Research on IEEE 1588 Clock Synchronization Error Correction Algorithm Based on Kalman Filter

Rongbin Ju[1], Rongmao Wang[1], Peng Yu[1], Rong Chen[1], Shunjiang Wang[1] and Guangfu Wang[2*]

[1]State Grid Liaoning Electric Power Supply Co., Ltd., Shenyang, Liaoning, 110006, China

[2]Graguate Department, Shenyang Institute of Engineering, Shenyang, Liaoning, 110136, China

[*]Corresponding author's e-mail: m18265537369@163.com

Abstract. In order to reduce the influence of network transmission jitter error on IEEE1588 time synchronization protocol and improve the accuracy of network timing, this paper studies the principle of IEEE1588 protocol. Then the Kalman smoothing filter correction algorithm is proposed. Firstly, according to the transmission mode of IEEE 1588 message, several factors affecting the synchronization accuracy are analyzed, and the noise model is introduced. The synchronization error is compensated by Kalman filter algorithm. Finally, the simulation comparison proves that this algorithm can greatly improve the accuracy of network pairs.

1. Introduction

With the rapid development of network system and industrial technology, the requirements for synchronization accuracy in various fields are increasingly high, especially in the fields of industrial automation, distributed system and remote communication system, the requirements for synchronization accuracy have reached the order of microseconds, which is a severe challenge to the existing synchronization technology[1]. The commonly used network synchronization technologies include network time protocol (NTP), simple network time protocol (SNTP) and global positioning system (GPS).

NTP is mainly used to make computers on the Internet keep time synchronization of a communication protocol, the protocol synchronous message receiving and sending time in the application layer to obtain, which determines that its synchronization accuracy can only reach the level of milliseconds, unable to meet the needs of high precision field[2]. SNTP is a simplified NTP protocol, which is mainly used in networks with less strict requirements on the accuracy of time synchronization. GPS global positioning system is currently widely used in multi-point synchronization technology, which can meet the needs of high-precision communication field, but it is easy to be interfered by environmental factors, and its security performance is not guaranteed[3]. In order to meet the demand of high-precision synchronization, the network precision clock committee drafted the IEEE1588 protocol. Compared with the above three technologies, the protocol has the advantages of high precision, simple configuration, fast convergence, low requirements on network bandwidth and low resource consumption, which is the future development trend of synchronization technology[4].

In view of this problem, the delay jitter error of network transmission affects the synchronization accuracy. This paper proposes the advantages of combining kalman smoothing filtering algorithm with

Journal of Physics: Conference Series **1449** (2020) 012109 doi:10.1088/1742-6596/1449/1/012109

kalman filtering method and simplifying filter parameters to improve the accuracy and reduce the complexity of the algorithm[5].

2. Introduction to IEEE 1588 synchronization protocol

2.1. IEEE 1588 principles and mechanisms

2.1.1. Master clock and slave clock. IEEE 1588 Precision clock synchronization protocol for network measurement and control system is commonly referred to as PTP (Precision Timing Proto Timing col). This protocol uses BMC (BestMaster Clock) to determine the most stable and accurate Clock in a group of clocks, which is called the master Clock. The master Clock provides the source time to correct the slave Clock by sending synchronous packets.

2.1.2. Synchronous message. When synchronization starts, the master clock sends a synchronization message (Sync) to the slave clock and records the exact moment of transmission t_{m1}. When this message is received, the precise time of receipt t_{s2} is recorded from the clock. Then the master clock sends the timestamp tm1 to the slave clock by following the message (Follow_Up). After receiving the Follow_Up message, two time values t_{m1} and t_{s2} are recorded from the slave clock. Assuming that the transmission delay between master clock and slave clock is t_{ms}, the following formula can be obtained:

$$t_{ms} = t_{s2} - t_{s1} \tag{1}$$

Synchronous messages are generally sent every 1s, so the delayed sampling period (T_{ms}) from the master clock to the slave clock is as follows:

$$T_{ms} = 1s \tag{2}$$

2.1.3. Delay request message. After receiving Follow_Up message, the slave clock sends Delay_Req request message to the master clock at an interval (generally 2 to 30 synchronous message intervals). The slave clock records the exact time t_{s3} sent by this message, while the master clock receives the Delay_Req message and records the exact time t_{m4} received. The transmission delay between the slave clock and the master clock is T_{sm}, and the formula can be obtained:

$$t_{sm} = t_{m4} - t_{s3} \tag{3}$$

2.1.4. One-way transmission delay. PTP protocol assumes that the transmission of the message is symmetric, so the average of the delay T_{ms} from the master clock to the slave clock and the delay T_{sm} from the clock to the master clock offset the error of the master and slave clocks in the delay measurement, thus calculating the one-way transmission delay t_{delay} of the message:

$$t_{delay} = (t_{sm} - t_{ms}) / 2 \tag{4}$$

2.1.5. Master slave clock offset. PTP protocol calculates the offset between master and slave clocks (t_{offset}), and corrects the slave clock with the calculated offset. The offset is calculated by the difference between the delay T_{ms} from the master clock to the delay T_{sm} from the slave clock and the one-way transmission delay t_{delay}, namely:

$$t_{offset} = t_{ms} - t_{delay} \tag{5}$$

The schematic diagram of the measurement mechanism is as follows:

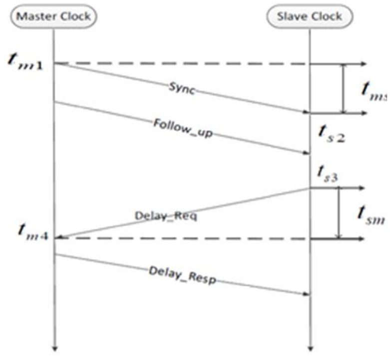

Figure 1. IEEE1588 delay-request-response mechanism

2.2. Factors that affect timing accuracy of network

According to the synchronization principle and mechanism of PTP, it can be concluded that the main factors affecting the timing accuracy of the network are as follows.

First, the delay jitter of the protocol stack. As shown in figure 2, after the generation of PTP synchronous packet in the application layer, it needs to pass through each layer of the protocol stack to reach the physical layer for transmission. However, the queuing time of the packet in the stack is uncertain, which causes synchronization error, which is called delay and jitter of the protocol stack. It can be seen that the closer the location of timestamp is to the physical layer, the higher the timing accuracy is. As shown in figure 2, there are three timestamp positions of A, B and C. A is timestamp in application layer software. If the timestamp is done at this point, the delay jitter error of the entire protocol stack is included, so the synchronization accuracy is the lowest. Point B is timestamped in the media access control layer, which is close to the underlying physical layer and improves accuracy compared to point A. Point C is timestamped by hardware in the physical layer, which avoids the delay jitter error of protocol stack and has the highest precision. PTP protocol is the use of physical layer hardware timestamp, so that the generated timestamp can more accurately reflect the message sent time.

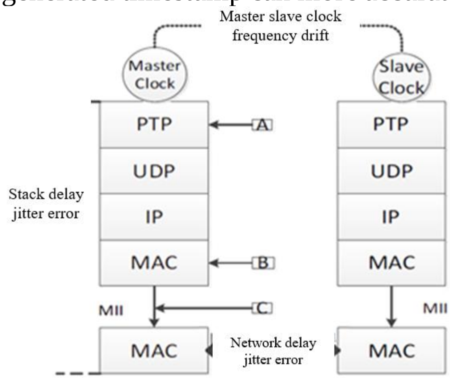

Figure 2. Synchronization error factors and timestamp selection

Second, the network transmission delay jitter error. PTP synchronization mechanism assumes that the network transmission delay is symmetric, but the actual network often contains many network components such as switches and routers. Due to the uncertainty of the internal storage and forwarding time of packets, the transmission delay of the actual network is asymmetric. Since the change of PTP network load and the residence time of packets in network switch are calculated by local clock, it is inevitable that some errors will be generated. At this time, it is necessary to adjust and compensate the calculation of network delay.

