ISSN:1983 7402 ITA, 23 a 25 de setembro de 2014

A Discussion on Security Evaluation of Time Synchronization Algorithm via Modeling and Simulation

Eloy Martins de Oliveira Junior, Marcelo Lopes de Oliveira e Souza
Instituto Nacional de Pesquisas Espaciais – INPE – Av. dos Astronautas, 1758, São José dos Campos, SP, Brasil

***Abstract* — Satellites, aircrafts, automobiles, power controls and traffic controls are part of critical infrastructure of a country. The security requirements for these systems are increasing to improve the capability to avoid malicious attacks. These systems integrate computations, communications and real time controls in a deterministic and distributed environment. Networked Control Systems (NCSs) are an example of this. An NCS interconnects, in different levels of operations, a large number of actuators, sensors and controllers. To achieve the determinism, it requires time synchronization among its nodes. However, the time synchronization process is a potential target for malicious attacks attempting to disrupt the normal operation of systems. This paper presents a discussion on security evaluation of fault-tolerant mid-point time synchronization algorithm via modeling and simulation. This discussion shows preliminarily: 1) the effects of malicious attack in time synchronization algorithm; 2) the potential effects on systems; and 3) suggestions for countermeasures.**

***Keywords* — cyber warfare, time synchronization, cyber-security**

I. INTRODUCTION

Current systems such as satellites, aircrafts, automobiles, turbines, power controls, satellite tracking stations and traffic controls are part of the National Critical Infrastructure (NIC) of a country. Such systems are becoming increasingly complex and/or highly integrated as prescribed by the SAE-ARP-4754A Standard [1]. They integrate computations, communications and real time controls in different levels of operations, using a large number of actuators, sensors and controllers implemented in a distributed architecture connected via a common network to form a Networked Control System (NCS), which frequently requires time synchronization among different devices and levels. Such systems require predictability in the logical domain and in the temporal domain [2], called in this work as “determinism”.

According to [3], the cyber warfare domain is a global domain within the information environment consisting of the interdependent network of information technology infrastructure, including the internet, telecommunications networks, computer systems, embedded processors and controllers. Such systems require a defense policy.

Time synchronization security is an important aspect when clock synchronization is used in commercial applications, in public networks or even in critical control applications which make use of a network. In general, the problem of time synchronization of logical clocks caused by natural imperfections is solved by algorithms. However, it is obvious that the overall functionality of those systems can be degraded or even disabled if the mechanism of synchronization of clocks is attacked [4]. The attacks to the mechanism of synchronization of clocks can cause faults and risks of accidents.

According to [5], defects in sensors, in actuators, in the process itself, or within the controller, can be amplified by the closed-loop control systems, and faults can develop into malfunction of the loop. At [6], it is shown how the control and response of dynamic systems are degraded due to a maliciously attack in a time synchronization process. It makes time synchronization process a prime target for malicious attacks attempting to disrupt the normal operation [7].

There are different ways used to protect systems today, covering the national infrastructure and processes used to protect, detect, react and recover the systems.

The security evaluation of time synchronization can be done by different means as Vulnerabilities Assessments (VAs) and Penetration Tests (PTs). VAs are designed to look for vulnerabilities on the network and then prioritize how to fix or mitigate them [3]. The PTs are designed to test the ability of system to respond to an intrusion [3]. However, VAs and PTs are usually done on systems already developed and running. So, if the VAs and PTs are not conducted properly, they both can be hard, expensive and probably impact the system operation.

Thus, this paper shows how the security evaluation of time synchronization algorithm can be performed in an affordable way via modeling and simulation focusing in a democratic approach. Basically, a democratic approach achieves clock synchronization using the concepts of microtick, macrotick and global time (virtual master clock) [8]-[9]. However, the security evaluation via modeling and simulation required a good knowledge not only about time synchronization process, but also an analytical model of entire distributed system.

II. BASIC CONCEPTS

This section presents: 1) a clock synchronization imperfections; and 2) the democratic approach. This paper uses the discrete clock synchronization in all simulation and models.

ISSN:1983 7402

Clock Imperfections

A physical clock and, therefore, logical clocks have some imperfections. These imperfections can be caused by environmental changes such as variations in temperature, voltage, aging of crystal in case of quartz clock, and maliciously attacks. Fig. 1 shows the main imperfections of a clock.

