Synchronization Performance of the Precision Time Protocol in Industrial Automation Networks

Ruxandra Lupas Scheiterer, Chongning Na, *Student Member, IEEE*,
Dragan Obradovic, *Member, IEEE*, and Günter Steindl

***Abstract*—This paper analyzes the factors that affect the synchronization performance in peer-to-peer precision time protocol (PTP). We first study the influence of frequency drift in the absence of jitter and compare the gravity of the master drift with that of the slave drift. Then, we study the influence of jitter under the assumption of constant frequencies and the effect of averaging. The analytic formulas provide a theoretical ground for understanding the simulation results, some of which are presented, as well as the guidelines for choosing both system and control parameters.**

***Index Terms*—Frequency drift, IEEE 1588, jitter, master, precision time protocol (PTP) Version 2, rate-compensation factor (RCF), slave, synchronization error, synchronization precision, transparent clock (TC).**

I. Introduction

ETHERNET [1], due to its cheap cabling and infrastructure costs, high bandwidth, efficient switching technology, and better interoperability, has been adopted in various areas to provide the basic networking solution. Many Ethernet-based applications require the networked clocks to be precisely synchronized. Typical examples include the synchronization of base stations for handover or interference cancellation [2] in telecommunication networks, distribution of audio/video streams over Ethernet-based networks [3], and motion control in industrial Ethernet [4]. Standard network time protocol (NTP) [5], [6] synchronization over Ethernet provides a time accuracy at the millisecond level, which is enough for processes that are not time critical. However, in many applications, base station synchronization or motion control, where only submicrosecond-level synchronization errors are allowed (the so-called isochronous mode), a more accurate solution is needed. The precision time protocol (PTP), which was delivered by the IEEE 1588 standard [7] and published in 2002, constitutes a promising Ethernet synchronization protocol in which messages carrying precise timing information are propagated in the network to synchronize the slave clocks to a master clock.

Boundary clocks adjust their own clock to the master clock and then serve as masters for the next network segment. The authors of [8] introduced the transparent clock (TC) concept, in which intermediate bridges are treated as network components with known delay. By doing this, the synchronization at the time client is not dependent on the control loop design in the intermediate bridges. The TC concept has been adopted in the new version of IEEE 1588 published in 2007 (http://ieee1588.nist.gov/). The IEEE 1588-2008, also known as 1588 Version 2, was approved by the IEEE on March 27, 2008.

The current state of the art is to guarantee a synchronization precision of 1 μs for topologies with 30 consecutive slaves. To expand this limit, it is important to study the factors that influence the quality of the synchronization process and to work on minimizing their effect. Industrial environments are such that unpredictable and independent temperature changes at each node are commonly encountered, causing short-term frequency drifts, unless precluded by expensive temperature-compensated (TCXO) or oven-controlled (OCXO) crystal oscillators. Factors that affect the synchronization quality achievable by PTP include the stability of oscillators, the resolution of time stamping the message, the frequency of sending synchronization messages, and the propagation delay variation caused by the jitter in the intermediate elements. Previous work on the subject includes [10], which derives the synchronization errors due to jitters and master frequency drift; [11] and [12], where improved algorithms are proposed; [13], where the line delay estimation error due to clock frequency drift is obtained; and [14], which derives the synchronization error due to slave frequency drift.

This paper summarizes the work in [10]–[14], adding analytic results on the effect of averaging, comparative simulation results, and guidelines for choosing both system and control parameters when applying PTP.

This paper is organized as follows. Section II introduces the 1588 peer-to-peer system model and describes the PTP. An analysis of the influence of frequency drifts is presented in Section III and of jitters in Section IV. Consequences are discussed in Section V and strengthened by simulation results in Section VI.

II. Description of PTP With TCs

Since the standard does not specify the details, this section introduces the system model and notation. Fig. 1 shows a system with $N+1$ cascaded elements connected in a line topology. The PTP has a master/slave structure.

The time server, called grand(master), provides the reference time to the other elements, called slaves, via time-aware bridges

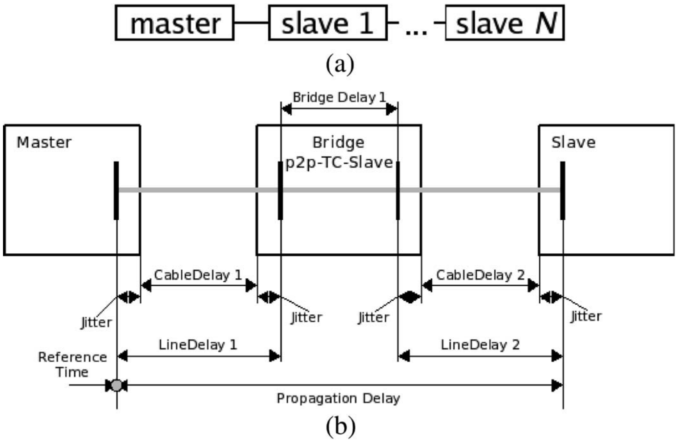
Manuscript received March 18, 2008; revised September 26, 2008. First published April 14, 2009; current version published May 13, 2009. The Associate Editor coordinating the review process for this paper was Dr. Georg Gaderer.

R. L. Scheiterer, C. Na, and D. Obradovic are with Corporate Technology, Information and Communications, Siemens AG, 81730 Munich, Germany (e-mail: ruxandra.scheiterer@siemens.com; na.chongning.ext@siemens.com; dragan.obradovic@siemens.com).

G. Steindl is with Automation & Drives, Industrial Automation Systems, Siemens AG, 92220 Amberg, Germany (e-mail: guenter.steindl@siemens.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIM.2009.2013655



(a)

(b)

Fig. 1. System model. (a) Network topology. (b) System parameters.

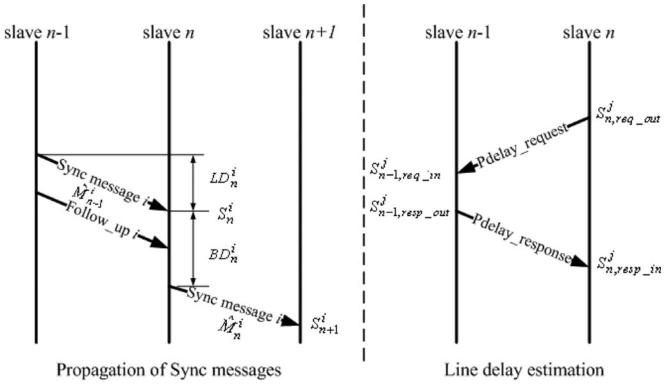


Fig. 2. PTP with TCs.

(TCs). The master sends Sync messages every T_{sync} seconds, which carry the counter state of the master clock M^i, stamped at sending, 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 superscript i. If f is the clock frequency, a delay D is measured by the local clock as the counter increase $D \cdot f$; or as an integral thereof, if the frequency drifts. The *line delay* LD^i_n is the propagation time between the nth slave and its uplink element, and is estimated by using the so-called "line delay estimation process" of (6). Each time a time stamp is read, a jitter of known distribution is incurred. The Sync message is forwarded after a *bridge delay* BD^i_n, which is recorded at each slave as the difference of the times stamped at reception and forwarding. A time labeled by S_n (respectively, M) means "measured in the local time of slave n (respectively, master time)"; a hat on a symbol means "estimate." For example, $\hat{S}_n(BD^i_n)$ means "the bridge delay that affects the ith Sync message at slave n, estimated in terms of the own local free-running clock." At slave n, we define LB^i_n as the sum of line delay plus bridge delay of Sync message i, and $\delta^{i,n}_{LB} = LB^i_n - LB^{i-1}_n$ as the difference between the LB values that affected Sync messages i and $i-1$.

