IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 19, 2020, accepted March 26, 2020, date of publication April 1, 2020, date of current version April 23, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2984785

Clock Synchronization in Wireless Sensor Networks Based on Bayesian Estimation

TING YANG, YUQING NIU, AND JIEXIAO YU
School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China

Corresponding authors: Yuqing Niu (niuyuq@foxmail.com) and Jiexiao Yu (yjx@tju.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61971305, in part by the National Key Research and Development Program of China under Grant 2017YFE0132100, in part by the Headquarters of State Grid Corporation of Science and Technology Project under Grant Key Technologies of Multidimensional Monitoring and Operational Assistant Decision for Distribution Internet of Things', and in part by the China Postdoctoral Science Foundation under Grant 2019M651037.

ABSTRACT Clock synchronization is essential for the operation of upper layer applications in Wireless Sensor Networks. When the network hops needed for clock synchronization message transmission is large, synchronization error will accumulate and synchronization accuracy may be reduced significantly. Moreover, in the existing synchronization algorithms, large number of communication resources and node energy will be expended in sending and receiving time messages. To solve the problem, this paper proposes a Bayesian estimation-based time synchronization (BETS) algorithm which uses synchronization error compensation to reduce the amount of time message interaction in clock synchronization. The key idea of BETS is to calibrate the prior information of synchronization error with a small amount of field sampling time information, which will eliminate the impact of environment on clock synchronization accuracy. In addition, the gradient descent method is used to estimate the relative clock drift rate, which provides the reference for setting algorithm execution cycle and ensures clock synchronization during network operation time. In order to evaluate the theoretical lower bound of the performance of BETS, the Bayesian Cramér–Rao bound (BCRB) is derived. Both simulation and hardware experiments show that BETS algorithm makes full use of the prior information of synchronization error, hence fewer time messages are required in synchronization and the resource constraints of WSNs are satisfied.

INDEX TERMS Clock synchronization, wireless sensor networks (WSNs), Bayesian estimation, synchronization error.

I. INTRODUCTION

Wireless sensor networks (WSNs) form a network by ubiquitous large-scale sensor nodes in a self-organized way, its objective is to perceive physical world accurately and perform control through information feedback. In WSNs, implementation of most functions needs the support of clock synchronization technology, such as the communication protocol [1], [2], data fusion [3], [4], upper level applications [5], [6] and information security [7]. Limited by cost and power constraints in WSNs, the clock synchronization is mainly performed by means of low-cost and low-complexity methods such as time message interaction. By calculating with time message, the clock deviation between network nodes could be estimated and adjusted. Some famous clock synchronization algorithms have been proposed, including DMTS [8], TPSN [9], RBS [10], etc.

With wide application of sensor networks and Internet of things, the requirements for clock synchronization are continue to increase, thus scholars have carried out more extensive research and put forward many clock synchronization algorithms, such as [11]–[16]. Based on DMTS, a flooding time synchronization protocol (FTSP) is proposed in [11], which tries to improve synchronization accuracy by stamping every byte in the time message, however, few low-power MCU or clock chips can provide such function. In [12], time synchronization with immediate clock adjustment (TSICA) is proposed. After receiving time message from upstream node, TSICA optimizes the timing of local clock adjustment to improve synchronization accuracy. Nevertheless, TSICA will delay actively before forwarding time message to subordinate nodes, which increases the additional synchronization error. To improve energy efficiency, in [13], an event-triggered

The associate editor coordinating the review of this manuscript and approving it for publication was Alessandro Pozzebon.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

synchronization mechanism is proposed, which combines sensor measurement data with time messages, and MAC layer timestamping technology is used to improve synchronization accuracy. However, the algorithm adopts one-way time message propagation, transmission delay will significantly reduce the accuracy of the time message [14]. In [15], a long period synchronization algorithm is proposed, which uses Bayesian estimation to improve the accuracy of least square method in clock drift rate estimation. In [16], synchronization accuracy is improved by increasing clock measurement accuracy. With the clock measurement of upstream node, subordinate nodes use Bayesian estimation to improve local clock measurement accuracy.

