I2MTC 2009 - International Instrumentation and Measurement
Technology Conference
Singapore, 5-7 May 2009

Probabilistic Model For Clock Synchronization Of Cascaded Network Elements

Chongning Na, Dragan Obradovic, Ruxandra Lupas Scheiterer
Siemens AG, Corporate Technology,
Information and Communications
Munich, Germany
{na.chongning.ext, dragan.obradovic, ruxandra.scheiterer}@siemens.com

***Abstract*—Precision Time Protocol (PTP) synchronizes clocks of networked elements by exchanging messages containing precise time-stamps. A master clock is carefully chosen to provide the reference clock to the rest elements in the network, called slaves. Using the time-stamps, slave element learns the relation between its own clock and the master clock so that it can synchronize its time to the reference time. Uncertainties, e.g., random stamping and quantization errors, affect the synchronization precision. This paper presents a probabilistic state-space model which quantifies the uncertainties and represents the relation between system variables. Estimation of hidden variables, i.e. the system states, is carried out by using Kalman filter. The performance of this approach is verified by numerical results.**

Keywords-clock synchronization; Precision Time Protocol (PTP); probabilistic model; state-space model; Kalman filter

I. Introduction

Nowadays, Ethernet is replacing the traditional dedicated buses to provide the networking infrastructure in many industry automation fields. Many real time applications require the networked clocks to be well synchronized. The standard Network Time Protocol (NTP)[1], [2] provides synchronization accuracy at the millisecond level, which is appropriate for processes that are not time critical. In the applications such as base station synchronization or motion control, a synchronization precision of base station synchronization or motion control is mandatory in order to let the system work properly. The Precision Time Protocol (PTP), delivered by the IEEE 1588 standard [3] provides an appropriate Ethernet synchronization protocol. It was enhanced by the transparent clock (TC) concept, introduced in [4], which has been adopted in the new version of IEEE 1588 published in 2008. After running the "Best Master Algorithm", which determines the so-called "*master*" clock, messages carrying precise timing information are periodically transmitted by the master and propagated by the so-called "*slave*" clocks after acquiring and updating the contained timing information. Intermediate bridges have to be "IEEE-1588-conform", i.e. are network components with known delay.

Factors that affect the synchronization quality achievable by PTP include the stability of oscillators, the resolution and precision of time stamping the message, the frequency of sending synchronization messages, and the propagation delay variation caused by the jitter in the intermediate elements. Some analytical work has been presented in [5][6] to show the influence of these factors on the synchronization accuracy. It can be seen from the analytical results that stamping imprecision, including quantization error and stamping jitters, have very adverse effects because the errors introduced by different elements accumulate along the network. Current implementation reduces uncertainties by averaging the results from several observations which is not the optimal solution. This paper uses a probabilistic model to present the synchronization problem where probabilities are used to model the uncertainties. On the other hand, each clock is a dynamic system. Synchronization can be understood as learning the relations between different system variables based on discrete observations, i.e., time-stamps. A state space model can be used to describe the relation between different clocks. Then we estimate the hidden state variables in this model based on the observations and the probabilistic relationships between variables. All the random variables involved in the model can be modeled by Gaussian distributions. In this case, we can solve the estimation problem using Kalman filter. Standard synchronization algorithm doesn't exploit the underlying model. The performance is optimal. In this paper, we will compare the performance between standard method and Kalman filter method.

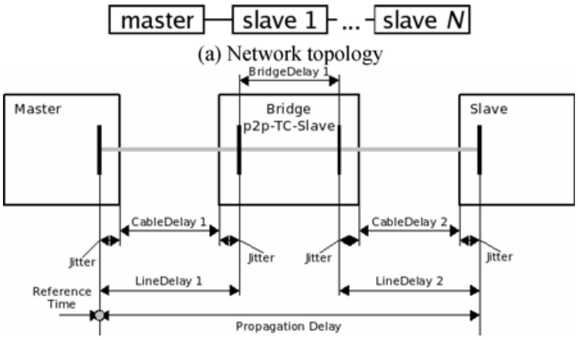
The whole paper is organized as follows. Section 2 introduces the IEEE 1588 peer-to-peer system model analyzed in this paper and briefly describes the PTP protocol, including the standard synchronization algorithm. Section 3 derives the state-space model for the networked clock synchronization. Simulation results are shown in section 4 to verify the performance of the state-space model and the Kalman filter. Finally, Section 5 concludes the paper.

