Scholar Publishing Group
Distributed Processing System
https://doi.org/10.38007/DPS.2021.020405
ISSN 2790-0916 Vol. 2, Issue 4: 36-43

Clock Synchronization Algorithm of Distributed System Based on Adaptive Technology

Norizani Diahane[*]

Binghamton University State University of New York, United States of America

[*]*corresponding author*

Keywords: Adaptive Technology, Distributed System, Clock Synchronization Algorithm, SFTRLS Algorithm

Abstract: The clock synchronization of distributed systems has gone through the process of logical clock synchronization to physical clock synchronization. Whether it is logical clock synchronization or physical clock synchronization, extra accurate time is not particularly important, if each node has the same time, then different events in the distributed system can be executed step by step. The purpose of this paper is to study the clock synchronization algorithm for distributed systems based on adaptive technology. The distributed clock synchronization prototype system is sorted out, different functional modules are proposed according to the requirements, and experiments are carried out on specific equipment and switches. Analyze and model the request-response mechanism of PTP synchronization in detail. Correction is made by introducing interference sources that affect the clock synchronization accuracy under actual working conditions. An adaptive local clock synchronization algorithm based on AR model and SFTRLS algorithm is designed. The clock synchronization accuracy and stability of the improved and pre- and post-synchronization algorithms are compared through computer simulation. The colored noise error with a variance of 3×10-5s2 is relatively reduced by 76.93 %. It is verified that the algorithm can effectively improve the clock synchronization performance.

1. Introduction

With the rapid development of power distribution systems such as industrial process control, chemical industry, and electric power, the requirements for clock synchronization accuracy are getting higher and higher. For example, in a data acquisition system, data acquisition must have good real-time performance, otherwise reading or saving data will be interrupted [1]. Therefore, a high degree of clock consistency is an important condition for the correct operation of distributed systems. At present, the DCS on the market usually adopts a real-time way to achieve clock synchronization [2-3].

While there are many existing clock synchronization algorithms, many lead to destructive discontinuities of virtual clocks and most rely on regular communication, which is not suitable for large systems [4]. Instead, Berneburg JA proposes a novel clock synchronization algorithm that allows for continuous virtual clocks, with a dynamic event-triggered communication strategy that is non-Zeno-free in that it guarantees a designable positive time between communication instances [5]. Ansere JA proposes an Adaptive Beacon Time Synchronization (ABTS) algorithm to enhance timing message synchronization. The ABTS algorithm selects the best time synchronization pair to reduce the number of timing messages transmitted. To guarantee the reliability of synchronization, the clock offset of connected vehicles is periodically readjusted to provide better synchronization accuracy and lowest energy consumption. Based on bidirectional timing message synchronization, we derive a maximum likelihood (ML) estimate of clock skew design in a generalized transmitter-to-receiver system and its equivalent Cramer-Rao lower bound (CRLB) by assuming a Gaussian noise model. We propose a node pair selection algorithm to improve pairwise beacon message time synchronization. Simulation results verify that our proposed ABTS algorithm outperforms other protocols in terms of synchronization accuracy, convergence speed and energy efficiency [6]. Therefore, we need to find an efficient and more accurate Ethernet synchronization method to achieve high-speed DCS clock synchronization [7-8].

Based on the existing clock synchronization algorithm, this paper proposes a clock synchronization algorithm based on adaptive clock synchronization technology, establishes a new adaptive clock synchronization algorithm, and derives and solves the clock phase deviation between nodes on the basis of the new timing mode, and the error model of the new algorithm is given. The performance of the new algorithm is tested through experiments. The experimental results show that the adaptive clock synchronization algorithm proposed in this paper can achieve smaller errors in the network environment of the distributed clock synchronization prototype system.

2. Research on Clock Synchronization Algorithm of Distributed System Based on Adaptive Technology

2.1. Basic Requirements of Distributed System Clock Synchronization

Compared with the constant clock of the integrated system, the distributed system does not have a fixed and unified time or a prescribed time, but the distributed system has higher requirements for clock coordination, such as job scheduling, multitasking, coordination algorithms, etc. [9-10]. In order to reduce the impact of wrong time on DCS, a physical or logical clock should be set as the master clock in the distributed system, and other nodes in the system use the master clock for time-related operations [11].