Third, the frequency drift of the slave clock with respect to the master clock. Local clocks are counted by crystal oscillator, and even if they are the same crystal oscillator, they will inevitably produce errors. In this way, there will be some deviation between master and slave clocks in each synchronization interval. In order to reduce deviation error, the crystal frequency of slave clocks must be controlled. It is assumed in this paper that the frequency offset of the master and slave clocks is a constant, so that the accumulated time deviation of the slave clock with respect to the master clock during each synchronization period is approximately the same.

Fourth, other influencing factors. For example, the stability of the network, mainly the change of the network topology structure, will have a great impact on the synchronization accuracy. In addition, the sending interval of synchronization message is also an important factor affecting synchronization accuracy. The smaller the interval, the higher the transmission frequency of synchronous message, and the higher the synchronization accuracy. However, it also increases the load of the network. If the network bandwidth is relatively small, it will be counterproductive. So you have to find a good frequency to send, usually every 1s or 2s.

3. The realization of kalman smoothing filter

3.1. Offset noise model

The delayed request and response mechanism assumes that the transmission delay of upstream and downstream of the master-slave clock is equal, but in reality, the delay of upstream and downstream of the network is asymmetric, that is, T_{ms}=T_{sm}. Let the proper time delay between master and slave clocks be expressed by t_d, and the error caused by the change of network load and the presence of packets in network elements be similar to gaussian noise[6]. Therefore, let the jitter errors of upper and lower line transmission be W_{ms} and W_{sm}, respectively, and W_{ms}~(0, σ_1^2), W_{sm}~(0, σ_2^2). Are available:

$$t_{ms} = t_d + w_{ms} \tag{6}$$

$$t_{sm} = t_d + w_{sm} \tag{7}$$

Substitute (6) and (7) into (4) and (5) to get:

$$\hat{t}_{offset} = \overline{t}_{offset} + (w_{sm} - w_{ms})/2 \tag{8}$$

On the type of $\hat{t}_{offset}$ t is calculated with delay jitter error offset value, $\overline{t}_{offset}$ mainly offset from the initial deviation, and by a type (8) the $\hat{t}_{offset}$~($\overline{t}_{offset}$, σ^2), by the above process can be seen that the calculated offset IEEE1588 synchronization mechanism $\hat{t}_{offset}$ should obey the mean as the $\overline{t}_{offset}$, the variance of the σ^2 of gaussian distribution. On this basis, can join filter in from the clock module, to measure the offset value $\hat{t}_{offset}$ filter the noise, as shown in figure 3.

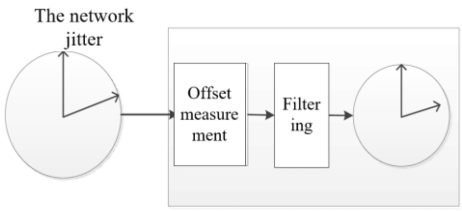

Figure 3. Add filter module from clock end

3.2. The design of kalman smoothing filter

Kalman filter is a kind of based on time domain and can use the software realization of the filter[7], it USES the principle of recursive estimation filtering calculation, using the estimates of the moment and the moment before the observation to the estimate of the current moment, the core of the algorithm is based on establishing the system of equation and observation equation to meet the need to deal with signals under the condition of minimum mean square error estimate[8].

Kalman filter contains two processes. The first process is called prediction process. It contains two basic equations to describe the discrete state model.

$$x(k+1) = Ax(k) + Bu(k) + Cw(k) \tag{9}$$

Measurement equation:

$$y(k) = Dx(k) + v(k) \tag{10}$$

In the above equation, *x(k)* represents the state quantity of the system at time k, *u(k)* represents the control input quantity of the system at time k, *y(k)* represents the measured value of the system at time *k*, *w* and *v* are the process noise and measurement noise of the system respectively, and *A*, *B* and *C* are the parameters of the system.