The above algorithms all adopt ideal hypothesis, for example, the prior distribution parameters for clock drift rate are assumed to be known [15]. Whereas in practical application of sensor networks, it is difficult for sensor nodes to obtain those parameters accurately only by calculating the time information. Also, above algorithms seldom consider the influence of time message transmission delay, interrupt service program (ISR) processing delay, crystal temperature drift and other complex factors on the process of synchronization. As a result, considerable synchronization errors still exist after execution of synchronization algorithms. The accumulation of single hop clock synchronization error will have a significant impact on network operation. Tests based on Berkeley motes platform show that the average single hop error of clock synchronization is 45.2μs, nevertheless, the accumulated error after five hops rises to 73.6 μs [9]. In large-scale long-distance sensor networks, such as ubiquitous IoT dedicated to high-voltage transmission line monitoring and oil/gas transmission pipeline inspection [17], there are several dozen hops along the transmission path. Thus, the accumulation of synchronization errors will seriously affect the normal operation of network functions.

In recent years, researchers have tried to model clock phase difference and time message delay in clock synchronization. In [18], time message transmission delay is assumed to obey exponential distribution, and designs a minimum variance unbiased estimator (MVUE) to estimate the clock phase difference. Assuming that the time message transmission delay is Gaussian, a maximum likelihood estimator (MLE) is proposed to estimate the clock phase difference and drift rate [19]. In [20], a low-complexity maximum likelihood estimator (LCMLE) is proposed, which maps the clock parameter estimation to geometric analysis of feasible solution region. Thus, both clock phase difference and time message transmission delay are estimated jointly with synchronization accuracy close to that of MLE. However, in large-scale sensor networks, due to different hardware configurations in sensor nodes and influence of operating environment, there is a large deviation in clock phase difference between nodes. Besides, there are many uncertain factors in time message sending and receiving process, which increase the deviation in estimation when employing above-mentioned models. For example, the random delay resulting from clock register reading and writing is difficult to observe and estimate.

To solve the above problem, this paper proposes a Bayesian estimation-based time synchronization (BETS) algorithm to compensate for clock synchronization residual error, which obtain the prior information of clock synchronization error through experimental measurement. Considering the change of synchronization parameters affected by field environment, BETS uses Bayesian estimation to modify the prior information of estimated synchronization error, thus a more accurate error compensation value can be obtained. Main contributions are as follows.

1) A synchronization error compensation method is proposed, which solves the problem of synchronization residual error caused by random delays, such as time message transmission delay and ISR processing delay, etc. Through single step error compensation, BETS achieves the goal of reducing synchronization errors, which is usually accomplished by executing the clock synchronization multiple times in traditional algorithms. Hence, with the proposed method, a large number of synchronization message transmission is avoided, and energy consumption of WSNs is reduced.

2) A novel experimental measurement method is proposed to obtain the prior information of synchronization error. By using Bayesian estimation, the prior information is modified to solve the problem that the synchronization error is susceptible to field environment. In addition, the method of obtaining prior information experimentally can further reduce the number of time messages sent and received during the execution of error compensation. Thus, both communication resource overhead and on-chip energy consumption can be reduced effectively.

3) The Bayesian Cramér–Rao bound for estimation of clock synchronization error is derived, which serves as a fundamental performance criterion for the case when the prior distribution of the clock parameters to be estimated is available.

The rest of this paper is organized as follows. In Section II, the source of clock synchronization error is analyzed. Section III presents the BETS algorithm and clock drift rate estimation in detail, and performance bound of BETS is derived in Section IV. In Section V, various simulation and actual test results are provided for performance evaluations. Section VI concludes this paper.

II. CLOCK SYNCHRONIZATION ERROR MODEL

In the hardware of sensor nodes, the sending and receiving time of a single clock message is only tens of microseconds. Hence, the impact of clock drift on synchronization accuracy can be ignored in the actual synchronization process. In the subsequent analysis, it is assumed that the clock synchronization error is only caused by phase synchronization error. The clock synchronization model based on time message exchange is shown in Fig. 1. Defining the timestamp set in the ith bidirectional time message sending and receiving process as $\{T_{1,i}, T_{2,i}, T_{3,i}, T_{4,i}, \}_{i=1}^{n}$. Where $T_{1,i}$ and $T_{3,i}$ are sending

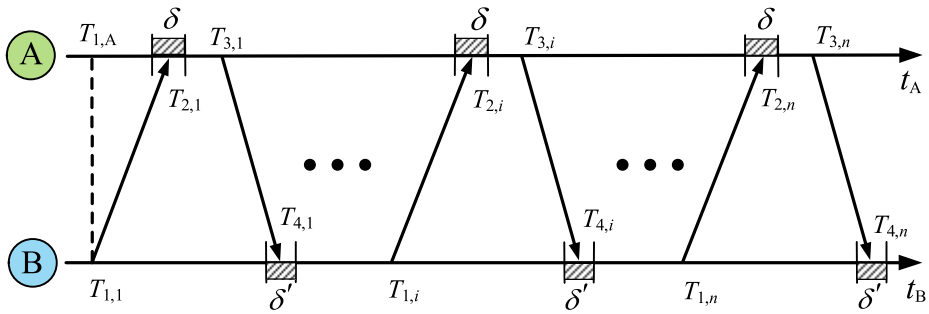


FIGURE 1. Random delay in timestamp message exchange.