2.2. Adaptive Clock Synchronization Technology

The adaptive clock recovery method realizes the synchronization of the packet network clock, and in the absence of a common reference clock, the sender's service clock can be recovered. When the working clock of the sender is unstable, the receiver can adjust and change the clock together with the sender in an appropriate way to achieve network synchronization [12]. There is no need to purchase high-precision clocks during network construction, which greatly reduces network construction costs [13-14]. However, adaptive clock recovery methods that reset the sender's service clock based on the receiver's buffer fill or packet arrival delay interval can be implemented in software with a high degree of automation. TDM devices that support clock recovery can operate independently, plug and play. However, due to the randomness of the data packet transmission delay in the intermediate packet transmission network, the clock recovery performance is greatly affected. This requires terminal equipment to adopt appropriate filtering algorithms and avoidance methods to reduce the impact of packet transmission delay on clock recovery performance in packet transmission networks [15-16].

2.3. Adaptive Clock Synchronization Algorithm

An adaptive PTP is designed for three main sources of interference introduced in the revised interference-based clock synchronization model: timestamp error w,(n), network noise d(n) and clock frequency drift p(n). Clock synchronization control algorithm, the functional block diagram of the local clock synchronization control system after applying the algorithm is shown in Figure 1. ζ (n) in the figure represents the descrambling residue, which is used as a feedback signal for clock frequency synthesis. The designed PTP clock synchronization control algorithm consists of an autoregressive model (AR) based interference canceller, an adaptive weight control mechanism based on a stable fast transverse recursive least squares (SFTRLS) algorithm and a clock rate calculation controller with a feedback controller. It is composed of other main modules, and by adaptively updating the filter coefficients of the interference canceller, the influence of interference on the time deviation θ(n) is reduced and the clock synchronization performance and stability are improved [17-18].

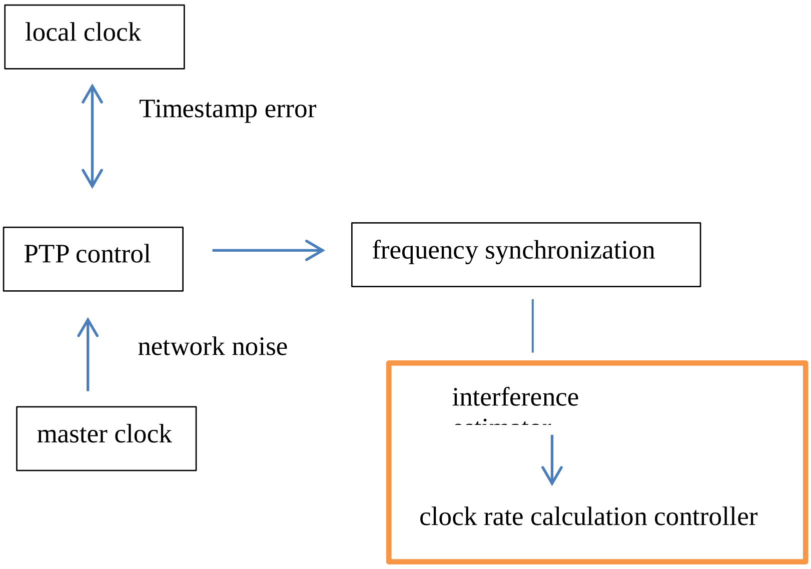

Figure 1. Functional block diagram of adaptive PTP local clock synchronization control system

The structure of the algorithm is based on AR model interference estimation, SFTRLS algorithm update of autocorrelation coefficient and direct compensation of θ(n). The interference estimation process based on AR model is represented by equation (1):

$$\hat{\theta}(n) = c_{ar} + \sum_{i=1}^{M} w_i \theta(n-i) \qquad (1)$$

The direct compensation process for θ(n) is expressed by equation (2):

$$\xi(n) = \theta(n) - \hat{\theta}(n) \qquad (2)$$

The SFTRLS algorithm stabilizes the algorithm by introducing computational redundancy feedback error, and each output sample value requires 9n multiplication and division operations.

3. Investigation and Research on Clock Synchronization Algorithm of Distributed System Based on Adaptive Technology

3.1. Overall Scheme Design of Hardware Platform

This experimental platform adopts a tree structure, and each synchronization hardware board is designed by the method described below. The overall structure diagram of the clock synchronization system network is shown in the figure below, which consists of a master clock node and N (numbers can change during operation) slaves composed of clock nodes. The overall block diagram of the clock synchronization network is shown in Figure 2:

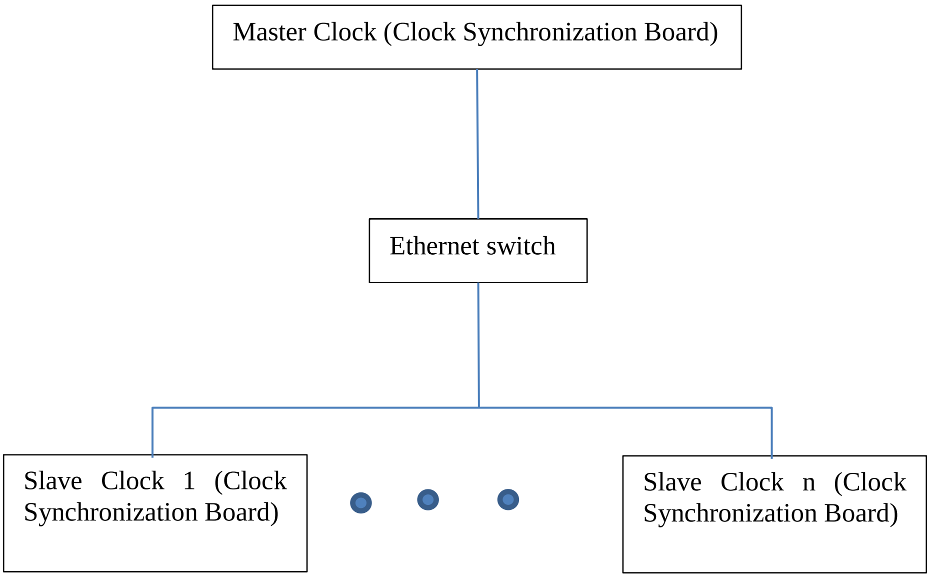
Master Clock (Clock Synchronization Board)
Ethernet switch
Slave Clock 1 (Clock Synchronization Board)
Slave Clock n (Clock Synchronization Board)

Figure 2. Overall block diagram of the clock synchronization system network

3.2. Main Functional Modules of the Distributed Clock Synchronization Prototype System

Clock Timing Module: This module mainly provides clock timing functions for clients or time servers. Since the performance counter QPC is used under the Windows system, the MyTimer class is implemented according to Microsoft's official documents on QPC.

Data filtering module: This module mainly provides data filtering and filtering functions for the client. For the timestamp data set obtained by the client, first, the confidence interval confidence_interval of the phase deviation and the data set deletion information drop array are obtained through the maximum intersection criterion, and then the minimum time The extension criterion selects the optimal data {T1, T2, T3, T4} from the remaining data according to the data deletion information array drop.

Frequency deviation correction module: This module mainly provides the client with the function of calculating and correcting clock frequency deviation. The frequency deviation correction module mainly has three parts, namely FrequencyCorrecting, LeastSquare and MyTimer.

Synchronization period adjustment module: This module mainly provides the client with the function of dynamically calculating the synchronization period, mainly including the PeriodAdjusting class, in which the PeriodAdjusting() function and the ~PeriodAdjusting() function are the constructor and destructor respectively; if the current clock is being corrected for the first time deviation, and wait for the next synchronization, use the firstCalculatePeriod() function to calculate the clock synchronization period value; if the phase deviation has not been corrected for M times of clock synchronization, use the secondCalculatePeriod() function to calculate the clock synchronization period; if it has experienced M times of clock synchronization to correct the phase deviation, execute the correction frequency deviation correction algorithm, and then use the thirdCalculatePeriod() function to calculate the clock synchronization period.

4. Simulation Verification and Result Analysis of Clock Synchronization Algorithm for Distributed System Based on Adaptive Technology

In order to verify the feasibility and superiority of the improved clock synchronization algorithm designed in this paper, the OMNeT++ network simulation platform is selected to conduct modeling and comparison simulation experiments between the improved algorithm and the traditional IEEE1588 clock synchronization algorithm. And add network noise, timestamp errors, and clock frequency drift as disturbances to the clock synchronization environment. The project is open sourced on GitHub and has a large library of development models. Compared with the expensive and single protocol model of other software, it has significant advantages.

Aiming at the impact of interference on the clock synchronization environment, by calling the LibPLN library to generate realistic clock noise to simulate clock drift, add the **.Slave.NIC.ClockType="LibPLN_ScheduleClock" field to the Slave's clock in the simulated network environment configuration file omnetpp.ini Type is configured as a clock with PLN noise, using the default configuration. The simulation is carried out with white noise and colored noise. It is realized by setting the value of the **.Slave.NIC.PHY[*].rxDelay field in the configuration file omnetpp.ini to increase the receiving delay of the Slave node by a random amount. Colored noise is modeled as a superposition of additive white Gaussian noise and a sinusoidal waveform.

