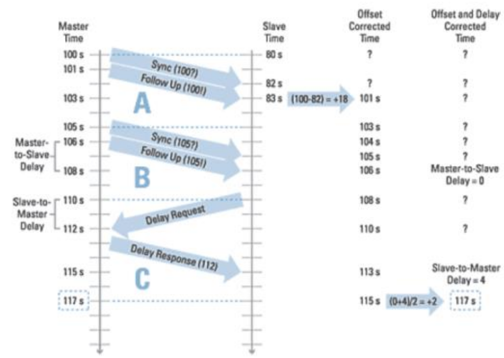

Figure 4. PTP scheme. Source: [5].

Model and Simulation

This paper simulates clock synchronization algorithms from literature in a networked control system with TDMA and CSMA/CD, as showed Figure 5.

For this simulation, this paper used the TrueTime/Matlab/Simulink environment [9] for simulations. The master clock, sensor, actuator and controller were connected via a time sensitive network. The controller used was a digital PID (Proportional, Integral and Derivative), under a computational device. The actuator/plant was a second order marginally stable continuous time system, according to the following transfer function:

$$\frac{1000}{s^2+s} \tag{1}$$

All nodes had logical clocks given by the virtual computer of the TrueTime Kernel; and they used the databus network to exchange data among them. The master time is the virtual time given by the logical clock of the TrueTime Kernel, namely master clock node. In the TrueTime Kernel it is possible to manage the virtual clocks. So, it is possible to insert faults (imperfections) and correct them.

The simulated cases choose are similar from literature. The cases intend to show that time synchronization techniques can affect the transient and stability of NCS showed in Figure 5, which is an simple version of cyber-physical system.

The simulated cases are :

1. PTP with TDMA;
2. FTM with CSMA/CD.

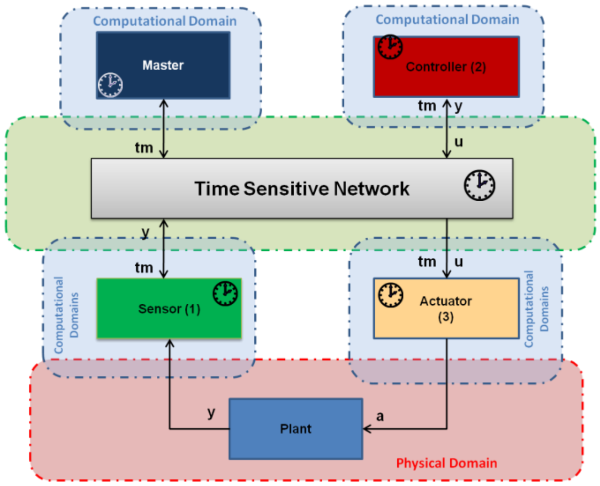

Figure 5. Simple cyber physical system example.

Case 1

This case is similar of the case presented in Oliveira Junior and Souza (2013) [4]. This case applies 0.39 seconds of initial offset on virtual clock of Sensor (1). The clock of sensor starts 0.39 seconds ahead than other nodes. This initial offset is excessive, but it was used only for didactic purposes.

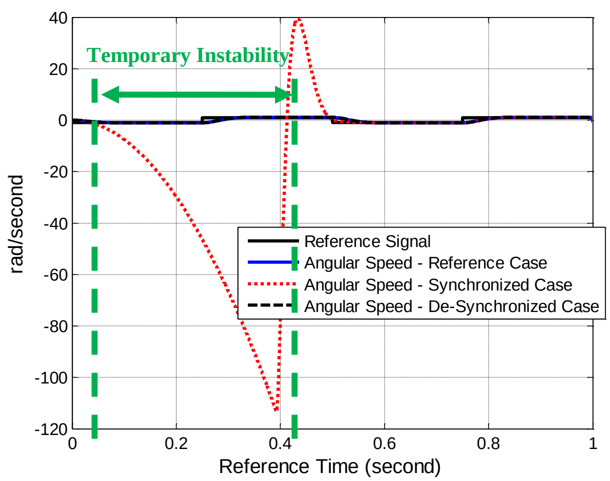

Figure 6. Successive Step responses of Networked Control System Model.