time stamps of node B and node A, and $T_{2,i}$ and $T_{4,i}$ are receiving time stamps of node B and node A, respectively. Since there is random delay in the transmission and processing of time messages, the time differences $T_{2,i}$-$T_{1,i}$ is also random when executing multiple bidirectional time message exchanges. For example, as shown in Fig. 1, $T_{2,i}$ may be located in the interval with the width of δ on the time axis of node A, Similarly, $T_{4,i}$ may be located in the interval with the width of $\delta\prime$ on the time axis of node B. Equation (1) and (2) can be obtained by introducing random delays into the phase difference calculation formula [19],

$$T_{2,i} = T_{1,i} - \Delta\varphi + d^{BA} + X_i^{BA} \tag{1}$$
$$T_{4,i} = T_{3,i} + \Delta\varphi + d^{AB} + X_i^{AB} \tag{2}$$

where $\Delta\varphi$ is the phase difference between node A and B, d^{BA} and X_i^{BA} represent the fixed and random delays of the time message from B to A, d^{AB} and X_i^{BA} represent the delays from A to B, respectively. Due to the same hardware and software architecture of sensor nodes, it can be considered that $d^{AB} = d^{BA} = d$. [19] and [20] assume that both X_i^{BA} and X_i^{BA} obey the same PDF (Probability Density Function), therefore, the mean of X_i is 0, and its variance is twice as much as X_i^{BA}. However, there is a phenomenon of synchronous error accumulation in the practical application of WSNs, that is, both mean and variance of X_i tend to increase with the expansion of network scale, thus the above assumption is difficult to meet. In practice, there are random differences in the microenvironment of different nodes, for example, in WSNs used to monitor the intertidal zone, the clock frequency of nodes will be affected by ambient temperature [21], which will lead to small changes in time message receiving and sending delay. Therefore, in this paper, we assume that X_i is a non-zero random variable. The delay corresponding to the ith time message exchange is defined by

$$U_i = T_{2,i} - T_{1,i} \tag{3}$$
$$V_i = T_{4,i} - T_{3,i} \tag{4}$$

From (1),(2),(3) and (4), the real phase difference between nodes is given by

$$\Delta\varphi = \Delta\hat{\varphi}_i - X_i \tag{5}$$

where X_i is given by

$$X_i = (X_i^{AB} - X_i^{BA})/2 \tag{6}$$

and $\Delta\hat{\varphi}_i = (V_i - U_i)/2$, which is the estimation for real phase difference $\Delta\varphi$ calculated with the time message in the ith round. From (5), the real phase difference is $\Delta\varphi$, however, node B cannot perceive any information about X_i, thus the clock difference can be compensated only by $\Delta\hat{\varphi}_i$, which will result in synchronization errors. Rewrite (5), we have $X_i = \Delta\hat{\varphi}_i - \Delta\varphi$, and its form is consistent with the definition of measurement error in [22]. Since the term synchronization error is widely used in the field of clock synchronization research, we define X_i as the clock synchronization error in this paper. Various kinds of PDF have been proposed to model X_i, such as gamma distribution, exponential distribution and single server M/M/1 queue model. In this paper, Gaussian distribution is selected, since the random delay of time message delivery is composed of a series of small delays in the transmission path. According to the central limit theorem, its sum converges to Gaussian distribution [23]. The rationality of this assumption is verified in [24] and [25].

III. BETS ALGORITHM

This paper presents the BETS algorithm based on Bayesian estimation, which uses experiment to obtain prior information of synchronization error compensation parameters. In practice, only a small amount of time message exchange is required to complete the correction and compensation of environment-induced errors. Meanwhile, the gradient descent method is used to estimate the relative clock drift rate, which is helpful for setting the optimal execution cycle of the synchronization algorithm and ensures the long-term clock synchronization of the network.