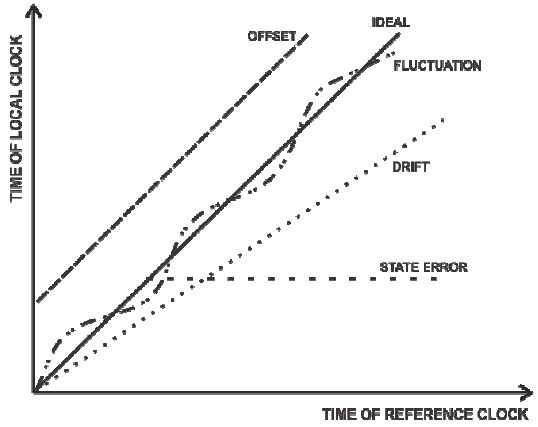

Fig. 1. Main Imperfections of a Clock.

These imperfections are known as drift, initial or instantaneously offset, fluctuation and state error:

• **Clock Drift:** Clock drift is when a local clock has a frequency of oscillation greater or less than another local clock and/or a reference clock; i.e., the drift is the rate of change between the two clocks.
• **Offset**: There are two types of offset: the initial offset is the difference between the initial times of local clocks and/or of a reference clock; the instantaneous offset is the difference between the instantaneous times of local clocks and/or of a reference clock.
• **Fluctuation**: The fluctuation or jitter is the uncertainty in the measurement of the time.
• **State Error:** The state error is when a local clock stops the measurement of the progression of time; i.e., the local clock stops on a fixed value.

These imperfections of a clock are impossible to eliminate. Thus, clock synchronization algorithms are used to achieve synchronization within a tolerance and to minimize the effects of these errors in a system. So, the attackers exploit these imperfections and clock synchronization algorithms to impact maliciously the system.

Besides the imperfections of clock, Figure 2 shows three steps that must be considered in the process of clock synchronization.

Figure 2.Clock Synchronization Steps.

• **Generation**: is a part that generates periodically events to increase the counter.
• **Distribution**: is a part that distributed a clock measure among a set of clocks to be synchronized.
• **Synchronization**: is a part of that synchronized each clock in relation to one reference (virtual or real).

This work is focused in models that represent clock synchronization algorithms (third block in Figure 2), abstracting all other unnecessary details. In fact, the generation and distribution are not disregarded, and their effects are modeled as clock drifts, offsets, fluctuations, delays and perturbations.

Democratic Approach

The democratic approach does not use a real master clock to synchronize the system. To achieve clock synchronization, this architecture uses the concepts of microtick, macrotick and global time (virtual master clock).

The microtick is a measure of physical clock, the macrotick is a logical clock and, global time is the reference clock of set of clocks achieved by one mathematical equation involving a subset of the set of clocks. To have a consistent view of the global state of the distributed system, all nodes must be synchronized via one clock synchronization algorithm.

This approach is frequently used in communications via databus, where the global time must be achieved for correct operation of network. Fig. 3 illustrates this approach over a databus.

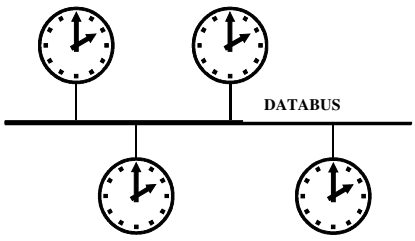

Fig. 3. Democratic Approach.

Fig. 3 shows four local clocks, where each of them calculates its correction. These calculations by each clock make the convergence to a global time, i.e., all clocks converge to the same time within a tolerance. This approach has the advantage of allowing byzantine fault tolerant clock synchronization algorithms. This increases the reliability of the system, but it increases the traffic of data and adds a cost on the precision achieved among the set of clocks. The FTA (Fault-Tolerant Average) algorithm [10] is an example of a byzantine fault tolerant algorithm for a distributed architecture. To ensure that all nodes have a consistent view of time, the re-synchronization of clocks is needed regularly (periodically).