The 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. To this end, each slave determines its frequency offset to the master. The *rate compensation factor* (RCF, which is also the "rate ratio" [9], [12]) is defined as the frequency ratio of two clocks. We use $RCF_{X/Y}$ to denote the frequency ratio between X and Y, i.e., ideally $RCF_{X/Y} = f_X/f_Y$. The rate offset to the master RCF_n can be estimated via the master counter estimates $\hat{M}$ in two Sync messages whose transmission times are separated by an interval T (equal to T_{sync}, or a multiple thereof) and the local counter values S at arrival of these messages at slave n, i.e.,

$$RCF_n \equiv RCF_{M/S_n} = \frac{\hat{M}^i_{n-1} - \hat{M}^{i-1}_{n-1}}{S^i_n - S^{i-1}_n}. \tag{1}$$

Fig. 2 illustrates the pillars of time synchronization: timing propagation, and line delay estimation. Upon reception of Sync message i, slave n hands an estimate of the master time at the time of reception to the control loop of its controlled clock. In addition, he acts as TC by passing on the received (estimated) master counter value $\hat{M}^i_{n-1}$ augmented by its local delay, i.e., the sum of its (skewed) line and bridge delays translated to master time by multiplication with RCF_n:

$$\begin{cases} \hat{M}^i_n = \hat{M}^i_{n-1} + \left(\hat{S}_n(LD^i_n) + \hat{S}_n(BD^i_n)\right) \cdot RCF^i_n \\ \hat{M}^i_0 = M^i = M(t_i). \end{cases} \tag{2}$$

This equation represents the "support" for an extrapolation process that advances the controlled local clock until the arrival of new information. The last ingredient necessary for the update of (2) is the estimation of the line delay to the predecessor, shown on the right in Fig. 2; j indexes the line delay computation. This process uses four time stamps: with periodicity R, node n (the requestor) sends a request message to node $n-1$ and records its time of departure S^j_{n,req_out}. Node $n-1$ (the responder) reports the two time stamps of receiving the request message and transmitting the reply S^j_{n-1,req_in} and $S^j_{n-1,resp\_out}$. The *responder delay* of node $n-1$ is RD^j_{n-1} in absolute time and is, in local time

$$S^j_{n-1,respD} := S^j_{n-1,resp\_out} - S^j_{n-1,req\_in}. \tag{3}$$

Node n records the time $S^j_{n,resp\_in}$ of receiving the desired reply after a *requestor delay* in node n time of

$$S^j_{n,reqD} := S^j_{n,resp\_in} - S^j_{n,req\_out}. \tag{4}$$

To be able to subtract the skewed time intervals of (3) and (4), each element maintains an "RCF peer" estimate, i.e., the frequency ratio estimate to its predecessor, that is estimated via

$$RCF^j_{S_n/S_{n-1}} = \frac{S^j_{n,req\_out} - S^{j-1}_{n,req\_out}}{S^j_{n-1,req\_in} - S^{j-1}_{n-1,req\_in}}. \tag{5}$$

Then, the line delay can be estimated as

$$\hat{S}_n\left(LD^j_n\right) = \frac{S^j_{n,reqD} - S^j_{n-1,respD} \cdot RCF_{S_n/S_{n-1}}}{2}. \tag{6}$$

Usually, several successive line delay estimates are averaged. This concludes the description of the synchronization process, whose accuracy is studied in this paper. In our analytic work, we examine each effect in isolation. However, all are simultaneously present in the subsequent simulations. We assume zero delay skew, i.e., direction-independent cable run time. This is a mild idealization since the IEC 61784-5-3 mandates stringent requirements for the delay skew, e.g., for PROFINET, it may not exceed 20 ns/100 m.

III. Analysis of Error Propagation—Part 1: The Effect of Clock Drift in the Absence of Jitter

A. *Scenario Description*

To estimate the master counter value from (2), estimates of the line delay and RCF value need to be available, and the RCF and the master counter estimation processes are tightly intertwined. The clock frequency drifts affect these estimates and influence the synchronization performance. In this section, we quantify these effects and compare the two scenarios where the master frequency stays constant, while the slave frequencies drift, and *vice versa.*

The frequency of all the elements is constant until t_0. Then, the drifting elements' frequencies linearly increase. The short-term linear frequency drifts are typically temperature induced, as shown by the corresponding characteristic curves. For other situations, our analysis is a local first-order approximation.

The frequency of an element k with constant frequency follows:

$$f_k(t_i) = f_k(t_{i-1}) = f_k \tag{7}$$

whereas the frequency of element n whose frequency drifts with slope Δ_n follows:

$$f_n(t_i) = f_n(t_{i-1}) + \Delta_n \cdot (t_i - t_{i-1}), \quad t_i > t_{i-1} > t_0 \tag{8}$$

i.e., $f_n(t_i)/f_n(t_{i-1}) = 1 + \Delta_n \cdot (t_i - t_{i-1})/f_n(t_{i-1})$.

The counter value increase of each element over the time interval (t_{i-1}, t_i) is the integral over the element's frequency. For the constant-frequency element, this results in

$$C_k(t_i) - C_k(t_{i-1}) = f_k \cdot (t_i - t_{i-1}) \tag{9}$$

whereas for drifting element n, it is calculated as

$$\begin{aligned} C_n(t_i)-C_n(t_{i-1}) &= \int_{t_{i-1}}^{t_i} f_n(t) \cdot dt \\ &= f_n(t_{i-1})\cdot(t_i-t_{i-1}) + \frac{\Delta_n}{2}\cdot(t_i-t_{i-1})^2. \end{aligned} \tag{10}$$

Due to the assumed linearity of the frequency change, we can rewrite (10) as the product of the frequency in the middle of the time interval times the length of the interval

$$C_n(t_i) - C_n(t_{i-1}) = f_n\left(\frac{t_i + t_{i-1}}{2}\right) \cdot (t_i - t_{i-1}). \tag{11}$$

For example, the bridge delay is approximately

$$\hat{S}_n\left(BD_n^i\right) = f_n(\text{mid Bridge Delay}) \cdot BD_n^i. \tag{12}$$

B. *Effect of Frequency Drift on the Accuracy of the Line Delay Estimate in the Absence of Jitter*

In [13], the line delay estimate of slave n is derived to be

$$\begin{aligned} \hat{S}_n\left(LD_n^{j(i)}\right) \approx\ & f_n\left(t_i + L_n^i - A_n^i\right) \cdot LD_n^{j(i)} \\ & + RD_{n-1}^{j(i)} \cdot \frac{R + RD_{n-1}^{j(i)}}{4} \cdot (\Delta_n - \Delta_{n-1}) \end{aligned} \tag{13}$$

where L_n^i is the propagation time of the ith Sync message to slave n (which is called *latency*) with $L_0^i = 0$, and

$$L_n^i = \sum_{k=1}^{n-1} LB_k^i + LD_n^i \approx \sum_{k=1}^{n} \left(LD_k^i + BD_k^i\right), \qquad n \geq 1 \tag{14}$$

and A_n^i is the "age" of the line delay computation valid for Sync Message i, i.e., the middle of the last line delay computation interval until the arrival of the current Sync message, upper bounded by the requestor interval R introduced earlier, with (denoting $\delta_L^{i,n} = L_n^i - L_n^{i-1}$)

$$A_n^i - A_n^{i-1} = \begin{cases} T_{\text{sync}} + \delta_L^{i,n}, & \text{if no new line delay estimate} \\ T_{\text{sync}} + \delta_L^{i,n} - R, & \text{else.} \end{cases} \tag{15}$$

C. *Error in RCF and Master-Counter Estimation Due to Master Frequency Drift for Nondrifting Slaves*

For the master, (8) and (11) apply, whereas for the slaves, (7) and (9) apply. The slaves' line delay estimates of (13) become

$$\hat{S}_n\left(LD_n^{j(i)}\right) \approx \begin{cases} f_1 \cdot LD_1^{j(i)} - RD_M^{j(i)} \cdot \frac{R+RD_M^{j(i)}}{4} \cdot \Delta_M & \\ \quad = f_1 \cdot LD_1^{j(i)} - \varepsilon_{LD}^{j(i),1}, & n = 1 \\ f_n \cdot LD_n^{j(i)}, & \text{else.} \end{cases} \tag{16}$$