A. ESTIMATION OF SYNCHRONIZATION ERROR COMPENSATION

The BETS uses the prior distribution of synchronization error and on-site time message samples of WSNs to calculate the posterior distribution, and its parameter is used to compensate for the actual running synchronization error accurately. The prior distribution parameters can be obtained by the following

methods: assuming that there are n nodes in the network, clock synchronization tests are carried out for all combinations of nodes (a total of C2 n) and synchronization errors are measured. With the measurement data, parameters of prior distribution $\pi(\theta)$ can be calculated according to [26]. On-site samples of time messages can be obtained according to the message transmission and reception method shown in Fig. 1. From Bayesian estimation theory, the accuracy of posterior distribution is higher than that of prior distribution, and the posterior distribution of parameter θ is give by

$$\pi(\theta|\boldsymbol{x}) = \frac{h(\boldsymbol{x},\theta)}{m(\boldsymbol{x})} = \frac{p(\boldsymbol{x}|\theta)\pi(\theta)}{\int_{\Theta} p(\boldsymbol{x}|\theta)\pi(\theta)d\theta}$$

where $\boldsymbol{x} = \{x_i\}_{i=1}^n$ is the sample set, $\pi(\theta)$ is the prior distribution of θ, $h(\boldsymbol{x}, \theta)$ is the joint distribution of samples and θ, $m(\boldsymbol{x})$ is marginal distribution of samples $\boldsymbol{x}$, $p(\boldsymbol{x}|\theta)$ is joint conditional probability density function.

Before executing the BETS algorithm, the first synchronization is performed by bidirectional time message exchange, and the synchronization error is compensated by the mean of prior distribution $\pi(\theta)$. It can be considered that the compensated node clock difference $\Delta\varphi \approx 0$. As the random delay changes, in order to get the effect of this change on the original synchronization error, BETS obtains the time message sample set $\{T_{1,i}, T_{2,i}, T_{3,i}, T_{4,i}\}_{i=1}^n$ through n rounds of sampling as shown in Fig. 1. From (1), (2) and (6), the synchronization error samples caused by random delay variation is given by

$$\boldsymbol{x} = \{x_i | x_i = (V_i - U_i)/2\}_{i=1}^n \tag{7}$$

where x_i is the sample value of X_i in (6). From Section 3, the clock synchronization error $X_i \sim N(\theta, \sigma^2)$, when assuming the prior distribution of θ is Gaussian, i.e., $\theta \sim N(\mu, \tau^2) = \pi(\theta)$. With joint conditional PDF $p(\boldsymbol{x}|\theta)$ obtained form (7) and prior distribution $\pi(\theta)$, the joint distribution $h(\boldsymbol{x}, \theta)$ is given by

$$h(\boldsymbol{x},\theta) = \pi(\theta)p(\boldsymbol{x}|\theta) = (2\pi)^{-1/2}\tau^{-1}\exp\left\{-\frac{(\theta-\mu)^2}{2\tau^2}\right\} \times (2\pi^{1/2}\sigma)^{-n}\exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i-\theta)^2\right\} \tag{8}$$

Let $\bar{x} = \sum_{i=1}^{n} \frac{x_i}{n}$ and $\sigma_0^2 = \sigma^2/n$,then the joint distribution of sample and parameter is given by

$$h(\boldsymbol{x},\theta) = D\exp\left\{-\frac{1}{2}[A\theta^2 - 2\theta B + C]\right\}$$

where $D = (2\pi)^{-(1+n)/2}\tau^{-1}\sigma^{-n}, A = \frac{1}{\tau^2} + \frac{1}{\sigma_0^2}, B = \frac{\mu}{\tau^2} + \frac{\bar{x}}{\sigma_0^2}$, $C = \frac{\mu^2}{\tau^2} + \frac{1}{\sigma^2}\sum_{i=1}^{n} x_i^2$.

Calculating the marginal distribution $m(\boldsymbol{x})$ gives

$$m(\boldsymbol{x}) = \int_{-\infty}^{\infty} h(\boldsymbol{x},\theta)d\theta = D(2\pi/A)^{1/2}\exp\left\{-\frac{1}{2}(C - B^2/A)\right\}$$

Let $h(\boldsymbol{x}, \theta)$ divided by $m(\boldsymbol{x})$, the posterior distribution is given by

$$\pi(\theta|\boldsymbol{x}) = (2\pi/A)^{-1/2}\exp\left\{-\frac{(\theta - B/A)^2}{2/A}\right\}$$

This formula conforms to the expression of Gaussian distribution, therefore, the mean of posterior distribution is given by

$$\mu' = \frac{B}{A} = \frac{\mu\sigma_0^2 + \bar{x}\tau^2}{\sigma_0^2 + \tau^2} = \frac{\mu\sigma_0^2 + \frac{\tau^2}{2n}\sum_{i=1}^{n}(V_i - U_i)}{\sigma_0^2 + \tau^2} \tag{9}$$

where μ and τ are known parameters of $\pi(\theta)$, which can be obtained by experiments and calculation; σ2 0can be obtained by V_i, U_i and standard deviation formula in Statistics. It is clear that, from (9), μ' can be regarded as a weighted average between μ and the amount of change in the synchronization error reflected by the time message.