III. CLOCK SYNCHRONIZATION MODEL

Clock models have been devised in the literature, each one with its advantages and disadvantages. In [11], it was proposed a clock synchronization algorithm, using clock state and rate correction algorithm, with the concepts of microtick, macrotick and global time. Fig. 4 shows the timer unit proposed by [11] to be used in safety critical communications based in a Time Division Multiple Access (TDMA).

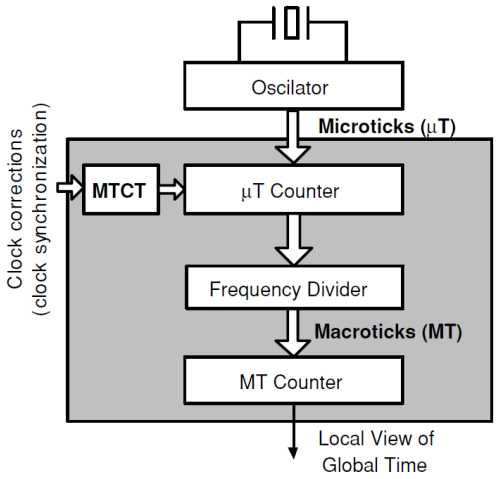

Fig. 4. Timer Unit Approach.
Source: [11].

However, the mathematical model of [11] is not in form of recurrence equation. The mathematical model in form of recurrence equations gives the possibility to use the z-transform theory to make the analysis about the system.

So, the mathematical model of Fig. 4 is modeled as a recurrence equation. Disregarding the uncertainties, the microtick of local clock *i* is defined as:

$$mt_i(t) = D_i t + b_i \quad (1)$$

Where:

- D_i is a drift rate of local oscillator;
- b_i is an initial offset in relation of reference clock.

Discretizing (1), then:

$$t = kT \quad (2)$$

Where:

- k is an instant with $k = 1, 2, 3, \ldots, n$;
- T is a sampling period.

So, from (1) and (2), the discrete equation of microtick is:

$$mt_i(k) = (D_i)kT + b_i \quad (3)$$

Making (k+1) at (3):

$$mt_i(k+1) = D_i kT + D_i T + b_i \quad (4)$$

Replacing (3) in equation (4), the microtick is:

$$mt_i(k+1) = mt_i(k) + D_i T \quad (5)$$

So, the equation (5) is the recurrence model of microtick of local clock *i*. The reference clock *z* is defined as:

$$mt_z(k+1) = mt_z(k) + T \quad (6)$$

The recurrence of macrotick of local clock *i*, doing the same algebraic manipulation of microtick, is defined as:

$$Mat_i(k+1) = Mat_i(k) + \frac{[mt_i(k+1) - mt_i(k - MMCFi)] \cdot c}{MMCF_i} \quad (7)$$

Where $MMCF_i$ is the microtick-macrotick correction factor and *c* is defined as an impulse train function:

$$c = \begin{cases} \sum_{k=1}^{N} \delta(t - kT \cdot MMCF_z) \\ 0, \quad if\ k \leq 0 \end{cases} \quad (8)$$

So, the global time of local clock *i* is defined as:

$$GT_i(k+1) = MaT_i(k+1) + adj_i(k+1) \quad (9)$$

Where $adj_i(k+1)$ is the adjustment function provided by fault-tolerant clock-state synchronization algorithm at node i. The FTA algorithm model is:

$$adj_i(t) = \frac{[\sum_{k=1}^{n} MaT_k(t)] - \max(MaT_k(t)) - \min(MaT_k(t))}{n-2} \quad (10)$$

Where:

- n – number of clock in system;
- max(MaTk(t)) – maximum value of set;
- min(MaTk(t)) – minimum value of set.

IV. SECURITY EVALUATION

The attacks to the mechanism of time synchronization can cause faults and risks of accidents. In general, each system requires a security policy of time synchronization. Each system has its own vulnerability list, as shown in [6].

This paper proposes a security analysis of time synchronization via modeling and simulation. There are many performance metrics that could be used as the stability. For the clock synchronization, there are other important metrics as precision and accuracy. So, each temporal system design has its own important metrics to be analyzed.

Many systems use the concept of microticks and macroticks. With this, it is possible to perform many vulnerability analyzes. This work analyzes the attack in the macrotick of clock synchronization process. To do it, this work uses the attack factor proposed by [6].