The RCF value computed by Slave$_1$ from the Sync message transmitted by the master at time t_i is

$$\begin{aligned} RCF_1^i &= \frac{M(t_i) - M(t_{i-1})}{S_1\left(t_i + LD_1^i\right) - S_1\left(t_{i-1} + LD_1^{i-1}\right)} \\ &= \frac{f_M\left(\frac{t_i + t_{i-1}}{2}\right) \cdot (t_i - t_{i-1})}{\left(T + LD_1^i - LD_1^{i-1}\right) \cdot f_1} \\ &= \frac{f_M\left(t_i - \frac{T}{2}\right)}{f_1} \cdot \frac{T}{T + \delta_{LD}^{i,1}}. \end{aligned} \tag{17}$$

Without jitter, $LD_n^i = \text{const}$ and $\delta_{LD}^{i,n} = 0$, hence:

$$RCF_1^i = \frac{f_M\left(t_i - \frac{T}{2}\right)}{f_1}. \tag{18}$$

Slave$_1$ forwards the received Sync message to Slave$_2$ at time $t_i + LB_1^i$ and replaces the received master counter content by its estimate at the time of transmission using (2), i.e.,

$$\hat{M}\Big|_{S_1\text{out}}^{t_i + LB_1^i} = M(t_i) + \left(\hat{S}_1\left(LD_1^{j(i)}\right) + \hat{S}_1\left(BD_1^i\right)\right) \cdot RCF_1^i. \tag{19}$$

Using (16), (12), and (18), we can rewrite this as

$$\begin{aligned} \hat{M}\Big|_{S_1\text{out}}^{t_i + LB_1^i} &= M(t_i) + \left(f_1 \cdot LD_1^{j(i)} - \varepsilon_{LD}^{j,1} + f_1 \cdot BD_1^i\right) \cdot \frac{f_M\left(t_i - \frac{T}{2}\right)}{f_1} \\ &= M(t_i) + LB_1^i \cdot f_M\left(t_i - \frac{T}{2}\right) - \varepsilon_{LD}^{j,1} \cdot \frac{f_M\left(t_i - \frac{T}{2}\right)}{f_1} \\ &\quad + \left(LD_1^{j(i)} - LD_1^i\right) \cdot f_M\left(t_i - \frac{T}{2}\right). \end{aligned} \tag{20}$$

If there is no jitter, $LD_1^{j(i)} = LD_1^i$, and the last term vanishes.

Assuming that the master frequency continues to increase with the same slope, the true value of the master counter at this time is [from (10)]

$$M\Big|^{t_i + LB_1^i} = M(t_i) + f_M(t_i) \cdot LB_1^i + \frac{\Delta_M}{2} \cdot LB_1^{i^2}. \tag{21}$$

Comparing (21) and (20), the error at this time in the TC transmission of Slave$_1$ is [using (8)]

$$\begin{aligned} M-\hat{M}\Big|_{S_1\mathrm{out}}^{t_i+LB_1^i} &= \left[f_M(t_i)-f_M\left(t_i-\frac{T}{2}\right)\right]\cdot LB_1^i \\ &\quad +\frac{\Delta_M}{2}\cdot {LB_1^i}^2+\varepsilon_{LD}^{j,1}\cdot\frac{f_M\left(t_i-\frac{T}{2}\right)}{f_1} \\ &= \frac{\Delta_M}{2}\cdot\left(T\cdot LB_1^i+{LB_1^i}^2\right)+\varepsilon_{LD}^{j,1}\cdot\frac{f_M\left(t_i-\frac{T}{2}\right)}{f_1}. \end{aligned} \tag{22}$$

The RCF value computed by Slave$_2$ from the Sync message transmitted by the master at time t_i, which now contains Slave$_1$'s master counter estimate, is:

$$\begin{aligned} RCF_2^i &= \frac{\hat{M}_{S_1\mathrm{out}}\left(t_i+LB_1^i\right)-\hat{M}_{S_1\mathrm{out}}\left(t_{i-1}+LB_1^{i-1}\right)}{S_2\left(t_i+LB_1^i+LD_2^i\right)-S_2\left(t_{i-1}+LB_1^{i-1}+LD_2^{i-1}\right)} \\ &= \frac{\left\{\begin{array}{l} M(t_i)-M(t_{i-1})+f_M\left(t_i-\frac{T}{2}\right)\cdot\delta_{LB}^{i,1} \\ +\left(f_M\left(t_i-\frac{T}{2}\right)-f_M\left(t_{i-1}-\frac{T}{2}\right)\right)\cdot LB_1^{i-1} \\ +\varepsilon_{LD}^{j(i-1),1}\cdot\frac{f_M\left(t_{i-1}-\frac{T}{2}\right)}{f_1}-\varepsilon_{LD}^{j(i),1}\cdot\frac{f_M\left(t_i-\frac{T}{2}\right)}{f_1} \end{array}\right.}{f_2\cdot\left[t_i+LB_1^i-\left(t_{i-1}+LB_1^{i-1}\right)\right]} \\ &\approx \frac{1}{f_2}\cdot\frac{f_M\left(t_i-\frac{T}{2}\right)\cdot\left(T+\delta_{LB}^{i,1}\right)+\Delta_M\cdot T\cdot LB_1^{i-1}}{T+\delta_{LB}^{i,1}} \\ &= \frac{1}{f_2}\cdot\left(f_M\left(t_i-\frac{T}{2}\right)+\frac{\Delta_M\cdot LB_1^{i-1}}{1+\delta_{LB}^{i,1}/T}\right). \end{aligned} \tag{23}$$

We omit the last term in step 3 because its magnitude is negligible in comparison with the other terms. Slave$_2$ passes on the received Sync message to Slave$_3$ at time $t_i+LB_1^i+LB_2^i$ and replaces the received master counter content by its estimate of the master counter value at the time of transmission. Using (16), (12), and (23), (2) becomes

$$\begin{aligned} \hat{M}\Big|_{S_2\mathrm{out}}^{t_i+LB_1^i+LB_2^i} &= \hat{M}\Big|_{S_1\mathrm{out}}^{t_i+LB_1^i}+\left(\hat{S}_2^{j(i)}(LD)+\hat{S}_2^i(BD)\right)\cdot RCF_2^i \\ &= \hat{M}\Big|_{S_1\mathrm{out}}^{t_i+LB_1^i}+\left(f_2\cdot LD_2^{j(i)}+f_2\cdot BD_2^i\right)\cdot RCF_2^i \\ &= \hat{M}\Big|_{S_1\mathrm{out}}^{t_i+LB_1^i}+\left(LD_2^i+BD_2^i\right)\cdot f_2\cdot RCF_2^i \\ &= M(t_i)+LB_1^i\cdot f_M\left(t_i-\frac{T}{2}\right)-\varepsilon_{LD}^{j,1}\cdot\frac{f_M\left(t_i-\frac{T}{2}\right)}{f_1} \\ &\quad +LB_2^i\cdot\left(f_M\left(t_i-\frac{T}{2}\right)+\frac{\Delta_M\cdot LB_1^{i-1}}{1+\delta_{LB}^{i,1}/T}\right) \\ &= M(t_i)+f_M\left(t_i-\frac{T}{2}\right)\cdot\left(LB_1^i+LB_2^i\right) \\ &\quad +\frac{\Delta_M\cdot LB_1^{i-1}\cdot LB_2^i}{1+\delta_{LB}^{i,1}/T}-\varepsilon_{LD}^{j,1}\cdot\frac{f_M\left(t_i-\frac{T}{2}\right)}{f_1}. \end{aligned} \tag{24}$$

For steps 2 to 3, we have used the fact that in the absence of jitter the line delay is constant; therefore, $LD_2^{j(i)}=LD_2^i$. From (10), the true value of the master counter at this time is

$$\begin{aligned} M\big|^{t_i+LB_1^i+LB_2^i} &= M(t_i)+f_M(t_{i-1}) \\ &\quad \cdot\left(LB_1^i+LB_2^i\right)+\frac{\Delta_M}{2}\cdot\left(LB_1^i+LB_2^i\right)^2. \end{aligned} \tag{25}$$