When using μ' as the Bayesian estimation of θ, the Mean Square Error (MSE) is a common criterion for evaluating the estimation effect and is given by

$$\begin{aligned} E[(\theta - \hat{\theta})^2] &= E[(\theta - \hat{\theta}_E) + (\hat{\theta}_E - \hat{\theta})]^2 \\ &= Var(\theta|\mathbf{x}) + (\hat{\theta}_E - \hat{\theta})^2 \end{aligned} \tag{10}$$

where $\hat{\theta}$ is the Bayesian estimation of θ and $\hat{\theta}_{\mathrm{E}} = \mu'$, which is the posterior expectation mean of θ. From (10), It can be seen that MSE reaches the minimum when $\hat{\theta} = \hat{\theta}_E$, thus μ' is selected as the optimal estimation of the synchronization error in BETS.

B. ESTIMATION OF CLOCK SYNCHRONIZATION EXECUTION PERIOD

By estimating and compensating for the synchronization errors, local clocks of network node can be adjusted to be consistent. However, with the influence of clock drift and other factors, the calibrated clocks will gradually generate cumulative errors, thus the synchronization need to be executed periodically. However, if the synchronization period T is too long, it will lead to temporary loss of synchronization in the WSNs. On the contrary, if the period T is too short, limited energy and computing resources will be consumed rapidly by frequent execution of synchronization algorithm. The clock synchronization execution period T is defined as

$$T = \Delta\varphi_{max}/\omega_{max} \tag{11}$$

where $\Delta\varphi_{max}$ represents the maximum allowed clock difference in the network, and ω_{max} is the maximum relative clock drift rate between network nodes, i.e., $\omega_{max} \geq \omega_{i,i+1}$, and $\omega_{i,i+1}$ $(0 \leq i \leq N-1)$ is the relative clock drift rate between node i and node $i+1$. If $\Delta\varphi_{max}$ is determined, the synchronization period will be determined by ω_{max}.

Fig. 2 shows that when synchronization is completed, with the influence of clock drift, the relationship between clocks

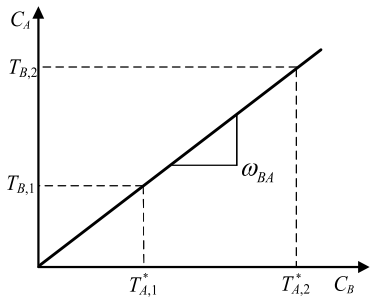


FIGURE 2. Relative clock drift rates of nodes *A* and *B*.

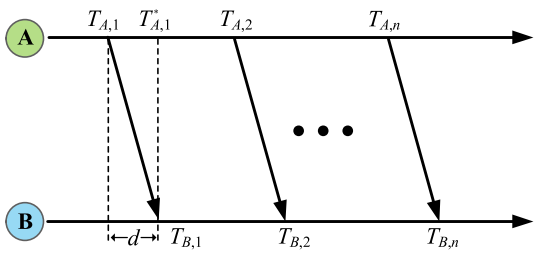


FIGURE 3. Time stamp sampling process.

of node A and B can be expressed as a linear equation, which is

$$C_B = \omega_{BA} C_A \tag{12}$$

where C_A and C_B are the clock values of nodes A and B, respectively. ω_{BA} is the relative clock drift rate between node B and node A. In order to estimate the relative clock drift rate, it is necessary to sample continuously varying node clocks to obtain discrete clock values. Defining $T_{A,i}$ as the node A clock value in the ith sampling and $T_{B,i}$ as the node B clock value at the same time. The clock sample set $\{T_{A,i}, T_{B,i}\}_{i=1}^{n}$ can be obtained after n times of sampling as shown in Fig. 3.