The second process is the correction process. The correction process is responsible for feedback and uses the measurement update equation to establish a posterior estimate of the improvement of the current state on the basis of the prior estimation value of the estimation process and the current measurement variable[9]. It is divided into two stages: TU(Time Update) and MU (Measurement Update). TU is the state transition stage between two sampling periods, and its update equation is as follows:

$$\hat{X}(k/k-1) = \hat{X}(k-1) + Bu(k-1) \tag{11}$$

$$\hat{P}(k/k-1) = A\hat{P}(k-1)A^T + Q \tag{12}$$

Updating equation of MU stage:

$$\hat{K}(k) = \hat{P}(k/k-1)C/(C\hat{P}(k/k-1)C^T + R) \tag{13}$$

$$\hat{X}(k) = \hat{X}(k/k-1) + \hat{K}(k)[Y(k) - C\hat{X}(k/k-1)] \tag{14}$$

$$\hat{P}(k) = [I - \hat{K}(k)C]\hat{P}(k/k-1) \tag{15}$$

The following is the interpretation of each quantity in equations (11) to (15) :

$\hat{X}(k/k-1)$ said moment by moment condition forecasting system of *k-1*. $\hat{X}(k-1)$ represent the best state of the *k-1* hour; $\hat{X}(k)$ is the best k moment, a moment before it is to use estimates to predict the result. Formula (14) is to update the status value of the system in combination with the prediction results. $\hat{P}(k/k-1)$ said $\hat{X}(k/k-1)$ covariance; $\hat{P}(k)$ said $\hat{X}(k)$ of covariance, the formula (12), (15) their updating equations respectively. $\hat{K}(k)$ of the kalman filter gain coefficient, the formula (13), the renewal of the gain coefficient equations. *Q* represents the covariance matrix of process noise; *R* represents the measurement noise covariance matrix; *I* is the identity matrix.

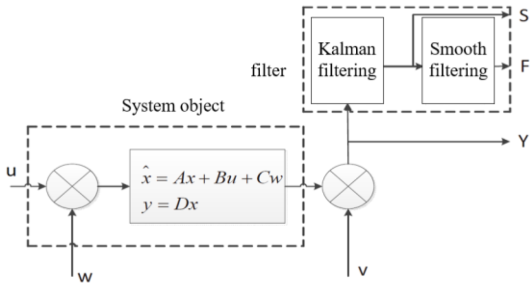

Figure 4. System structure diagram

The core of kalman filter is to use the above equations to continuously predict and update the state value, so as to achieve the best estimation of the state variable. On this basis, this paper simplified the parameters of kalman filter for the noise model of master-slave clock offset, and added the smoothing filter module (see figure 4). Since the main purpose of the experiment is to filter out noise, no control variable *u* is needed, so the parameter *B* of the control quantity is set to be 0. The state variable *X* is reduced to *A* one-dimensional variable in the algorithm. In order to simplify the operation, parameters A=1 and C=1 are taken here. In addition, the system deviation measured value *Y* is a first-order value, so it can be known that the parameter value of *Y* in the above formula should also be a value[10]. In the simulation, the filter model was simplified, and its parameter *D*=1 was taken.

In figure 4 joined the simple smooth filtering module, in this module, we every N points to calculate an average, *N*=20, in the design of this namely smoothing filter output $F(k)=\sum_{i=0}^{N-1} S(k-i)/N$, type of $k<0$, $S(k)=0$. *S(k)* and *F(k)* correspond to *S* and *F* in figure 4, respectively, and are the outputs of kalman filter and smoothing filter. The advantage of this module is that the hardware is simple and the filtering effect is obvious.