II. System Model and Precision Time Protocol

A. System Model

Fig. 1 shows a system with $N+1$ cascaded elements connected in a line topology. The PTP has a master/slave structure. $N+1$ elements are connected one by one to form a network with a line topology. The first element is the time source, also called (grand)master, which provides the reference time to the rest N elements, called slave elements.

978-1-4244-3353-7/09/$25.00 ©2009 IEEE



(a) Network topology

(b) System parameters

Fig. 1. System model

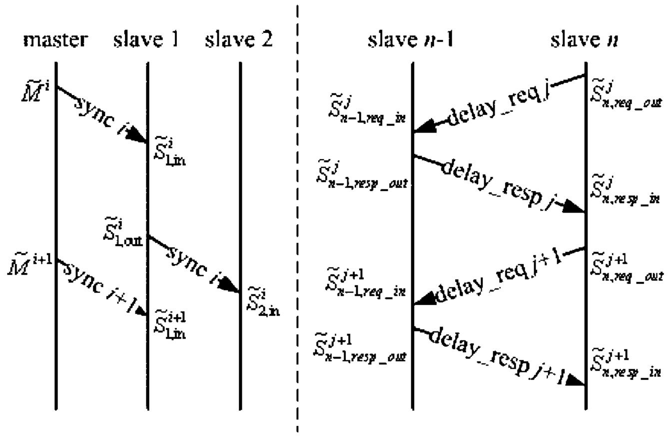


Fig. 2. PTP Messages

B. Messages in Precision Time Protocol

Fig. 2 illustrates the messages defined in PTP for the time synchronization. The master element periodically sends Sync messages which carry the (time)counter state of the master clock M^i, stamped at the sending time, and are propagated along the network. Quantities, certain or not, linked with the Sync message transmitted by the master at time t_i are labeled by the superscript i. Upon the reception of a Sync message, a slave, e.g. slave n, records according to its own clock the reception time S_n^i. Each time a time-stamp is read, a jitter ξ of known distribution is incurred, e.g. due to the quantizing effect of having to wait for the next rising edge of the logic circuitry. A time labeled by S_n (resp. M) means "measured in the local time of slave n (resp. master time)"; a tilde on a symbol means measurement corrupted by jitter, e.g. $\tilde{S}_n^i = S_n^i + \xi_n^i$; a hat on a symbol means "estimate".

The *line delay* LD_n^i, is the propagation time between the n^{th} slave and its uplink element, and is estimated by using the "line delay estimation process". The Sync message is forwarded after a *bridge delay* BD_n^i, which is recorded at each slave as the difference of the times stamped at reception and forwarding. Slave n forwards the Sync message to the next slave. An estimate of the master counter state at the time of forwarding is transmitted to slave $n+1$ for its own estimation of master time.

Time intervals measured by two different clocks will be called "*skewed*". To be able to add or subtract them from each other they have to be converted to the same time basis. The *rate compensation factor* (RCF, also "rate ratio") is defined as the frequency ratio of two clocks. We use $RCF_{X/Y}$ to denote the estimated frequency ratio between X and Y, i.e. ideally $RCF_{X/Y} = f_X / f_Y$.

C. Delay Estimation

The bridge delay can be estimated by:

$$\hat{S}_n(BD_n^i) = \tilde{S}_{n,\text{out}}^i - \tilde{S}_{n,\text{in}}^i \tag{1}$$

The estimation of the line delay to the predecessor is shown on the right in Fig. 2; j indexes the line delay computation. This process uses 4 time-stamps: node n (the requestor) sends a request message to node $n-1$ and records its time of departure, $\tilde{S}_{n,req\_out}^j$. Node $n-1$ (the responder) reports the two time-stamps of receiving the request message and transmitting the reply: $\tilde{S}_{n-1,req\_in}^j$ and $\tilde{S}_{n-1,resp\_out}^j$. The *responder delay* of node $n-1$ is RD_{n-1}^j in absolute time, and is in local time:

$$\hat{S}_{n-1,respD}^j := \tilde{S}_{n-1,resp\_out}^j - \tilde{S}_{n-1,req\_in}^j \tag{2}$$