Considering that the message propagation delay d needs to be compensated, after the time synchronization is completed, it can be considered that $C_A = C_B$. At this moment, node A sends the first time message and $d = T_B, 1 - T_{A,1}$, let $T^*_{A,i} = T_{A,i} + d(1 \le i \le N)$, where $T^*_{A,i}$ is the clock value of node A at time $T_{B,i}$. After sampling for n times, ω_{BA} could be estimated with the compensated sample set$\{T^*_{A,i}, T_{B,i}\}_{i=1}^{n}$. Assuming that the clock difference between node pair has been adjusted to be consistent by last time synchronization, the node B clock represented by a linear equation is given by

$$C_B = \omega_{BA} T^*_{A,i}.$$

Defining the cost function as

$$J(\omega_{BA}) = \frac{1}{2n} \sum_{i=1}^{n} (C_B - T_{B,i})^2 \tag{13}$$

Which represents the square error function between the estimated value and the sampled value, the value of ω_{BA} that minimize this function is the optimal parameter. Differentiating (13) with respect to ω_{BA} gives

$$\begin{aligned} \frac{dJ(\omega_{BA})}{d\omega_{BA}} &= \frac{d}{d\omega_{BA}} \left[\frac{1}{2n} \sum_{i=1}^{n} (\omega_{BA} T^*_{A,i} - T_{B,i})^2 \right] \\ &= \frac{1}{n} \sum_{i=1}^{n} T^*_{A,i} (\omega_{BA} T^*_{A,i} - T_{B,i}) \end{aligned} \tag{14}$$

With the initial value of ω_{BA}, the gradient α and the convergence condition, iteration can be performed as

$$\omega^+_{BA} = \omega_{BA} - \alpha \frac{dJ(\omega_{BA})}{d\omega_{BA}} \tag{15}$$

where ω^+_{BA} represents the updated value of ω_{BA} after each iteration, the iteration continues until the difference between ω^+_{BA} and ω_{BA} is smaller than convergence condition, and final value of ω_{BA} is used as the estimation of relative clock drift rate.

IV. PERFORMANCE ANALYSIS OF BETS

A. SYNCHRONIZATION ERROR VARIANCE COMPARISON

In addition to the mean value of synchronization error, the algorithm performance metric also includes the synchronization error variance, which describes the dispersion degree of node clocks after synchronization. The smaller the variance is, the better the uniformity of the clock synchronization errors are. In the parameter estimation, the performance lower bound indicates the optimal accuracy in the sense of dispersion degree that an algorithm could achieve. In [19] and [20], Cramer-Rao Lower Bound (CRLB) for MLE and its improved algorithm LCMLE is derived, which is $\sigma^2/2n$. Since there are some differences between Bayesian estimation and MLE algorithm, CRLB cannot be directly applied to Bayesian estimation as a performance criterion. Therefore, assuming that the clock synchronization error is Gaussian, this section derives Bayesian Cramer-Rao Bound (BCRB) for BETS, and analyses its performance.

Similar to CRLB, when regular conditions are satisfied, the Bayesian information matrix J_B is given by

$$J_B = J_D + J_P \tag{16}$$

For the convenience of calculation, J_B is decomposed into J_D and J_P [27], where J_D term represents the contribution of the sampled data and the J_P term represents the contribution of the prior information. The J_D term is derived first, calculating the natural logarithm of $p(\boldsymbol{x}|\theta)$ in (8) gives

$$L(\theta) = -\frac{n}{2} \ln(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (x_i - \theta)^2 \tag{17}$$

Differentiating the logarithm of (17) with respect to θ gives

$$\frac{\partial L(\theta)}{\partial \theta} = \frac{n(x_i - \theta)}{\sigma^2}$$

The Fisher information is given by

$$J_F(\theta) = -E\{\frac{\partial^2 L(\theta)}{\partial \theta^2}\} = n/\sigma^2$$

Finally, the J_D is given by

$$J_D = E\{J_F(\theta)\} = n/\sigma^2 \tag{18}$$

Then, Calculating the natural logarithm of $\pi(\theta)$ in (8) gives

$$\ln \pi(\theta) = -\frac{n}{2}\ln(2\pi\tau^2) - \frac{(\theta-\mu)^2}{2\tau^2} \tag{19}$$

Differentiating the logarithm of (19) with respect to θ gives

$$\frac{\partial L(\theta)}{\partial \theta} = \frac{\mu-\theta}{\tau^2}$$

Finally, the J_P is given by

$$J_P = -E\{\frac{\partial^2 \ln \pi(\theta)}{\partial \theta^2}\} = 1/\tau^2 \tag{20}$$

From (16), (18) and (20), the BCRB for the posterior distribution mean μ' is given by

$$\mathrm{var}(\mu') \geq J_B^{-1} = \frac{\sigma^2\tau^2}{n\tau^2+\sigma^2} \triangleq \mathrm{BCRB} \tag{21}$$

From (21), BCRB is a monotonic decreasing function of the number of time message. The smaller the BCRB is, the higher the theoretical estimation accuracy of the algorithm could achieve. Since σ (variance of synchronization error) is usually much larger than τ, which is the variance of the average of σ, hence, when $\sigma/\tau > \sqrt{n}$, the relationship between BCRB and CRLB is given by (22)

$$\mathrm{BCRB} \triangleq \frac{\sigma^2\tau^2}{n\tau^2+\sigma^2} < \frac{\sigma^2\tau^2}{2n\tau^2} = \frac{\sigma^2}{2n} \tag{22}$$

If variable to be estimated is Gaussian, (22) shows that the BCRB of the Bayesian estimator is lower than CRLB of MLE, showing that the BETS algorithm can effectively improve the estimation accuracy by making use of the prior distribution information.