Hence, the error at this time in the TC transmission of Slave$_2$ is

$$\begin{aligned} & M-\hat{M}\Big|_{S_2\mathrm{out}}^{t_i+LB_1^i+LB_2} \\ &= \left(f_M(t_{i-1})-f_M\left(t_i-\frac{T}{2}\right)\right)\cdot\left(LB_1^i+LB_2^i\right) \\ &\quad +\frac{\Delta_M}{2}\cdot\left(LB_1^i+LB_2^i\right)^2-\Delta_M\cdot\frac{LB_2^i\cdot LB_1^{i-1}}{1+\delta_{BD}^{i,1}/T}+\varepsilon_{LD}^{j,1}\cdot\frac{f_M\left(t_i-\frac{T}{2}\right)}{f_1} \\ &= \frac{\Delta_M}{2}\cdot\left[T\cdot\left(LB_1^i+LB_2^i\right)+\left(LB_1^i+LB_2^i\right)^2\right] \\ &\quad -\Delta_M\cdot\frac{LB_2^i\cdot LB_1^{i-1}}{1+\delta_{BD}^{i,1}/T}+\varepsilon_{LD}^{j,1}\cdot\frac{f_M\left(t_i-\frac{T}{2}\right)}{f_1}. \end{aligned} \tag{26}$$

Using the latency defined in (14), the RCF and error expressions generalize as follows to slave N (inductive proof without error in line delay in [10]):

$$RCF_N^i\approx\frac{1}{f_N}\cdot\left(f_M\left(t_i-\frac{T}{2}\right)+\Delta_M\cdot\frac{L_{N-1}^{i-1}}{1+\delta_L^{i,N-1}/T}\right) \tag{27}$$

$$\begin{aligned} M-\hat{M}\Big|_{S_N\mathrm{out}}^{t_i+L_N^i} &= \frac{\Delta_M}{2}\cdot\left(T\cdot L_N^i+{L_N^i}^2\right)-\Delta_M\cdot\sum_{n=1}^{N-1}\frac{LB_{n+1}^i\cdot L_n^{i-1}}{1+\delta_L^{i,n}/T} \\ &\quad +\varepsilon_{LD}^{j,1}\cdot\frac{f_M\left(t_i-\frac{T}{2}\right)}{f_1} \end{aligned} \tag{28}$$

or, in an equivalent form, which makes clearer the dependence on the *difference* in latency values

$$\begin{aligned} M-\hat{M}\Big|_{S_N\mathrm{out}}^{t_i+L_N^i} &= \frac{\Delta_M}{2}\cdot\left[L_N^i\cdot T+\sum_{n=1}^{N}\left(LB_n^i\right)^2\right] \\ &\quad +\Delta_M\cdot\sum_{n=1}^{N-1}\frac{LB_{n+1}^i\cdot\delta_L^{i,n}\cdot\left(1+L_n^i/T\right)}{1+\delta_L^{i,n}/T}+\varepsilon_{LD}^{j,1}\cdot\frac{f_M\left(t_i-\frac{T}{2}\right)}{f_1}. \end{aligned} \tag{29}$$

The first term in the second line of (29) is a zero mean term that is present due to the different Sync message propagation times. The generalization for Slave$_\mathrm{N}$'s master counter estimate is

$$\begin{aligned} \hat{M}\Big|_{S_N\mathrm{out}}^{t_i+L_N^i} &\approx M(t_i)+f_M\left(t_i-\frac{T}{2}\right)\cdot L_N^i \\ &\quad +\Delta_M\cdot\sum_{n=1}^{N-1}\frac{LB_{n+1}^i\cdot L_n^{i-1}}{1+\delta_L^{i,n}/T}-\varepsilon_{LD}^{j,1}\cdot\frac{f_M\left(t_i-\frac{T}{2}\right)}{f_1}. \end{aligned} \tag{30}$$

From this, we see that, regardless of the subsequent master behavior, the slaves' estimates grow with the master frequency gradient in the most recent two Sync messages. To see what error is added in each element, it is useful to write the foregoing formulas in incremental form. The *incremental* master estimate follows directly from (2), using (16), (12), and (27), and the same reasoning as in (24), i.e.,

$$\begin{aligned} \hat{M}\Big|_{S_N\mathrm{out}}^{t_i+L_N^i} &= \hat{M}\Big|_{S_{N-1}\mathrm{out}}^{t_i+L_{N-1}^i}+\left(LD_N^i+BD_N^i\right)\cdot f_N\cdot RCF_N^i \\ &= \hat{M}\Big|_{S_{N-1}\mathrm{out}}^{t_i+L_{N-1}^i}+LB_N^i\cdot f_M\left(t_i-\frac{T}{2}\right)+\Delta_M\cdot\frac{LB_N^i\cdot L_{N-1}^{i-1}}{1+\delta_L^{i,N-1}/T}. \end{aligned} \tag{31}$$

Whereas the *incremental* true master counter is [from (10)]

$$M\big|^{t_i+L_N^i}=M\big|^{t_i+L_{N-1}^i}+f_M\left(t_i+L_{N-1}^i\right)\cdot LB_N^i+\frac{\Delta_M}{2}\cdot{LB_N^i}^2 \tag{32}$$

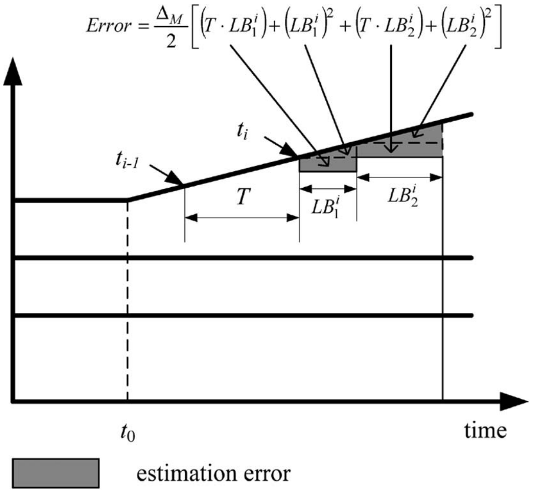

$Error = \frac{\Delta_M}{2}\left[\left(T\cdot LB_1^i\right)+\left(LB_1^i\right)^2+\left(T\cdot LB_2^i\right)+\left(LB_2^i\right)^2\right]$
freq
t_i
t_{i-1}
f_M
f_{S1}
f_{S2}
T
LB_1^i
LB_2^i
t_0
time
estimation error

Fig. 3. Synchronization error at slave$_2$.

so that the *incremental* master counter estimate error is

$$\begin{aligned} M-\hat{M}\Big|_{S_N\text{out}}^{t_i+L_N^i} &= M-\hat{M}\Big|_{S_{N-1}\text{out}}^{t_i+L_{N-1}^i} + \frac{\Delta_M}{2}\cdot LB_N^{i^2} \\ &\quad + \left(f_M\left(t_i+L_{N-1}^i\right) - f_M\left(t_i-\frac{T}{2}\right)\right) \\ &\quad \cdot LB_N^i - \Delta_M\cdot\frac{LB_N^i\cdot L_{N-1}^{i-1}}{1+\delta_L^{i,N-1}/T} \\ &= M-\hat{M}\Big|_{S_{N-1}\text{out}}^{t_i+L_{N-1}^i} + \Delta_M\cdot LB_N^i \\ &\quad \cdot\left(L_{N-1}^i+\frac{T+LB_N^i}{2}-\frac{L_{N-1}^{i-1}}{1+\delta_L^{i,N-1}/T}\right) \\ &= M-\hat{M}\Big|_{S_{N-1}\text{out}}^{t_i+L_{N-1}^i} + \Delta_M\cdot LB_N^i \\ &\quad \cdot\left(\frac{T+LB_N^i}{2}+\frac{\delta_L^{i,N-1}\cdot\left(1+L_{N-1}^i/T\right)}{1+\delta_L^{i,N-1}/T}\right). \end{aligned} \tag{33}$$

This shows the new error term added in each element if the master drifts. The first term does not change sign, and the second is zero mean due to the variation in line and bridge delays. If these were constant in time, then the error would increase by

$$(M-\hat{M})\big|_{S_N\text{out}}^{t_i+L_N^i} - (M-\hat{M})\big|_{S_{N-1}\text{out}}^{t_i+L_{N-1}^i} = \frac{\Delta_M}{2}\cdot\left(T\cdot LB_N^i+LB_N^{i^2}\right) \tag{34}$$

for all elements after slave$_1$. Slave$_1$ also incurs the error in line delay estimation caused by the frequency drift in the master clock [last term in (29)]. Fig. 3 shows the error term of (34) for the second slave and illustrates the incremental propagation down the line as long as the master drift persists. The areas of the gray rectangles correspond to the first error summand, and the area of the triangles to the second error summand.