Node n records the time, $\tilde{S}_{n,resp\_in}^j$ (4th), of receiving the desired reply, after a *requestor delay* in node n time of:

$$\hat{S}_{n,reqD}^j := \tilde{S}_{n,resp\_in}^j - \tilde{S}_{n,req\_out}^j \tag{3}$$

To be able to subtract the skewed time intervals of (2) and (3), each element maintains an "RCF peer" estimate, i.e. frequency ratio estimate to its predecessor, estimated via:

$$RCF_{S_n/S_{n-1}}^j = \frac{\tilde{S}_{n,req\_out}^j - \tilde{S}_{n,req\_out}^{j-1}}{\tilde{S}_{n-1,req\_in}^j - \tilde{S}_{n-1,req\_in}^{j-1}} \tag{4}$$

Then the line delay can be estimated as:

$$\hat{S}_n(LD_n^j) = \frac{\tilde{S}_{n,reqD}^j - \tilde{S}_{n-1,respD}^j \cdot RCF_{S_n/S_{n-1}}}{2} \tag{5}$$

Usually several successive line delay estimates are averaged. The result of the averaged line delay estimates is the constant cable delay plus the mean of several i.i.d. random variables ξ^j. According to the *Central Limit Theorem*, for several summands this is well approximated by the cable delay plus additive Gaussian noise:

$$\hat{S}_n(LD_n^j) = S_n(CD) + \nu_n^j \tag{6}$$

D. Standard Synchronization Algorithm

Upon the reception of a Sync message, slave n updates its estimation of the rate offset to the master via the master counter estimates in two Sync messages and the local counter values at arrival of these messages:

$$RCF^i_{M/S_n} = \frac{\hat{M}^i_{n-1} - \hat{M}^{i-1}_{n-1}}{\tilde{S}^i_n - \tilde{S}^{i-1}_n} \tag{7}$$

Slave n then translates to master time the delay measured in local time, by multiplying it by RCF^i_{M/S_n}, and updates the received estimated master counter value $\hat{M}^i_{n-1}$ according to:

$$\begin{cases} \hat{M}^i_n = \hat{M}^i_n + \left(\hat{S}_n(LD^j_n) + \hat{S}_n(BD^i_n)\right)\cdot RCF^i_{M/S_n} & \text{for } n > 1 \\ \hat{M}^i_1 = \tilde{M}^i \end{cases} \tag{8}$$

where $\hat{S}_n(BD^i_n)$ is estimated by (1) and $\hat{S}_n(LD^j_n)$ is estimated by (6).

The synchronization precision of standard synchronization algorithm has been analyzed in some of our previous works. Some analytical work has been presented in [5][6] to show how different factors affect the synchronization performance. An enhancement was proposed in [7] to minimize the influence of frequency drift. Another dominant error contributor, i.e., stamping jitter, is still to be dealt with in order to improve the synchronization precision. The next section will model the uncertainties by a probabilistic state space model and use Kalman filter to estimate the hidden state variables, i.e., the master counter values. In the simulation section, we will compare the performance of these two algorithms.

III. Probabilistic State Space Model for Clock Synchronization

Based on the time stamps delivered by the PTP messages, each slave builds a relationship between the slave clock and the master clock so that for any given slave counter value, the estimated master counter value at that time point can be calculated. To build such a relationship, we define a hidden state variable x^i_n that is the true master counter state corresponding to the slave time-stamp $\tilde{S}^i_n$. Figure 3 illustrates the relationship between the variables. $x^i_{n,\text{in}}$ and $x^i_{n,\text{out}}$ are defined as the true master counter values that correspond to the slave time-stamps $\tilde{S}^i_{n,\text{in}}$ and $\tilde{S}^i_{n,\text{out}}$, i.e., the time-stamps generated at the reception and forwarding of the i^{th} sync message. $M^i_{n,\text{in}}$ and $M^i_{n,\text{out}}$ are defined as the true master counter values that correspond to the true slave times $S^i_{n,\text{in}}$ and $S^i_{n,\text{out}}$. In the following sections, we will introduce a state space model from master to slave 1, and from slave n to slave $n+1$.