B. COMPLEXITY COMPARISON

Since the BETS algorithm is only executed between a pair of nodes every time, the computational complexity for sensor nodes is regardless of the network size and is only related to the number of synchronous time messages N.

The computational complexity is dictated by the total number of computations involved in additions and multiplications. From (9), we can see that the complexity of BETS is O(N), which is the same order as LCMLE. However, when comparing the number of computations involved in time message processing, we noticed that LCMLE involves 3 sub-algorithms, and the complexity of each sub-algorithm is O(N), thus its total computational cost is 3 times that of BETS in the worst case, which indicates that the BETS could achieve lower complexity.

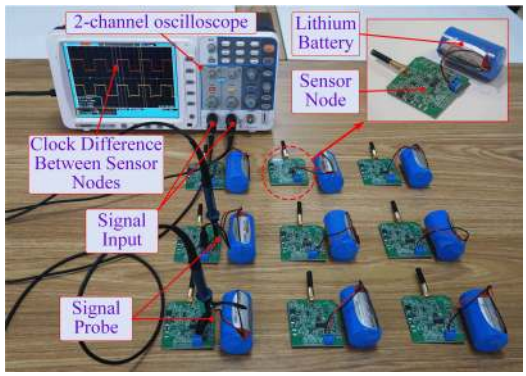


FIGURE 4. Test platform of WSNs.

V. EXPERIMENT AND PERFORMANCE TEST

To evaluate the performance of BETS algorithm, we have conducted simulation and outdoor experiments based on hardware platform. The simulation was performed in MATLAB, the network was organized in a linear topology with 100 hops [2]. In simulation, we tested the compensation effect of BETS with variation of synchronization errors. In the outdoor experiment, the hardware of WSNs node is implemented using STM8 low-power MCU. The RF transceiver uses Si4438 which meets IEEE 802.15.4 standard, and its wireless communication rate is 115.2 kbps. A 16 MHz oscillator is used as the clock source. The photo of sensor nodes in our test is shown in Fig. 4. On the hardware platform, we measured parameters of prior distribution of synchronization error mean and performed outdoor clock synchronization.

The BETS was compared with the other four algorithms, including TPSN, TSICA, Static Compensation (SC) and LCMLE algorithm. TPSN achieves clock synchronization by performing bidirectional time message exchange once; TSICA uses one-way time message diffusion to achieve multi hop network synchronization, and improves the accuracy of clock parameter estimation by optimizing the timing of local clock value adjustment; SC uses the estimation of prior distribution mean to compensate synchronization error; LCMLE is an improved algorithm based on MLE, which is characterized in that the estimation accuracy of the clock phase difference is improved by multiple bidirectional time message exchanges.

The results are analyzed from three aspects, (1) Synchronization error: including average synchronization error and its standard deviation; (2) number of time messages: the number of messages sent and received by BETS was tested; (3) effect of parameter selection on relative clock drift rate estimation: convergence analysis and number of iteration were tested when choosing different parameters.

A. MEASUREMENT OF PRIOR DISTRIBUTION PARAMETERS

In BETS, it is necessary to determine the parameters of prior distribution, which is susceptible to ambient temperature. In order to compensate synchronization errors accurately at

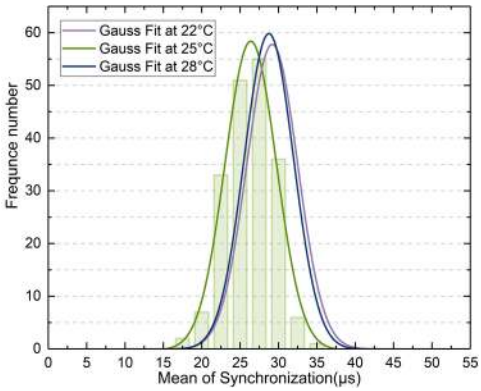


FIGURE 5. Distribution of the mean of synchronization error.