The standard PTP local clock synchronization algorithm and the improved adaptive local clock synchronization algorithm are simulated respectively. Set the simulation time to 5000s, the time synchronization period and the observation period of the TimeDiffObserver module to be 2s, enable TimeJump and Syntonize, and set the length of the SFTRLS filter to 25. In order to exclude the influence caused by the initial value, the clock deviation after 1200s is taken for analysis. Here, the root mean square value is used to describe the time synchronization performance. The experimental results of the simulation experiments are shown in Table 1:

It can be seen from the results of the simulation experiments that the performance of the adaptive algorithm is better than that of the standard algorithm. Especially for colored noise with variance

3×10-5s2, the error is relatively reduced by 76.93%.

Table 1. Simulation results

Noise type	Noise variance/s^2	Relatively reduced error
White noise	3x10^{-5}	2.17
White noise	3x10^{-2}	4.74
Colored noise	3x10^{-5}	76.93
Colored noise	3x10^{-2}	10.4

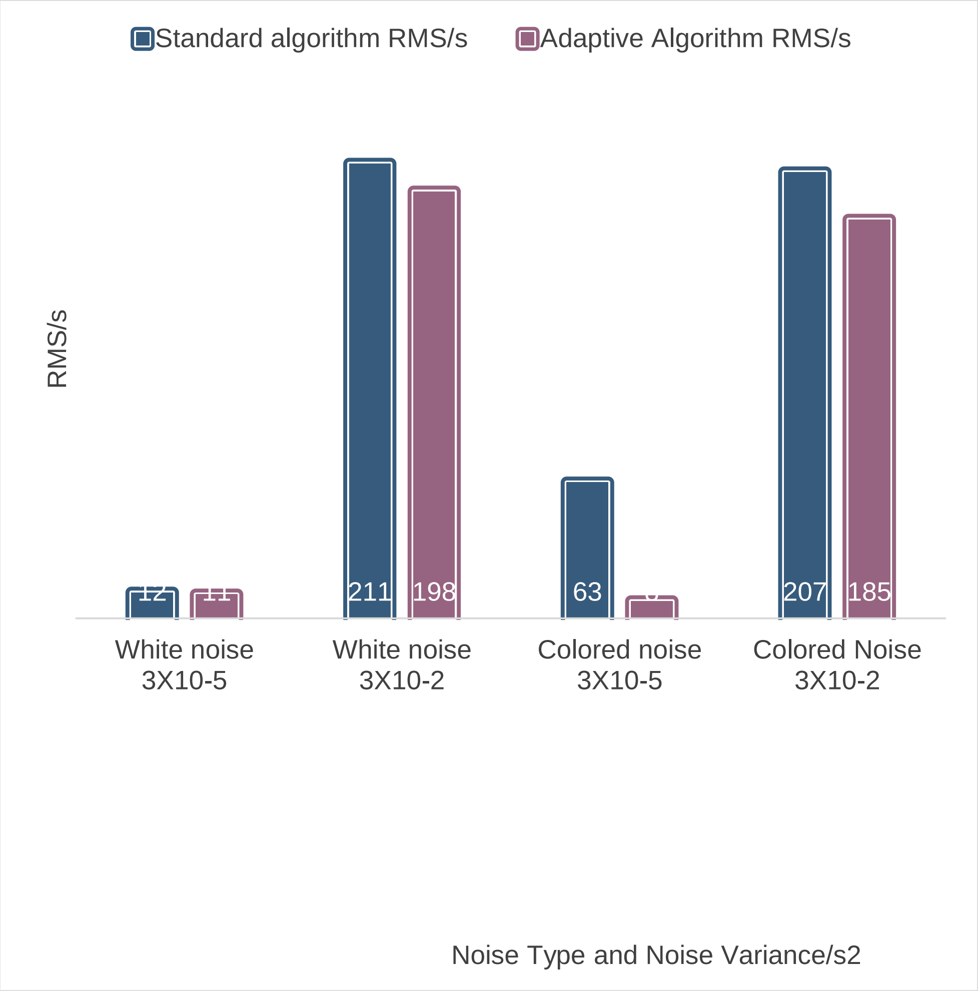

Figure 3. Comparison of standard and adaptive algorithms (X10-3)

5.Conclusion

The core idea of this paper is to design a new clock synchronization algorithm based on adaptive technology, improve and enhance the existing PTP clock synchronization control algorithm, and verify that the performance of the adaptive algorithm is better than the standard algorithm. However, in order to further improve the accuracy of clock synchronization and the performance of the system, future research work mainly includes the following aspects: The timestamps used for clock synchronization in this paper are at the application level. In the future, we can consider starting from the network protocol level or hardware level to ensure that all Timestamps are the exact moment when data packets are sent, thereby further improving the clock synchronization accuracy; this paper adopts the linear clock model, and the nonlinear clock model can be further considered in the future to further improve the clock synchronization accuracy; this paper can further consider using the data filtering stage Maximum likelihood estimation or other more mature data processing methods.