All clocks were synchronized using PTP. However, due to clock synchronization algorithm (PTP) the control law and dynamic response is degraded.

Case 2

This case is similar to presented in Oliveira Junior and Souza (2015) [10]. This case shows a clock synchronization with the FTM algorithm synchronizing the node in a networked control system over CSMA / CD.

This case presents the follow configuration for simulation:

1) Initial offsets values:

- Sensor 1: 0;
- Control 2: 0.1 sec;
- Actuator 3: 0.01 sec;
- Master: -0.1 sec.

2) Drifts:

- Sensor 1: 0;
- Control 2: 1%;
- Actuator 3: 0.01%;
- Master: -0.01%.

3) MMCF:

- Sensor 1: 1;
- Control 2: 2;
- Actuator 3: 2;
- Master: 2.

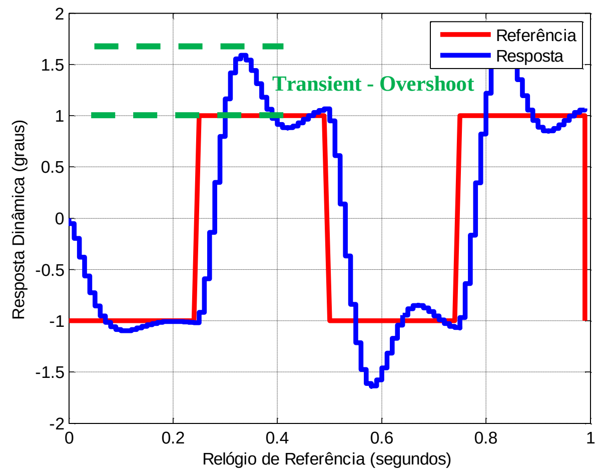

Figure 7. Dynamic Response.

The FTM causes large excursion of time which degrades the control law, and, consequently, causing the overshoot in dynamic response. It occurs because the FTM algorithm is not designed to address the initial de-synchronization, and hence to achieve clock synchronization, the control system is degraded.

Mitigation

The simulation shows that the time synchronization can affect cyber-physical systems. In fact, most of time synchronization algorithms have been designed to synchronize clocks, but, not taking care the integration among different domains. This paper shows the degradation of dynamic response, that presents the transitory and instability in it response.

The ways to alleviate these effects on dynamic response can be due in different ways. In fact, many solutions have been designed to take care these effects in cyber-physical systems.

The first case shows a temporary instability. The instability is caused because the clock is corrected, but due to the correction of time the computer schedule is mistaken, and hence, the tasks are temporary stopped. So, to correct it clock synchronization algorithms could be re-designed and avoid huge excursion of time . However, the most prominent solution is in re-design programming languages to take care timing in his semantics. However, in Lee (2015) [6] shows that better models together with better implementations yield determinism (on the modeling side) and high fidelity (in the physical implementation). Further, in Lee (2015) [6] is proposed PRET machines that provides an abstraction layer between the PRET hardware and the modeling language, to give to the programmers control over timing. The acronym “PRET” stands variously for PREcision-Timed processors, Predictable, REpeatable Timing, and Performance with REpeatable Timing. However this approach requires new technologies, software and paradigms.

In Oliveira Junior and Souza (2015) [10] is proposed a reconfigurable clock synchronization algorithm that changes his states. This approach faces with the different behavior of plant and hence, it is possible to apply different time synchronization algorithms for each state of system, taking care different parameters. This approach requires new algorithms and software, however, does not requires new paradigms.