ISSN:1983 7402

Many attack cases could be simulated considering the presence of an attack factor only at macrotick. The model of macrotick is redefined by:

$$Mat_i(k+1) = [Mat_i(k) + \frac{[mt_i(k+1) - mt_i(k - MMCF(t)] \cdot c}{MMCF_i}]\lambda \quad (11)$$

To proceed with the security evaluation, we choose an *ad hoc* attack factor λ to show how the system is robust in relation to logical attack at the macrotick.

TABLE I. PARAMETERS CONFIGURATION FOR ATTACK FACTOR

Case	Attack Factor (λ)	Clock Attacked	Attack Type
0	***1***	No Attack	No Attack
1	***0.05***	Clock 1	Offset Attack
2	***0.1***	Clock 1	Offset Attack

Case 0

The results without attack are presented in Figures 5 to 8.

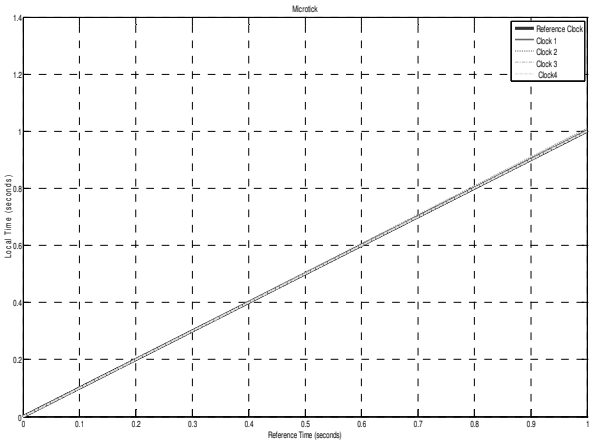

Fig. 5. Microtick without attack

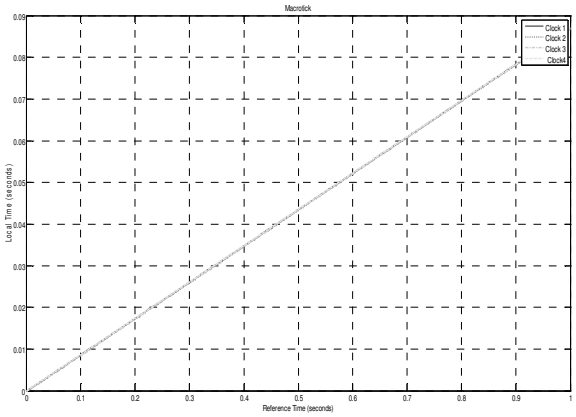

Fig. 6. Macrotick without attack.

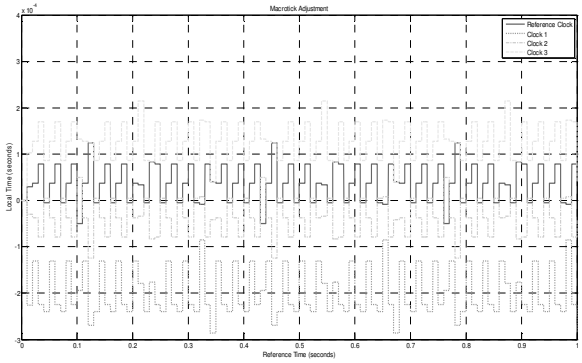

Fig. 7. Macrotick adjustment without attack.

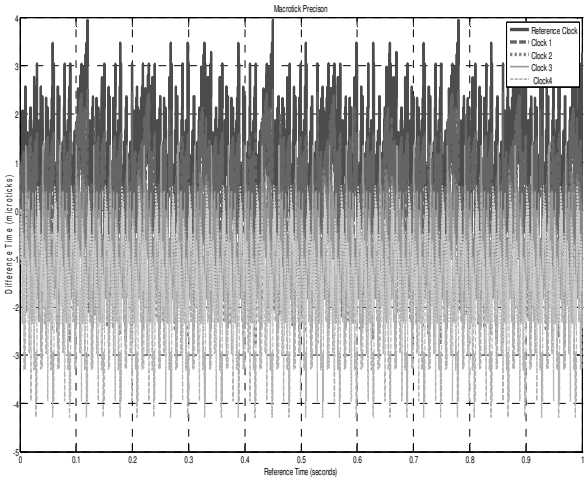

Fig. 8. Macrotick Precision without attack.