A. State-Space Model: Master to Slave 1

It can be easily observed that $x^i_{n,\text{in}} - x^{i-1}_{n,\text{in}}$ and $\tilde{S}^i_{n,\text{in}} - \tilde{S}^{i-1}_{n,\text{in}}$ are the same time interval measured by different clocks, i.e., master clock and slave clock. They are related by the frequency ratio of the two clocks, that is:

$$x^i_{n,\text{in}} - x^{i-1}_{n,\text{in}} = \left(\tilde{S}^i_{n,\text{in}} - \tilde{S}^{i-1}_{n,\text{in}}\right)\cdot R_{M/S_n} \tag{9}$$

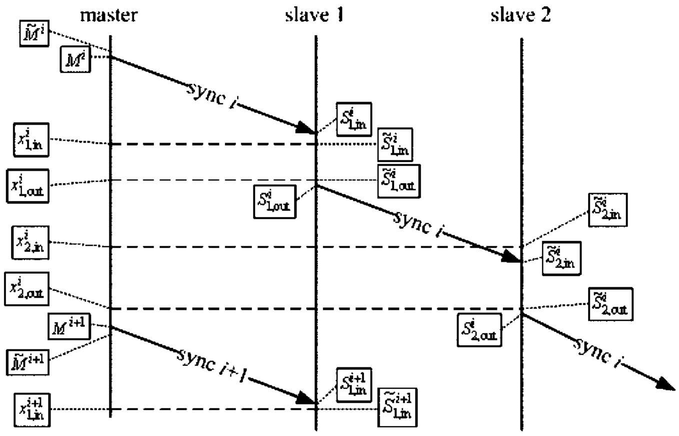


Fig. 3. Observed and hidden variables

where R_{M/S_n} is the true frequency ratio between master and slave clocks. We assume that master and slave clocks have different frequencies, but they are constant.

The frequency ratio between master clock and slave 1 can be estimated by:

$$\begin{aligned} RCF^i_{M/S_1} &= \frac{\tilde{M}^i - \tilde{M}^{i-1}}{\tilde{S}^i_{1,\text{in}} - \tilde{S}^{i-1}_{1,\text{in}}} = \frac{M^i - M^{i-1} + \xi^i_M - \xi^{i-1}_M}{\tilde{S}^i_{1,\text{in}} - \tilde{S}^{i-1}_{1,\text{in}}} \\ &= \frac{\left(S^i_{1,\text{in}} - S^{i-1}_{1,\text{in}}\right)\cdot R_{M/S_1} + \xi^i_M - \xi^{i-1}_M}{\tilde{S}^i_{1,\text{in}} - \tilde{S}^{i-1}_{1,\text{in}}} \\ &= \frac{\left(\tilde{S}^i_{1,\text{in}} - \tilde{S}^{i-1}_{1,\text{in}} - \xi^i_{S_{1,\text{in}}} + \xi^{i-1}_{S_{1,\text{in}}}\right)\cdot R_{M/S_1} + \xi^i_M - \xi^{i-1}_M}{\tilde{S}^i_{1,\text{in}} - \tilde{S}^{i-1}_{1,\text{in}}} \\ &\approx R_{M/S_1} + \frac{\xi^i_M - \xi^{i-1}_M - \xi^i_{S_{1,\text{in}}} + \xi^{i-1}_{S_{1,\text{in}}}}{\tilde{S}^i_{1,\text{in}} - \tilde{S}^{i-1}_{1,\text{in}}} \end{aligned} \tag{10}$$

where the approximation made in the 4[th] line in (10) is based on the fact that the stamping error is small and the frequency ratio is very close to 1. Since it is the sum of 4 i.i.d. RVs, we use the Gaussian RV $\eta^i \sim N(0, Q)$ to approximate the last term in (10). So (10) can be rewritten as:

$$RCF^i_{M/S_1} = R_{M/S_1} + \frac{\eta^i}{\tilde{S}^i_{1,\text{in}} - \tilde{S}^{i-1}_{1,\text{in}}} \tag{11}$$

Inserting (11) into (9) and reformulating, we obtain:

$$x^i_{1,\text{in}} = x^{i-1}_{1,\text{in}} + \left(\tilde{S}^i_{1,\text{in}} - \tilde{S}^{i-1}_{1,\text{in}}\right)\cdot RCF^i_{M/S_1} - \eta^i_1 \tag{12}$$