The worst-case scenario is that the drift changes its direction right after the transmission of the ith Sync message. In this case, Slave$_\text{N}$'s master counter estimate follows (30). However, if Δ'_M is the slope of the new frequency change, then the true counter value of the master at the corresponding time is

$$M|^{t_i+L_N^i} = M|^{t_i+L_{N-1}^i} + f_M\left(t_i+L_{N-1}^i\right)\cdot LB_N^i + \frac{\Delta'_M}{2}\cdot LB_N^{i^2}. \tag{35}$$

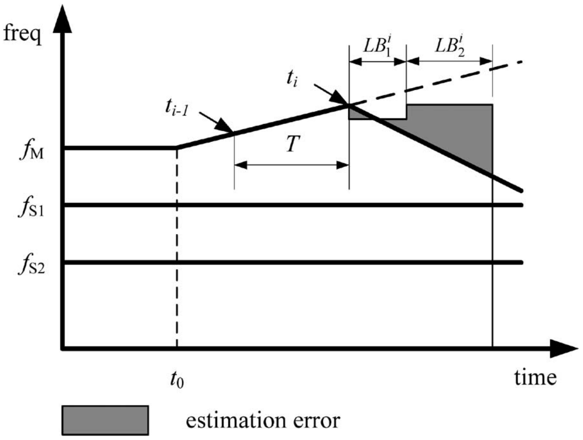


Fig. 4. Synchronization error at slave$_2$ after a change in the master frequency drift.

The *incremental* estimation error in this case will be

$$\begin{aligned} M-\hat{M}\Big|_N^{t_i+L_N^i} &= (M-\hat{M})\Big|_{N-1}^{t_i+L_{N-1}} + \frac{\Delta'_M}{2}\cdot LB_N^{i^2} \\ &\quad + \left(f_M\left(t_i+L_{N-1}^i\right) - f_M\left(t_i-\frac{T}{2}\right)\right) \\ &\quad \cdot LB_N^i - \Delta_M\cdot\frac{LB_N^i\cdot L_{N-1}^{i-1}}{1+\delta_L^{i,N-1}/T} \\ &= M-\hat{M}\Big|_{S_N\text{out}}^{t_i+L_{N-1}^i} + \Delta'_M\cdot LB_N^i\cdot\left(L_{N-1}^i+\frac{LB_N^i}{2}\right) \\ &\quad + \delta_M\cdot LB_N^i\cdot\left(\frac{T}{2}-\frac{L_{N-1}^{i-1}}{1+\delta_L^{i,N-1}/T}\right) \\ &= M-\hat{M}\Big|_{S_{N-1}\text{out}}^{t_i+L_{N-1}^i} + \left(\Delta'_M-\Delta_M\right)\cdot LB_N^i \\ &\quad \cdot\left(L_{N-1}^i+\frac{LB_N^i}{2}\right) + \Delta_M\cdot LB_N^i \\ &\quad \cdot\left(\frac{T+LB_N^i}{2}+\frac{\delta_L^{i,N-1}\cdot\left(1+L_{N-1}^i/T\right)}{1+\delta_L^{i,N-1}/T}\right). \end{aligned} \tag{36}$$

Fig. 4 shows in gray this error at the second slave for constant line and bridge delays, where the last summand in (36) vanishes. Figs. 3 and 4 show that the slaves partially follow the frequency change of the master. As the calculation of RCF uses two consecutive Sync messages, slave elements learn the trend of the frequency change of the master from the counters delivered in these two Sync messages. If the frequency drift changes its direction after the departure of a Sync message, then the propagation of this Sync message does not contain any information about the new frequency drift, and the slaves will still follow the old frequency drift. At that moment, the (old) line delay estimation does not reflect the new frequency drift either.

D. Error in RCF and Master-Counter Estimation Due to Slave Frequency Drift for Nondrifting Master

For the nondrifting master, (7) and (9) apply, whereas for the drifting slaves, (8) and (11) apply. Summarizing from [14], the

RCF and the master counter estimation error for slave N are

$$RCF_N^i \cong \frac{f_M}{f_N\left(t_i + L_N^i - \frac{T+\delta_L^{i,N}}{2}\right)} \tag{37}$$

$$\begin{aligned} M-\hat{M}\Big|_{S_N\mathrm{out}}^{t_i+L_N^i} &= M-\hat{M}\Big|_{S_{N-1}\mathrm{out}}^{t_i+L_{N-1}^i} + f_M\cdot\left(LD_N^i - LD_N^{j(i)}\right) \\ &\quad + \frac{f_M\cdot\Delta_{N-1}}{f_N(t_i)}\cdot\frac{RD_{N-1}^{j(i)}}{4}\cdot\left(R+RD_{N-1}^{j(i)}\right) - \frac{f_M\cdot\Delta_N}{f_N(t_i)} \\ &\quad \cdot\left(LD_N^{j(i)}\cdot\left(\frac{T+\delta_L^{i,N}}{2} - A_N^i\right) + RD_{N-1}^{j(i)}\right. \\ &\quad \left.\cdot\frac{R+RD_{N-1}^{j(i)}}{4} + BD_N^i\cdot\left(BD_N^i + T + \delta_L^{i,N}\right)/2\right). \end{aligned} \tag{38}$$

This error has four components. The 1st term is the error handed down by the predecessor. The 2nd term is due to the estimated line delay being different from the actual incurred line delay, when there are jitters. The 3rd term is the own error, even if the own frequency is constant due to the error in line delay computation if the predecessor is drifting. Finally, the 4th is the own error due to the own frequency drift, which also includes a clear line delay computation error. We see that each drifting slave leads to an additive error of identical structure, which will be passed on unchanged down the line. An additional smaller error, due to the error in the line delay estimation, is contributed to his successor. For every drifting slave, these error terms get added both in the drifting slave and its successor and are passed down the line together with the previous accumulated error.

IV. Analysis of Error Propagation—Part 2: The Influence of Jitter at Constant Clock Speeds

Without frequency drift, rounding and receive/transmit stamp jitter are the main sources of synchronization error. Both effects accumulate along the line, compromising the synchronization precision of distant elements. The resulting error propagation is not additive because jitter also affects the RCF estimation. Since in (2), the line and bridge delay estimates enter via their sum, we have in [10] "virtually" allocated the whole jitter to the line delay. Here, we drop this simplification. Let the sum of the random variables "receive and transmit stamp jitter" in element n be $J_n = J_n^{\mathrm{rec}} + J_n^{\mathrm{trans}}$, with mean $\overline{J}_n$. The jitter that affects an element's estimate of its bridge delay $\hat{S}_n(BD_n^i)$ is $I_{BD}^i(J_n)$, which is an instantiation of J_n, and let $I(\breve{J}_n)$ be a sample of $J_n - \overline{J}_n$. All are in time, not counters. The line delay of a message and the RCF estimate of element n are affected by samples of $J_n' = J_{n-1}^{\mathrm{trans}} + J_n^{\mathrm{rec}}$. To simplify, we consider all elements to be equal, i.e., $J_n' = J_n = J$. Then, $\hat{S}_n(BD_n^i) = f_n\cdot(BD_n^i - I_{BD}^i(J))$. The (constant) cable delay to element n is CD_n, with $LD_n^i = CD_n + I(J_n')$. Adapting to jitter and constant frequencies the formula obtained in [13] for the line delay estimate of slave n [cf. (13)], we obtain