Case 1

The attack factor is a good way to see the results of clock synchronization with a parameter variation. The results with attack factor λ = 0.05 at Macrotick of Clock 1 are presented in Figures 9 to 12.

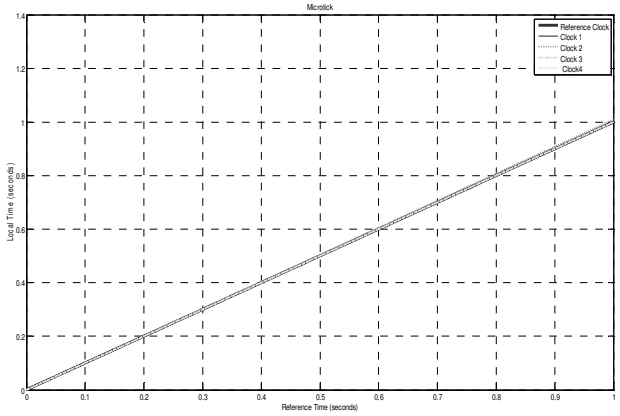

Fig. 9. Microtick under attack at Macrotick with 0.05 AF.

ISSN:1983 7402

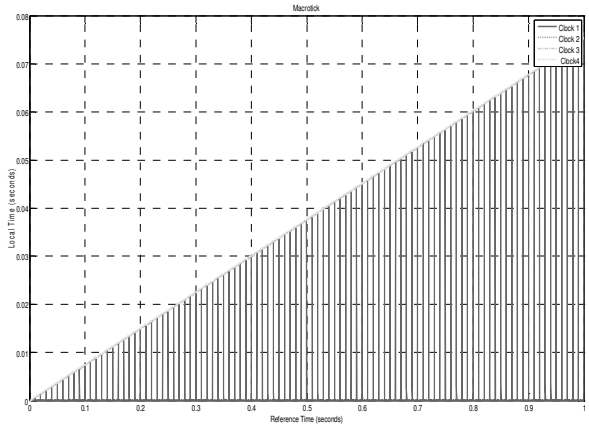

Fig. 10. Macrotick under attack with 0.05 AF.

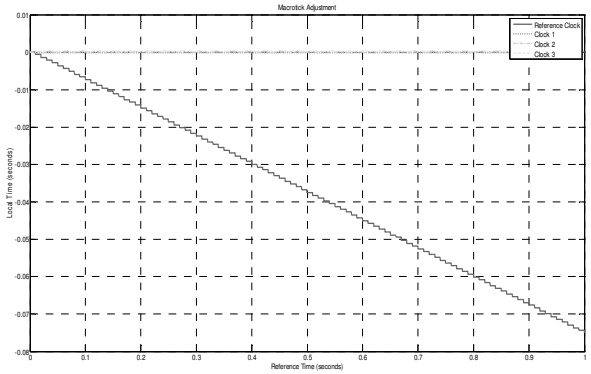

Fig.11. Macrotick adjustment under attack at Macrotick with 0.05 AF.

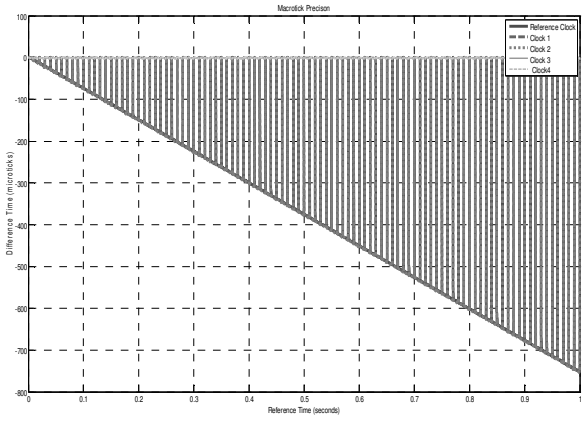

Fig. 12. Macrotick Precision under attack at Macrotick with 0.05 AF.

Fig.10, 11 and 12 shows that with a minimum variation of attack factor (AF) degrades the clock synchronization in comparison with case 0.