This constitutes ***the state transition model.***

$x^i_{1,\text{in}}$ and M^i are related by the line delay:

$$\begin{aligned} \tilde{M}^i - \xi^i_M &\approx x^i_{1,in} - \xi^i_{S_1,\text{in}} - \left(\hat{S}_1(CD^j_1) - v^j_1\right)\cdot RCF^i_{M/S_1} \\ &= x^i_{1,in} + \hat{S}_1(CD^j_1)\cdot RCF^i_{M/S_1} - \xi^i_{S_1,\text{in}} + RCF^i_{M/S_1}\cdot v^j_1 \end{aligned} \tag{13}$$

The approximation made in the first line is based on the fact that the cable delay is usually very small so that it attenuates the error made in the RCF estimation. We can combine the last three terms in (13) and approximate it with a Gaussian RV $\varepsilon^j \sim N(0,U)$. So (13) can be rewritten as:

$$\widetilde{M}^i = x^i_{1,in} + \hat{S}_1(CD^j_1)\cdot RCF^i_{M/S_1} + \varepsilon^i_1 \tag{14}$$

This constitutes ***the observation model***.

B. State-Space Model: slave n to slave $n+1$

Like in (9) and (10), slave n has a ***state transition model*** that can be expressed by:

$$x^i_{n+1,\text{in}} = x^{i-1}_{n+1,\text{in}} + \left(\widetilde{S}^i_{n+1,\text{in}} - \widetilde{S}^{i-1}_{n+1,\text{in}}\right)\cdot RCF^i_{M/S_{n+1}} - \eta^i_{n+1} \tag{15}$$

In this case, the RCF is computed by:

$$RCF^i_{M/S_{n+1}} = \frac{\hat{x}^i_{n,\text{out}} - \hat{x}^{i-1}_{n,\text{out}}}{\widetilde{S}^i_{n+1,\text{in}} - \widetilde{S}^{i-1}_{n+1,\text{in}}} \tag{16}$$

where $\hat{x}^i_{n,\text{out}}$ is estimated at slave n by:

$$\hat{x}^i_{n,\text{out}} = \hat{x}^i_{n,\text{in}} + \hat{S}_n(BD^i_n)\cdot RCF^i_{M/S_n} \tag{17}$$

The results is transmitted to slave $n+1$. Let's assume that the error made in the estimation is $\gamma^i_{n,\text{out}}$, i.e.,

$$M^i_{n,\text{out}} = \hat{x}^i_{n,\text{out}} + \gamma^i_{n,\text{out}} \tag{18}$$

Inserting (18) into (16), we obtain:

$$\begin{aligned} RCF^i_{M/S_{n+1}} &= \frac{\hat{x}^i_{n,\text{out}} - \hat{x}^{i-1}_{n,\text{out}}}{\widetilde{S}^i_{n+1,\text{in}} - \widetilde{S}^{i-1}_{n+1,\text{in}}} = \frac{M^i_{n,\text{out}} - M^{i-1}_{n,\text{out}} - \gamma^i_{n,\text{out}} + \gamma^{i-1}_{n,\text{out}}}{\widetilde{S}^i_{n+1,\text{in}} - \widetilde{S}^{i-1}_{n+1,\text{in}}} \\ &= R_{M/S_{n+1}} + \frac{\xi^i_{S_{n,\text{out}}} - \xi^{i-1}_{S_{n,\text{out}}} + \xi^i_{S_{n+1,\text{in}}} - \xi^{i-1}_{S_{n+1,\text{in}}} - \gamma^i_{n,\text{out}} + \gamma^{i-1}_{n,\text{out}}}{\widetilde{S}^i_{n+1,\text{in}} - \widetilde{S}^{i-1}_{n+1,\text{in}}} \\ &= R_{M/S_{n+1}} + \frac{\eta^i_{n+1,\text{in}}}{\widetilde{S}^i_{n+1,\text{in}} - \widetilde{S}^{i-1}_{n+1,\text{in}}} \end{aligned} \tag{19}$$

Again, in the last line of (19), we use a Gaussian RV $\eta^i_{n+1,\text{in}}$ to approximate the sum of the random errors.