4. Simulation results and analysis

MATLAB simulation platform to write the program to build the master and slave clock simulation model, through the model can verify the effectiveness of kalman smoothing filter module. Initial deviation $\overline{t}_{offset}$ in the simulation of master-slave clock offset of 10us, master-slave clock frequency drift for 10ppm, synchronous intervals of 1s, so we can calculate each synchronization interval, the actual deviation of master-slave clock accumulated 10us, algorithms detect master-slave offset value in the 100s, and carries on the filter, and then use the offset value from the clock for correction of measurement, analysis and comparison the master-slave clock error size before and after correction.

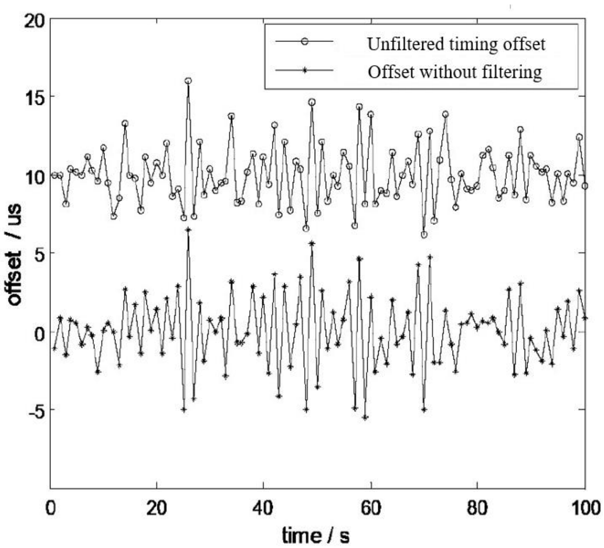

Figure 5. Master slave clock offset before and after synchronization without filtering

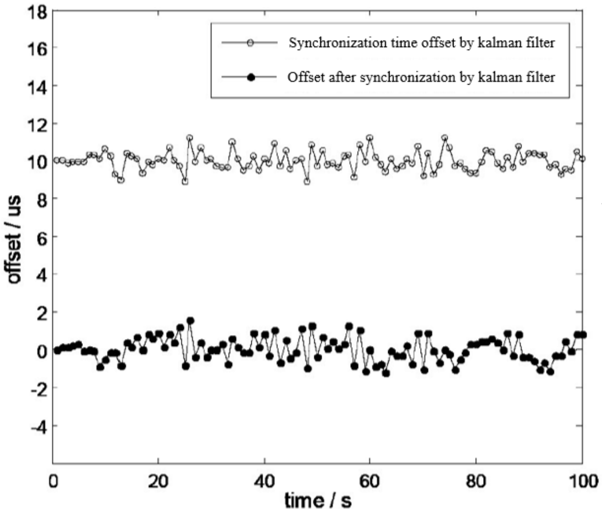

Figure 6. Master slave clock offset before and after synchronization by kalman filter

Figure 5 is used without filter filter master-slave clock offset value $\hat{t}_{offset}$ from clock offset synchronization. In the figure, the upper curve shows the master/slave clock deviation before the filter synchronization time, which fluctuates roughly between 5 and 15 microns, and approximates the gaussian distribution with an average of 10 microns. The bottom curve shows the measured offset adjusted for the deviation from the clock, which shows that the error margin after synchronization is about -5 to 5 microns. Although the error is reduced to a certain extent, the accuracy is not high.

Figure 6 shows that the measured offset value is first filtered by kalman filter and then synchronized with the slave clock. The upper curve represents the variation of master and slave clocks before the kalman filter synchronization time, ranging from 8 microns to 12 microns. The lower curve is the master/slave clock deviation synchronised by the kalman filter, ranging from -2ms to 2ms. At the same time, by comparing figure 5 and figure 6, it can be seen that the synchronization accuracy is greatly improved after filtering by kalman filter.