Case 2

The results with 0.1 of attack factor at macrotick of Clock 1 is presented at Figures 13 to 16.

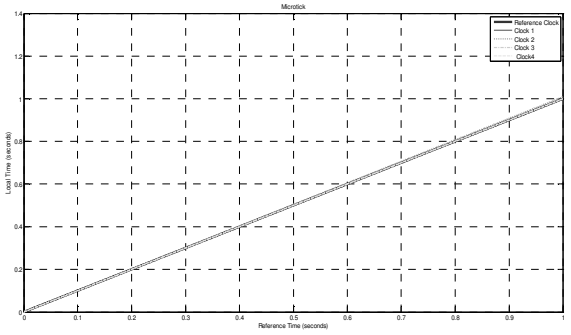

Fig. 13. Microtick under attack at Macrotick with 0.1 AF.

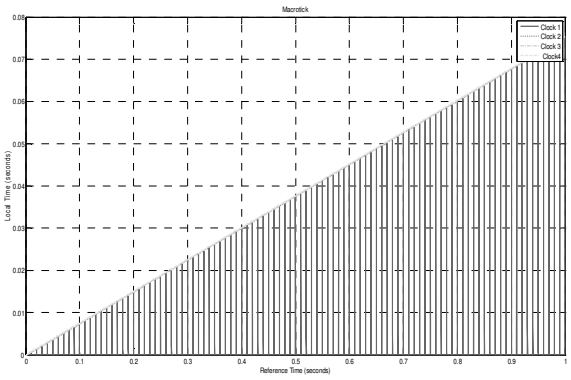

Fig. 14. Macrotick under attack with 0.1 AF.

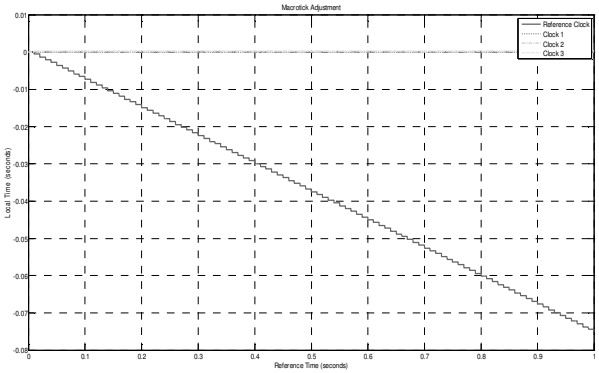

Fig.15. Macrotick adjustment under attack at Macrotick with 0.1 AF.

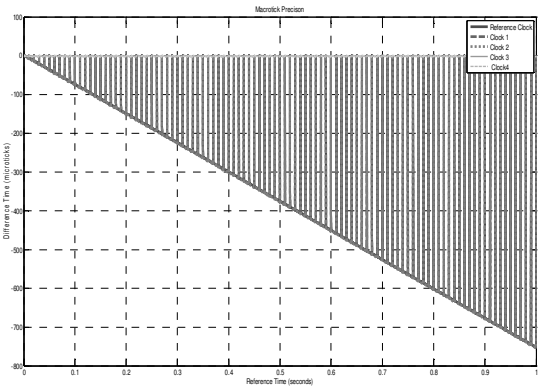

Fig. 16. Macrotick Precision under attack at Macrotick with 0.1 AF.

Figures 14, 15 and 16 show again that a small variation of attack factor (AF) degrades the clock synchronization, in comparison with Case 0.

The result obtained by simulation suggests that the microtick keeps stable even if the macrotick is attacked. The macrotick is more vulnerable to malicious attacks than the microtick. So, the microtick could be used as defense metric to the macrotick. It is possible to make an attack at the microtick; however, it is harder for an external malicious attacker reach this level.

Other important result is that due to the byzantine ability of FTA clock synchronization algorithm, even the Clock 1 under attack the other clock macroticks is not affected. However, this mechanism could be attacked.

V. CONCLUSIONS

The FTA algorithm provided a good technique to achieve clock synchronization and proved efficient to achieve the time synchronization under byzantine attacks. However, the effect over a dynamic response of networked control system is not analyzed. The attacks at clocks can be cause time discontinuities. This effect may cause a problem to the dynamic response of closed-control loop. In fact, if the control is dependent of time synchronization as the NCSs, even attacks like presented degraded the control law and dynamic response, which can be catastrophic to the system.