The line delay connects the state variable $x^i_{n+1,\text{in}}$ with the estimate $\hat{x}^i_{n,\text{out}}$:

$$\hat{x}^i_{n,\text{out}} = x^i_{n+1,\text{in}} - \hat{S}_{n+1}(CD^j_{n+1})\cdot R_{M/S_{n+1}} + \varepsilon^i_{n+1} \tag{20}$$

where ε^i_n is a Gaussian RV modeling all the uncertainties. (20) gives ***the observation model*** at slave $n+1$.

C. State-Space Model: Graphical Representation

To summarize the equations for the state space model and for a better illustration, we use a graphical model to represent the whole probabilistic model, which is shown in Fig. 4. In Fig. 4, circles represent variables, either observed (shaded circles) or hidden (white circles). Squares represent the relationship between the variables that these squares are connected to. Numbers in the squares give the equation numbers indicating via which equations those variables are connected. Based on the state space model presented in Fig. 4, the state variable $x^i_{n,\text{in}}$ can be estimated by different methods. This paper presents results obtained by Kalman filter.

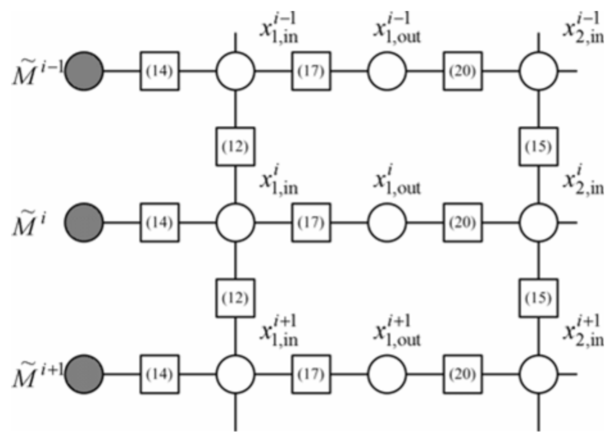


Fig. 4. Graphical representation of the state space model

D. Comparision: Standard Algorithm v.s. Kalman Filtering

In the standard synchronization algorithm, only the estimates of the master counter value, i.e., the results of (8) will be transmitted to the next slave elements whereas in Kalman filtering approach, the estimation uncertainty, encapsulated in the estimation variance will also be transmitted. On the other hand, the state space model also uses state-transition equation to describe the dynamics of the clock, which is not considered in the standard synchronization algorithm.

IV. SIMULATION RESULTS

In order to verify our new synchronization method, we have implemented the algorithm and simulated the synchronization protocol. Parameters for the simulation are summarized in Table 1.

TABLE I. SIMULATION SETTINGS

Parameter	Value
Quartz precision	50ppm
Cable delay	100ns
Bridge delay	uniform 2ms+[5 125] µs
Interval of Sync message	32ms
Interval of Delay_request	8s
Stamping jitter	uniform [-40 40] ns
Jitter in line delay	uniform [-40 40] ns
Number of line delay averaging	8

Synchronization errors, i.e. the difference between the estimated master time and the true master time are shown in Fig. 5. For more remote slaves it takes time for the algorithm to converge. Fig. 6 shows the same results but using a different x and y-axis scale. It can be observed that the error oscillates around zero due to the uncertainties in the time stamping. The magnitude of the oscillation increases along the line since the latter slave element estimates the master time based on the estimates of the previous slaves so that error propagates. For a comparison, we also simulated the standard synchronization algorithm using the same simulation settings. The results are shown in Fig. 7 and Fig. 8. From the result we can see that Kalman filtering based synchronization algorithm converges faster than the standard synchronization algorithm. We can also see that by correctly modeling the dynamics of the clock and the uncertainties in the time-stamping, the synchronization precision is improved.

V. Conclusion and Discussions

This paper presents a state-space model for time synchronization, using the noisy time stamps provided by the PTP protocol. Performance of the novel algorithm is evaluated through simulation results. Comparing the simulation results, we can see that Kalman filtering based synchronization method improves the standard synchronization algorithm in both convergence time and synchronization precision. In this model, the RCF is not defined as a hidden state variable in the state space model. It is not estimated by Kalman filter but another process. The estimate of RCF is used as a parameter in Kalman filter. We are working on a new state space model which defines RCF as a hidden state variable. So in this model, Kalman filter will jointly estimate $x^{i}_{n,\text{in}}$ and RCF.