Funding

This article is not supported by any foundation.

Data Availability

Data sharing is not applicable to this article as no new data were created or analysed in this study.

Conflict of Interest

The author states that this article has no conflict of interest.

References

[1] Khattak Z H, Magalotti M J, Fontaine M D. Estimating safety effects of adaptive signal control technology using the Empirical Bayes method. Journal of safety research, 2018, 64(FEB.):121-128.

[2] Rouhani S, Deters R. Data Trust Framework Using Blockchain Technology and Adaptive Transaction Validation. IEEE Access, 2021, PP(99):1-1.

[3] Valerievich G, Ivanovich P, Liudmila S. Program for the rational choice of highly cost-effective adaptive technology of grain cultivation for various conditions of the European part of the Russian Federation. Istrazivanja i Projektovanja za Privredu, 2020, 18(2):216-221.

[4] Nikitin D A, Lukina N M, Ivanova T N, et al. Application Of Immunotropic Agents In The Adaptive Calf Management Technology. Reports, 2021, 2(336):5-12.

[5] Berneburg J A, Garcia E, Gerlach A, et al. Strongly Non-Zeno Event-Triggered Wireless Clock Synchronization. IFAC-PapersOnLine, 2020, 53(2):2745-2750.

[6] Ansere J A, Han G, Wang H. A Novel Reliable Adaptive Beacon Time Synchronization Algorithm for Large-Scale Vehicular Ad Hoc Networks. IEEE Transactions on Vehicular Technology, 2019, 68(12):11565-11576.

[7] Toufik A M, Yao J, Yuan J. Chorus-Line Algorithm For Clock Synchronization. IEEE Access, 2018, PP(99):1-1.

[8] Guarro M, Sanfelice R. An Adaptive Hybrid Control Algorithm for Sender-Receiver Clock Synchronization. IFAC-PapersOnLine, 2020, 53(2):1906-1911.

[9] Pinkl J, Cash E K, Evans T C, et al. Short-Term Pediatric Acclimatization to Adaptive Hearing Aid Technology. American Journal of Audiology, 2021, 30(1):76-92.

[10] Nguyen V Q, Nguyen T H, Jeon J W. An Adaptive Fuzzy-PI Clock Servo Based on IEEE 1588 for Improving Time Synchronization over Ethernet Networks. IEEE Access, 2020, PP(99):1-1.

[11] Contzen M P. Consensus based synchronization of clocks to diminish the effect of clock drifts in microgrids. IFAC-PapersOnLine, 2020, 53(2):12980-12985.

[12] Lyu, Hanbaek. Global Synchronization of Pulse-Coupled Oscillators on Trees. SIAM Journal on Applied Dynamical Systems, 2018, 17(2):1521-1559.

[13] Khanchandani P, Lenzen C. Self-Stabilizing Byzantine Clock Synchronization with Optimal Precision. Theory of computing systems, 2019, 63(2):261-305.

[14] Skiadopoulos K, Tsipis A, Giannakis K, et al. Synchronization of Data Measurements in Wireless Sensor Networks for IoT Applications. Ad Hoc Networks, 2019, 89(JUN.):47-57.

[15] Suzuki A, Masutomi K, Ono I, et al. CPS-Sim: Co-Simulation for Cyber-Physical Systems with Accurate Time Synchronization. IFAC-PapersOnLine, 2018, 51(23):70-75.

[16] Pourranjbar A, Baniasadi M, Abbasfar A, et al. A Novel Distributed Algorithm for Phase Synchronization in Unmanned Aerial Vehicles. IEEE Communications Letters, 2020, PP(99):1-1.

[17] Isaac, S, Chang, et al. Novel Method for Synchronization of Multiple Biosensors. IEEE Sensors Letters, 2019, 4(1):1-4.

[18] Karatalay O, Psaromiligkos I, Champagne B, et al. A Distributed Pulse-Based Synchronization Protocol for Half-Duplex D2D Communications. IEEE Open Journal of the Communications Society, 2021, PP(99):1-1.