These faults may be maliciously explored to cause serious consequences for security. More results need to be analyzed in NCS, closed loops and other attacks.

The results obtained for this case suggest that: 1) one attack at macrotick caused de-synchronization at logical clocks, but keep the microticks stable; 2) the clock synchronization, the control and the response of dynamic systems can be degraded; 3) new analytical metrics techniques are needed to avoid and constructed better security policies; and 4) the countermeasures and a cyber-security policy are important to avoid the maliciously attacks at the local clocks and hence to avoid faults and defects at control systems.

Suggestions for Countermeasures

Based on these conclusions, to reach the required security goals, we suggest installing various countermeasures on various levels as:

- Firewalls to protect the transmission media;
- FDIR means to Detect an error in the macrotick, Isolate and Identify an error, and Reconfigure the architecture to avoid this attack;
- New time synchronization algorithms to identify and prevent any abrupt change in macroticks at instant, and that minimize a time de-synchronization;
- The attack factor may be help the designers to analyze the weakness of time synchronization;
- New metrics are required to better analysis;
- Introduction of real time clock, as a GPS, to provide a real global time to increase a time synchronization reliability;
- The use of microtick to establish a constraint to macrotick changes. It could be a solution to avoid a huge drift of macrotick.

ACKNOWLEDGMENTS

Both authors thank the INPE, its Course and Division of Space Mechanics and Control for supporting them during the writing of this paper. The first author acknowledges Dr. Jairo C. Amaral for the help during the writing of this paper, and CNPQ financial support during his Ph.D. Program and the writing of this paper.

REFERENCES

[1] SAE, "Certification Considerations for Highly-Integrated or Complex Aircraft Systems," Aerospace Recommended Practice ARP-4754a, SAE, Dec. 2010.

[2] J. A. Stankovic, "Misconceptions about Real-Time Computing: a Serious Problem For Next-Generation Systems," IEEE Computer Society, vol.21, no.10, pp.10,19, Oct. 1988, doi: 10.1109/2.7053.

[3] Winterfeld, Steve, and Jason Andress. The Basics of Cyber Warfare: Understanding the Fundamentals of Cyber Warfare in Theory and Practice. Newnes, 2012.

[4] E. M. Oliveira Junior, M. L. O. Souza, " A Brief Comparison of Security Aspects of Time Synchronization in Networked Control Systems using CSMA/CD versus TDMA Protocols", Proceedings of XIV SIGE, São José dos Campos, Brazil, Sept. 2012.

[5] Blanke, Mogens, Marcel Staroswiecki, and N. Eva Wu. "Concepts and methods in fault-tolerant control." American Control Conference, 2001. Proceedings of the 2001. Vol. 4. IEEE, 2001.

[6] E. M. Oliveira Junior, M. L. O. Souza, " An Overview of Security Aspects of the PTP Algorithm and their Effects in Networked Control Systems with TDMA under Time Discontinuity Faults ", Proceedings of XV SIGE, São José dos Campos, Brazil, Sept. 2013.

[7] Ganeriwal, Saurabh, et al. "Secure time synchronization in sensor networks." ACM Transactions on Information and System Security (TISSEC) 11.4 (2008).

[8] Kopetz, H. Real time systems: design principles for distributed embedded applications. Norwell, MA, USA : Kluwer Academic Publishers, 1997.

[9] G. Gaderer, S. Rinaldi, N. Kero, "Master Failures in the Precision Time Protocol," Precision Clock Synchronization for Measurement, Control and Communication, 2008. ISPCS 2008. IEEE International Symposium on , vol., no., pp.59,64, 22-26 Sept. 2008, doi: 10.1109/ISPCS.2008.4659214.

[10] H. Kopetz, W. Ochsenreiter, "Clock Synchronization in Distributed Real-Time Systems," Computers, IEEE Transactions on , vol.C-36, no.8, pp.933,940, Aug. 1987.

[11] Kopetz, H., Ademaj, A., and Hanzlik, A. "Combination of clock-state and clock-rate correction in fault-tolerant distributed systems", Real-Time Systems, 33(1-3), 139-173, 2006.