$$\hat{S}_n(LD_n^j) \approx f_n\cdot\left[CD_n + \frac{1}{2}\cdot\left(I(J_{n-1}) + I(J_n) + \frac{RD_{n-1}^j}{R}\cdot\delta_{LD}^{j,n}\right)\right] \tag{39}$$

where $\delta_{LD}^{j,n}$ is zero mean. Since for equal elements $J_{n-1} = J_n = J$, averaging results in

$$avg\left(\hat{S}_n\left(LD_n^j\right)\right) \approx f_n\cdot(CD_n + \overline{J}) \tag{40}$$

i.e., averaging the line delay estimates removes the mean of the line delay jitter from the master counter estimation error. Using the fact that for equal elements $LD_n^i = CD_n + I_{LD}^i(J)$

$$\begin{aligned} avg\left(\hat{S}_n\left(LD_n^j\right)\right) &\approx f_n\cdot\left(LD_n^i - I_{LD}^i(J) + \bar{J}\right) \\ &= f_n\cdot\left(LD_n^i - I_{LD}^i(\breve{J})\right). \end{aligned} \tag{41}$$

Thus, the total line and bridge estimate jitter in (2) is

$$\begin{aligned} &avg\left(\hat{S}_n\left(LD_n^j\right)\right) + \hat{S}_n(BD_n^i) \\ &\quad\approx f_n\cdot\left(LD_n^i - I_{LD}^i(\breve{J})\right) + f_n\left(BD_n^i - I_{BD}^i(J)\right) \\ &\quad= f_n\cdot\left(LB_n^i - I_{LD}^i(\breve{J}) - I_{BD}^i(J)\right) \\ &\quad\equiv f_n\cdot\left(LB_n^i - \widetilde{J}_n^i\right). \end{aligned} \tag{42}$$

We have denoted $\widetilde{J}_n^i \equiv I_{LD}^i(\breve{J}) + I_{BD}^i(J)$.

In addition, the master counter estimate needs to be rounded to an integer before it can be passed on. This is captured in the rounding error $\rho_{\hat{M}}^{i,n}$, which is a zero-mean random variable with $|\rho_{\hat{M}}^{i,n}| \le 1/2$. Summarizing from [10], the RCF estimate of $\mathrm{Slave_N}$ is

$$RCF_N^i = \frac{f_M}{f_N}\cdot\left(1 - \varepsilon_{RCF}^{i,N}\right) \tag{43}$$

and the TC transmission error of $\mathrm{Slave_N}$ is

$$M - \hat{M}\Big|_{S_N\mathrm{out}}^{t_i+L_N^i} = f_M\cdot\sum_{n=1}^{N}\left(\left(LB_n^i - \widetilde{J}_n^i\right)\cdot\varepsilon_{RCF}^{i,n} + \widetilde{J}_n^i - \frac{\rho_{\hat{M}}^{i,n}}{f_M}\right) \tag{44}$$

where the error in the RCF computation of the Nth slave is given by the recursive formula (45), shown at the bottom of the page, and $\varepsilon_{RCF}^{i,1} = \delta_{LD}^{i,1}/(T + \delta_{LD}^{i,1})$ is the particularization of (45) to $N = 1$. In (44), the first term is recursively defined and reflects the errors in RCF computation. The other two additive terms reflect the direct impacts of jitter and rounding error. They are present even if the RCF computation is perfect.

The benefit of RCF estimate averaging becomes obvious in this scenario. Let two Sync messages be used for RCF computation, and let Δt_M and Δt_S be their interdeparture and

$$\varepsilon_{RCF}^{i,N} = \frac{\sum_{n=1}^{N}\left(\breve{J}_n^i - \breve{J}_n^{i-1}\right) + \sum_{n=1}^{N-1}\frac{\rho_{\hat{M}}^{i,n} - \rho_{\hat{M}}^{i-1,n}}{f_M} + \sum_{n=1}^{N-1}\left(\left(LB_n^i - \widetilde{J}_n^i\right)\cdot\varepsilon_{RCF}^{i,n} - \left(LB_n^{i-1} - \widetilde{J}_n^{i-1}\right)\cdot\varepsilon_{RCF}^{i-1,n}\right)}{T + \sum_{n=1}^{N-1}\delta_{BD}^{i,n} + \sum_{n=1}^{N}\left(\breve{J}_n^i - \breve{J}_n^{i-1}\right)} \tag{45}$$

interarrival times measured in absolute time. Then, $\Delta t_S = \Delta t_M + \delta_{LD}$, and the corresponding counter increases are $\Delta C_M = f_M \cdot \Delta t_M$ and $\Delta C_S = f_S \cdot \Delta t_S$. Hence

$$RCF = \frac{\Delta C_M}{\Delta C_S} = \frac{f_M}{f_S} \cdot \left(1 - \frac{\delta_{LD}}{\Delta t_S}\right). \tag{46}$$

Since the second term is zero mean, averaging will result in the desired convergence of RCF to the quotient of the clock frequencies. The following section discusses the consequences of the obtained analytical results.

V. Discussion and Recommendations

We have examined the error at the "points of support" of the synchronization process, i.e., the times when an element receives new information via a Sync message. In between, the estimate of the master counter is updated according to

$$\hat{M}_n|_{t_i+L_n^i+\tau} = \hat{M}_n|_{t_i+L_n^i} + S_n(\tau) \cdot RCF_n^i, \tau \in \left[0, T_{\text{sync}} + \delta_L^{i+1,n}\right). \tag{47}$$

The quality of any synchronization result in between these time instants is critically dependent on the correctness of the support estimates and on the stability of the master quartz during Sync message interarrival intervals (holdover time).

To derive recommendations for parameter choice, a system-identification process has to determine whether the momentary limiting factor is due to frequency drift or to jitters.

If the limiting factor is frequency drift, equations (28) and (29) indicate how to achieve a possible reduction of the time synchronization error, namely by

1) minimizing the influence of outside disturbances (e.g., temperature change) on the oscillator (e.g., insulation);
2) choosing oscillators that are not sensitive to disturbances;
3) decreasing T (inter-transmission time of the two Sync messages used for RCF calculation); however, this conflicts with the goal to minimize the influence of jitter;
4) shortening the latency (line and bridge delays);
5) regulating the forwarding of Sync messages so that the bridge delays are approximately constant over time, removing the zero mean error term;
6) shortening the response delay and interval between delay messages to reduce the error in line delay estimation;
7) developing new synchronization algorithms that take into consideration the effect of frequency change. A possible solution was presented in [11].

   If the limiting factor is jitter, then (44) and (45) indicate how to achieve a reduction of the synchronization error, namely by
8) increasing T [see 3)] to decrease the RCF error; however, this is detrimental if the master frequency is changing over time;
9) decreasing bridge delay (and cable delay if possible);
10) reducing jitter by improved hardware;
11) averaging over a number of consecutive RCF estimates to reduce the error in the recursive additive term.

On the other hand, the comparison of master drifting versus slaves drifting [see (33) and (38)] shows that the single-slave frequency drift is very benign compared with the master frequency drift, which is only matched by all slaves drifting. We can conclude that a costly master brings a disproportionately large synchronization benefit as compared with allocating the corresponding cost fraction to improving each slave.