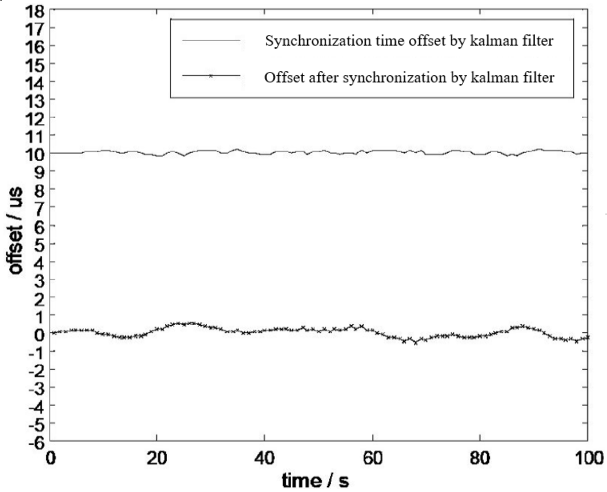

Figure 7. Master and slave clock offsets before and after synchronization by kalman smoothing filter

The upper and lower curves in Fig 7 are the errors of master and slave clocks before and after the synchronization time of kalman smooth filter, respectively. Compared with Fig 6, it is obvious that the synchronization accuracy is further improved. The above experimental results show that the kalman smoothing filter module can effectively improve the network delay jitter error and the synchronization accuracy of PTP.

5. conclusion

In this paper, the IEEE1588 protocol is studied, and a kalman smoothing filter is proposed to reduce the network jitter error. This method combines the advantages of kalman filtering and smoothing filtering to simplify the filter parameters, and the module is simple, easy to implement, and the filtering effect is good. The experimental results show that the synchronization accuracy of this algorithm is much higher than that of the single kalman filter. IEEE1588 protocol is gradually improved in theory, but it has not been popularized in practice. In the future, theoretical research should be combined with practice to achieve the expected effect and reduce the complexity in practical application as far as possible.

Acknowledgments

This paper is supported by “State grid science and technology project(2019YF-28)”.

References

[1] Exel, R. (2014) Mitigation of Asymmetric Link Delays in IEEE 1588 Clock Synchronization Systems. IEEE Communications Letters, 18: 507-510.
[2] Lee, S. (2008) An Enhanced IEEE 1588 Time Synchronization Algorithm for Asymmetric Communication Link using Block Burst Transmission. IEEE Communications Letters, 12: 687-689.
[3] Chin, W., Chen, S. (2009) IEEE 1588 clock synchronization using dual slave clocks in a slave. IEEE Communications Letters, 13: 456-458.
[4] Lv, S., Lu, Y., and Ji, Y. (2010) An Enhanced IEEE 1588 Time Synchronization for Asymmetric Communication Link in Packet Transport Network. IEEE Communications Letters, 14: 764-766.
[5] Guo, H. and Crossley P. (2017) Design of a Time Synchronization System Based on GPS and IEEE 1588 for Transmission Substations. IEEE Transactions on Power Delivery, 32: 2091-2100.
[6] Hammi, F.M., Ghannouchi. (2009) Twin nonlinear two-box models for power amplifiers and transmitters exhibiting memory effects with application to digital pre-distortion. IEEE Trans on Microw Wireless Compon. 19: 530-532.
[7] Diversi, R., Guidorzi, R. and Soverini, U. (2005) Kalman filtering in extended noise environments. IEEE Transactions on Automatic Control, 50: 1396-1402.
[8] Wang, S., Feng, J. and Tse, C. K. (2013) Analysis of the Characteristic of the Kalman Gain for 1-D Chaotic Maps in Cubature Kalman Filter. IEEE Signal Processing Letters, 20: 229-232.
[9] Pantaleon, C., and Souto, A. (2005) Comments on "An aperiodic phenomenon of the extended Kalman filter in filtering noisy chaotic signals. IEEE Transactions on Signal Processing, 53: 383-384.
[10] Lu, X., Wang, L. and Wang, H. (2016) Kalman filtering for delayed singular systems with multiplicative noise. IEEE/CAA Journal of Automatica Sinica, 3: 51-58.