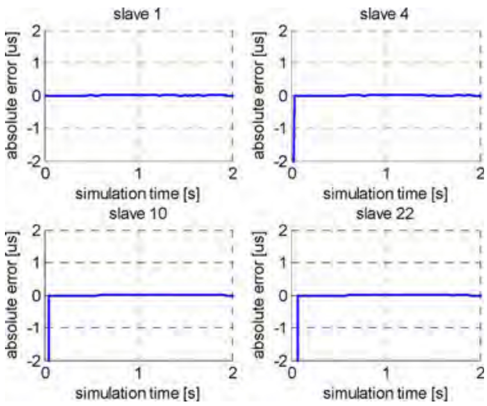


Fig. 5. Synchronization error of Kalman filtering

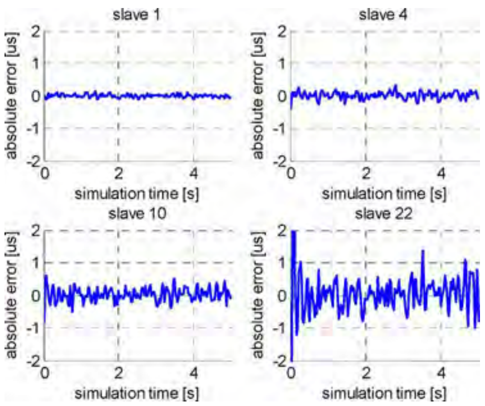


Fig. 6. Synchronization error of Kalman filtering

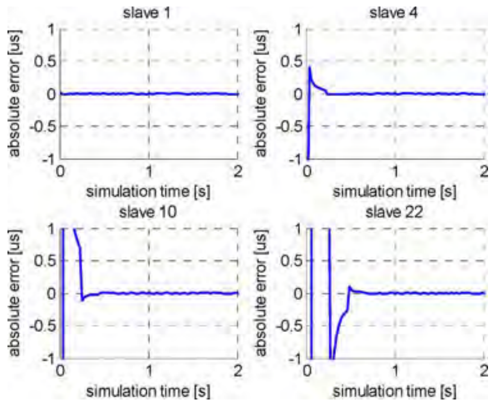


Fig. 7. Synchronization error of standard synchronization algorithm

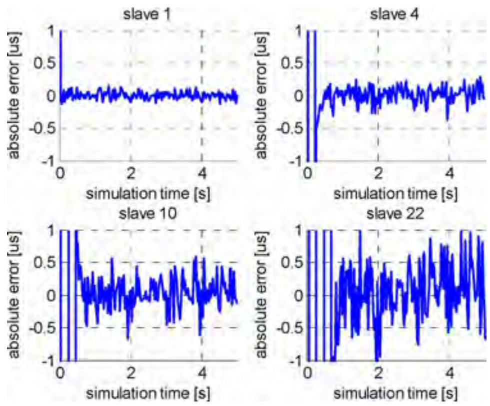


Fig. 8. Synchronization error of standard synchronization algorithm

References

[1] D. L. Mills, "Internet time synchronization: The network time protocol," Network Working Group Request for Comments, 1989.

[2] D. L. Mills, "Precision synchronization of computer network clocks," *ACM SIGCOMM Computer Communication Review*, Vol. 24, pp. 28-43, 1994.

[3] IEEE.(2002). IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems. IEEE, New York.

[4] J. Jasperneite, K. Shehab, K. Weber. Enhancements to the time synchronization standard IEEE-1588 for a system of cascaded bridges. In: Proc. of WFCS 2004, pp. 239-44.

[5] C. Na, D. Obradovic, R.L. Scheiterer, G. Steindl and F.J. Goetz, "Synchronization Performance of the Precision Time Protocol," Proc. of ISPCS 2007, Vienna, 2007.

[6] R.L. Scheiterer, C. Na, D. Obradovic, G. Steindl and F. J. Goetz, "Synchronization performance of the Precision Time Protocol in industrial automation networks," accepted by the ISPCS07 Special Issue of the IEEE Transactions on Instrumentation and Measurement, 2008.

[7] C. Na, D. Obradovic, R.L. Scheiterer, G. Steindl and F. J. Goetz, "Ehancement of the precision time protocol in automation networks with a line topology," Proc. Of 17th IFAC World Congress, Seoul, Korea 2008.