TABLE I
SIMULATION SETTINGS

Parameter	Value
Number of elements	80
Nominal Frequency	100MHz
Cable delay	100ns
Bridge delay	Uniform [5 15]ms
Frequency Change	3ppm/s
Interval of Sync Message	32ms
Interval of Delay_request	8s
Interval of RCF calculation	200ms
Number of line delay averaging	8
Number of RCF averaging	7

VI. Simulation Results

We have developed a MATLAB simulation tool to test and analyze the synchronization performance of IEEE 1588 in a line with cascaded bridges. We have used this tool to simulate PTP in PROFINET [15]. The model parameters, which are summarized in Table I, are given by the Siemens Automation and Drive Department. In the simulation performed for this paper, we investigate the effect of linear frequency drifts of 3 ppm/s. Since we are not investigating quartzes but are investigating synchronization, we have herein combined the effects of several possible perturbation causes, such as aging, temperature drift, voltage fluctuations (electromagnetic compatibility), etc. The frequency change starts at 20 s, increases with maximal slope in the next 20 s, and then stays constant again. For these simulations, we have used bridge delays of 5–15 ms, which have been state-of-the-art until recently, and which clearly illustrate the obtained results. Decreased bridge delays of 5–125 μs lead to significant performance gains.

A. Simulation Scenario 1: Master Drifting

In this section, we simulate the case where the master is drifting while the slaves' frequencies are constant.

We first look at the error in line delay estimation, shown in Fig. 5. The first slave estimates the line delay based on the time stamped in itself and in the master, whose frequency is drifting during [20, 40] s. As shown, error is introduced in the line delay estimation. This can be observed in subplot 1. Subplot 2 shows the line delay estimation error at slave 4. Since the estimation of line delay at slave 4 only depends on the frequencies of slaves 4 and 3, which do not drift, the estimate is correct. Subplot 1 also shows that by shortening either the response delay or the interval between delay messages, the magnitude of the line delay estimation error is reduced.

The errors in the master counter estimation are shown in Fig. 6. We see that the frequency change introduces a bias to the estimated master time (the error is not zero mean). The magnitude of this bias increases along the line. Our analytic result in (33) also shows that each slave adds its own contribution to this bias. When the frequency change stops, this bias disappears. The noise on top of the bias is present due to the variation of the line delays and bridge delays.

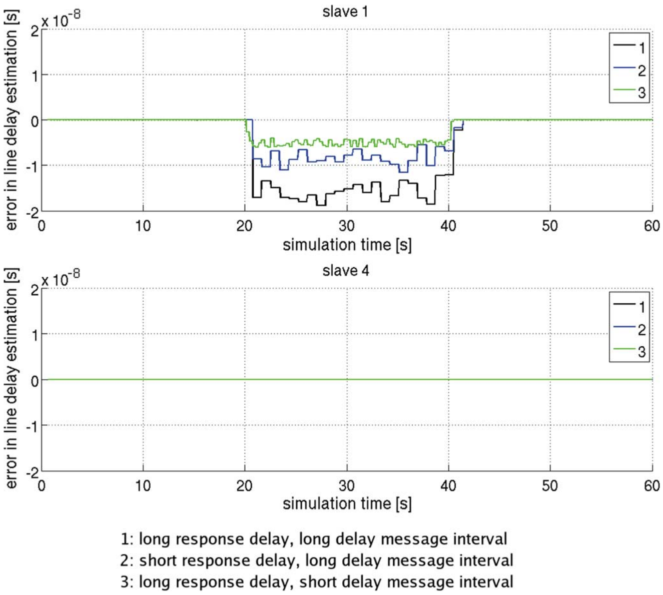


Fig. 5. Line delay estimation error when the master is drifting.

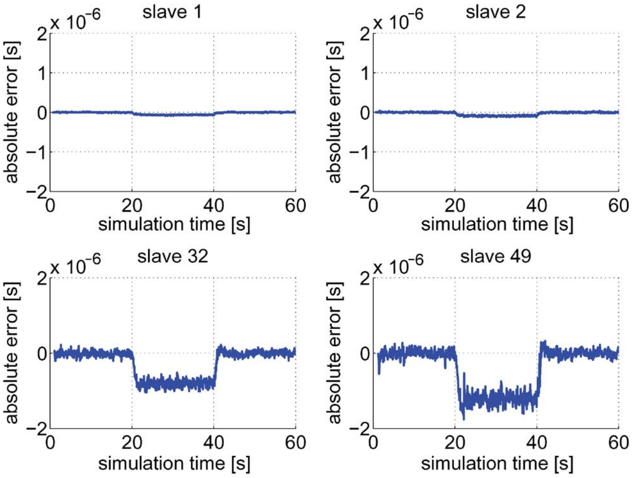


Fig. 6. Sync error at slaves 1, 2, 32, and 49 when the master is drifting.

As discussed in Section V, several measures can be taken to reduce the effect of outside disturbances. We halve the interval of RCF calculation and run the simulation again. The results, which are shown in Fig. 7, have clearly improved.

B. *Simulation Scenario 1: Different Slaves Drifting*

In this section, we simulate the case when drifting takes place at different slaves while the master frequency is constant and display the synchronization error in the following figures: Fig. 8 for the case that only the first slave is drifting; Fig. 9 if exactly the first two slaves are identically drifting; and Fig. 10 if all the slaves are identically drifting.

Comparing Figs. 8–10 with Fig. 6, we observe that, if only one element is drifting, the error at the end of the line is by a factor of 10 larger if this element is the master and not a slave. The same relationship holds for the master versus two slaves drifting. Only if all the slaves exhibit identical nonzero frequency drift do they match the effect of "master only drifting."

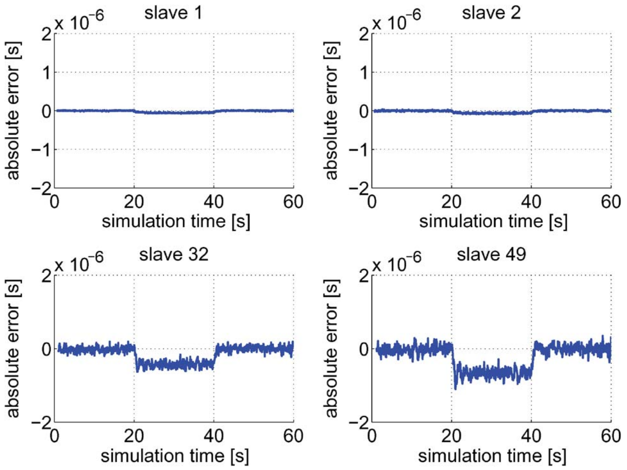


Fig. 7. Sync error at slaves 1, 2, 32, and 49 when master is drifting with the interval of RCF calculation halved to 100 ms.

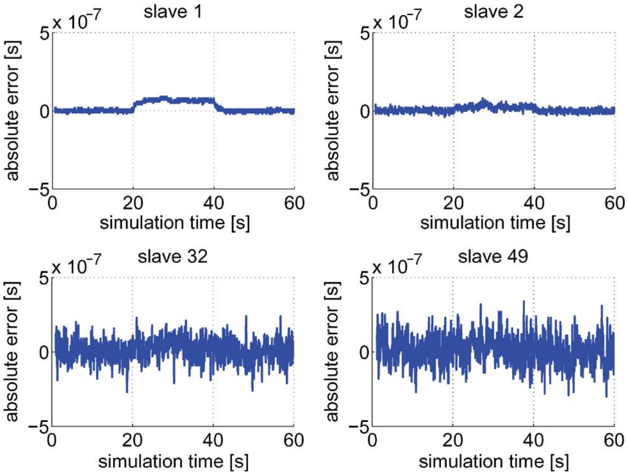


Fig. 8. Sync error at slaves 1, 2, 32, and 49 when slave 1 is drifting.

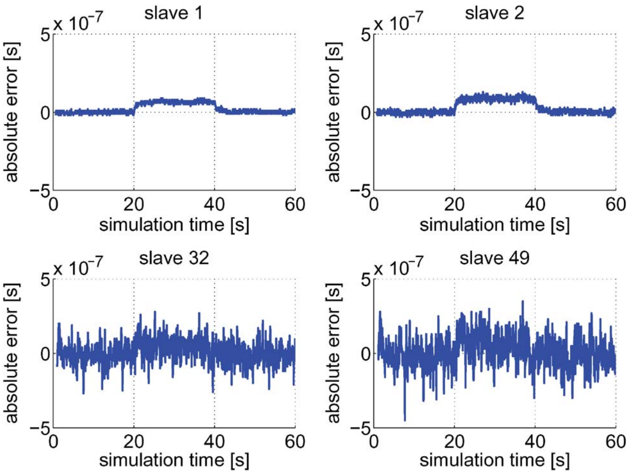


Fig. 9. Sync error at slaves 1, 2, 32, and 49 when slaves 1 and 2 are drifting.