Conclusions

The approach of Lee (2015) [6] intend to extend the paradigm of determinism and predictability to computational domain. So, it requires the development of new technologies and paradigms in programming language. Today, the determinism of computation is achieve abstracting to the network level, using synchronous languages and formal methods. The approach of improves the current paradigms as in Oliveira Junior and Souza (2015) [10] requires new algorithms and models to improve the performance and reduce the loose of capability. Therefore, the cyber physical systems have been developed in two ways. The first, is a middle-term solution, improving the current technology and algorithms. The second, is a long-term solution which consist in development of new technologies and paradigms for all domains. Both solutions improves the response of cyber-physical systems and reduce the effects of time synchronization due to the lack of timing.

References

1. Lee, E. A. "Cyber physical systems: Design challenges." Object Oriented Real-Time Distributed Computing (ISORC), 2008 11th IEEE International Symposium on. IEEE, 2008.
2. Lundelius, J. and Lynch, N., “A New Fault-Tolerant Algorithm for Clock Synchronization”, presented at Third Annual ACM Symposium on Principles of Distributed Computing, Vancouver, Canada, 1984.
3. Kopetz, H. Real time systems: design principles for distributed embedded applications. Norwell, MA, USA : Kluwer Academic Publishers, 1997.
4. Oliveira Junior, E. M. and Souza, M. L. O., "An Overview of Clock Synchronization Algorithms and their Uses in Aerospace and Automotive Systems," SAE Technical Paper 2013-36-0541, 2013, doi:10.4271/2013-36-0541.
5. Kopetz, H., Hexel, R., Krüger, A., Millinger, D. et al., "A Synchronization Strategy for a TTP/C Controller," SAE Technical Paper 960120, 1996, doi:10.4271/960120.
6. Lee, Edward A. "The past, present and future of cyber-physical systems: A focus on models." Sensors 15.3 (2015): 4837-4869.
7. Gaderer, G.; Rinaldi, S.; Kero, N. "Master Failures in the Precision Time Protocol," Precision Clock Synchronization for Measurement, Control and Communication, 2008. ISPCS 2008.
8. National Instruments White Paper, "Introduction to Distributed Clock Synchronization and the IEEE 1588 - Precision Time Protocol", http://www.ni.com/white-paper/2822/en, Fev. 2013, Access on 01/05/2013.
9. Ohlin, M.; Henriksson, D.; Cervin, A., “TrueTime 1.5 – Reference Manual”, Department of Automatic Control, Lund University, Sweden, Jan. 2007.
10. Oliveira Junior, E. M.; Souza, M. L. O. "A New Clock Synchronization Algorithm to Compensate the Initial De-Synchronization of Clocks Based on a Deadbeat Controller for Networked Control Systems." No. 2015-36-0400. SAE Technical Paper, 2015.
11. Broman, D., Derler, P. and Eidson, J., 2013. Temporal issues in cyber-physical systems. Journal of the Indian Institute of Science, 93(3), pp.389-402.

Contact Information

Eloy M. Oliveira Junior, Ph.D

Former Graduate Student[1]
System Developer[2]

National Institute for Space Research-INPE[1]
Course of Space Engineering and Technology/Option Space[1]
Mechanics and Control-ETE/CMC[1]
Av. dos Astronautas, 1758 - Satellite Building, Room 31[1]
São José dos Campos, São Paulo, 12227-010, Brazil[1]

Konatus[2]
Rua Benedito Osvaldo Lecques, 51 - 6° Floor - Room 608[2]
São José dos Campos, São Paulo, 12246-021, Brazil[2]
Phone: 55-12-981465075, E-mail: eloymojr@gmail.com

Marcelo L. O. Souza, Ph.D

Professor and Senior Researcher
National Institute for Space Research-INPE
Division of Space Mechanics and Control-DMC
Av. dos Astronautas, 1758 - Satellite Building, Room 27
São José dos Campos, São Paulo, 12227-010, Brazil
Phone: 55-12-32086198, E-mail: marcelo.souza@inpe.br).

Acknowledgments

Both authors thank the National Institute for Space Research - INPE of Brazil, its Course and Division of Space Mechanics and Control-CMC/DMC for supporting them during the writing of this paper.

ISSN 0148-7191