VII. Future Work

Our next steps will be to further investigate short-term frequency deviations, e.g., due to vibration and shocks, as well as work toward more precise statements concerning the influence

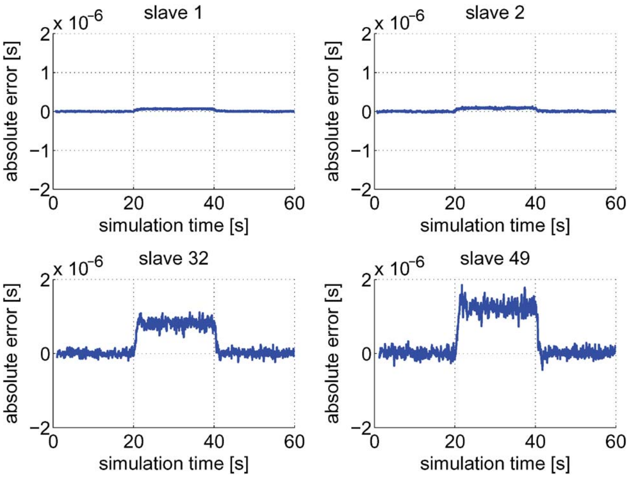


Fig. 10. Sync error at slaves 1, 2, 32, and 49 when all the slaves are drifting.

of quantization errors on synchronization accuracy. A first report of this work has recently been published in [16].

References

[1] *IEEE Standard for LAN/MAN CSMA/CD Access*, IEEE 802.3, 1997.

[2] J. Nieminen, "Synchronization of next generation wireless communication systems," M.S. thesis, Helsinki Univ. Technol., Finland, 2007.

[3] *IEEE Standard for Local and Metropolitan Area Networks—Timing and Synchronization for Time-Sensitive Applications in Bridged Local Area Networks*, IEEE 802.1AS, 2007.

[4] B. Chen, Y. P. Chen, J. M. Xie, Z. D. Zhou, and J. M. Sa, "Control methodologies in networked motion control systems," in *Proc. Int. Conf. Mach. Learn. Cybern.*, Guangzhou, China, 2005, pp. 1088–1093.

[5] D. L. Mills, "Internet time synchronization: The network time protocol," *Network Working Group Request for Comments: 1129*, Oct. 1989.

[6] D. L. Mills, "Precision synchronization of computer network clocks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 24, no. 2, pp. 28–43, Apr. 1994. [Online]. Available: citeseer.nj.nec.com/mills94precision.html

[7] *IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*, ANSI/IEEE Std. 1588-2002.

[8] J. Jasperneite, K. Shehab, and K. Weber, "Enhancements to the time synchronization standard IEEE-1588 for a system of cascaded bridges," in *Proc. IEEE Int. Workshop Factory Commun. Syst.*, 2004, pp. 239–244.

[9] *IEEE P1588TM D2.2 Draft Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*, 2007.

[10] C. Na, D. Obradovic, R. L. Scheiterer, G. Steindl, and F. J. Goetz, "Synchronization performance of the precision time protocol," in *Proc. IEEE Int. Symp. Precision Clock Synchronization Meas., Control Commun.*, Vienna, Austria, 2007, pp. 25–32.

[11] C. Na, D. Obradovic, R. L. Scheiterer, G. Steindl, and F. J. Goetz, "Enhancement of the precision time protocol in automation networks with a line topology," in *Proc. IFAC World Congr.*, Seoul, Korea, 2008.

[12] D. Obradovic, R. L. Scheiterer, C. Na, G. Steindl, and F. J. Goetz, "Clock synchronization in industrial automation networks: Comparison of different synchronization methods," in *Proc. 5th Int. Conf. Informat. Control, Autom. Robot.*, Funchal, Portugal, 2008.

[13] R. L. Scheiterer, D. Obradovic, C. Na, G. Steindl, and F. J. Goetz, "Synchronization performance of the precision time protocol: Effect of clock frequency drift on the line delay computation," in *Proc. WFCS*, Dresden, Germany, 2008, pp. 243–246.

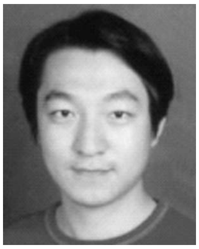
[14] R. L. Scheiterer, C. Na, D. Obradovic, G. Steindl, and F. J. Goetz, "Synchronization performance of the precision time protocol in the face of slave clock frequency drift," in *Proc. IEEE CASE Conf.*, Washington DC, 2008, pp. 554–559.

[15] J. Jasperneite and J. Feld, "PROFINET: An integration platform for heterogeneous industrial communication systems," in *Proc. 10th IEEE ETFA*, 2005, pp. 9–12.

[16] R. L. Scheiterer, C. Na, D. Obradovic, G. Steindl, and F. J. Goetz, "1 μs-conform line length of the transparent clock mechanism defined by the precision time protocol (PTP Version2)," in *Proc. IEEE Int. Symp. Precision Clock Synchronization Meas., Control Commun.*, Ann Arbor, MI, 2008, pp. 92–97.

Ruxandra Lupas Scheiterer received the Vordiplom degree in electrical engineering from the Technical University of Munich, Germany, in 1983 and the M.S. degree in electrical engineering and computer science and the Ph.D. degree in information sciences and systems from Princeton University, NJ, in 1985 and 1989, respectively.

Since 1989, she has been with Corporate Technology, Siemens AG, Munich, Germany, with occasional maternity arrangements. She is currently a Senior Research Scientist with the Information and Communications Department. Her interest in math was sparked as a 12-year-old in Romania by her grandmother, a mathematician. She won the German National High-School Math Competition in 1979, 1980, and 1981. Her R&D interests include optimization and learning for intelligent systems and the interworking of telecommunication, control, and decision making.



Chongning Na (S'08) received the B.E. degree in mechanical engineering in 2002 from Tsinghua University, Beijing, China, and the M.S. degree (with an academic award) in electrical engineering in 2005 from the Technical University of Munich, Germany, where he is currently working toward the Ph.D. degree. His Ph.D. work is funded by Siemens AG, Corporate Technology, Information, and Communications.

In 2002, he was with Microsoft Research Asia, Beijing. Since Summer 2004 he has been with Siemens AG, Munich, and since Spring 2005 with Corporate Technology, Information, and Communications. His research interests are in information theory, probabilistic inference, machine learning and their application in telecommunications, control systems, and signal processing.

Dragan Obradovic (M'86) received the B.E. and M.E. degrees in mechanical engineering from the University of Belgrade, Yugoslavia, in 1980 and 1985, respectively, and the Ph.D. degree in mechanical engineering from the Massachusetts Institute of Technology, Cambridge, in 1990.

Since 1992, he has been with Corporate Technology, Siemens AG, Munich, Germany, where he is currently a Senior Research Scientist with the Information and Communications Department. He is also an Industrial Lecturer with the Munich University of Technology, where he has been teaching since 2001 with the international Masters of Science in Communication Engineering program within the Electrical Engineering Department. His research involves the development and application of machine learning methods for system control and signal processing. He has published dozens of journal and international conference papers and the book *An Information-Theoretic Approach to Neural Computing* (New York, Springer, 1996) and is author of several patents.

Günter Steindl received the Master degree in electrical engineering from Fachhochschule Regensburg, Ratisbona, Germany, in 1990.

In 1990, he joined the Industry Sector, Automation Systems, Research, and Development Department, Siemens AG, Amberg, Germany, where he developed the communication protocol PROFIBUS DP. In 2001, he began the design of the architecture of the Ethernet communication protocol PROFINET IO. In 2002, he started work on the industrial standardization of PROFINET IO by the Profibus User Organisation for Siemens AG and, in 2005, the international standardization of the IEC SC 65.