different temperatures, a prior distribution family should be measured. Let *Tenv* $i(1\leq\ i\ \leq\ N)$ represents the ambient temperature when the prior distribution is measured. In the interval [*Tenv 1*, *Tenv n*], the difference between adjacent discrete temperature is constant, which is given by

$$\Delta T^{env} = T =_{i+1}^{env} - T_i^{env}$$

By setting different temperatures in the lab, the prior distribution family corresponding to the different temperature can be measured and is given by: $N_1(\mu_1, \tau_1^2), \ldots, N_n(\mu_n, \tau_n^2)$. The parameters of the prior distribution family will be stored in the sensor nodes. When performing BETS in the deployment site, the real-time temperature will be measures by the on-chip temperature sensor [28], and distribution parameter closest to the current temperature should be selected for error compensation.

Extensive tests are required for measuring the parameters of priori distribution family, in order to evaluate the effectiveness of BETS, a priori distribution family of synchronization error mean θ at 22°C, 25°C and 28°C was measured. In order to reduce the influence of node hardware difference on the test, we use 20 nodes in the test and measured synchronization errors of all combinations of node pairs (C_{20}^2), and this method is also adopted by [10]. After synchronization, a positive pulse signal with a period of 500 ms will be output from the node IO, by measuring the difference of the rising edge of pulses with a digital oscilloscope, clock synchronization error can be obtained.

Each node pair was tested for 5 times and averaged. The results are shown in Tables 1 and Fig. 5. Table 1 shows the frequency of synchronization error mean θ at 25°C. It can be seen that θ is concentrated in the range of 22.5 μs to 30 μs. In Fig. 5, horizontal axis represents the synchronization error mean, and vertical axis represents the corresponding frequency number. Experimental results at three different ambient temperatures were performed by Gaussian fitting. From the curve, it is reasonable to assume that θ obeys Gaussian distribution. According to the median values of each interval and the corresponding frequency in Table 1, the estimation of

TABLE 1. Results of the mean of synchronization error.

Median Value	Frequency Number	Frequency
17.5	2	0.010
20	7	0.037
22.5	33	0.174
25	51	0.268
27.5	55	0.289
30	36	0.189
32.5	6	0.032
35	1	0.005

parameter μ at 25°C (represented by $\hat{\mu}_{25}$) is given by

$$\hat{\mu}_{25} = 17.5 \times 0.010 + \cdots + 35 \times 0.005 \approx 26.4\mu s(25°\text{C})$$

Similar calculation gives

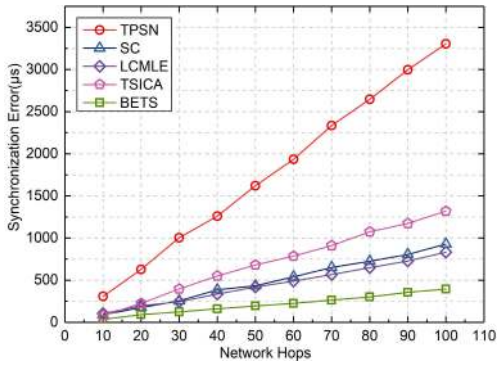
$$\hat{\mu}_{22} = 28.7\mu s(22°\text{C}), \quad \hat{\mu}_{28} = 29.1\mu s(28°\text{C})$$

B. SIMULATION OF BETS ALGORITHM

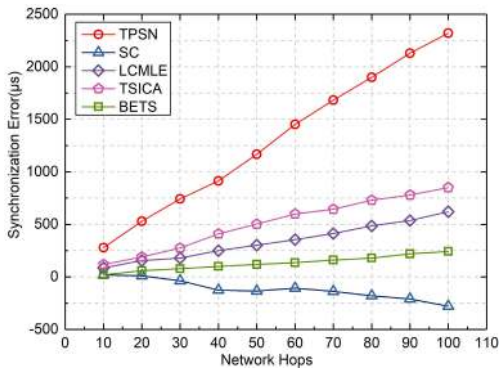
To simulate clock synchronization in large-scale WSNs, network hops is set to 100, network nodes are numbered from 0 to 100 and the clock of node 0 is defined as the reference. The synchronization error of the *i*th hop is defined as the clock difference between node *i* and node 0 after synchronization is completed. The clock model is built by Simulink modules in MATLAB, both clock difference calculation (by LCMLE and TSICA) and error compensation parameter adjustment (by BETS) can be conducted with the timestamp information extracted from this model. The clock runs at 16 MHz and the drift rate is 30 PPM. The initial clock differences of sensor nodes obey uniform distribution in the interval [-10 s, 10 s]. By adding Gaussian noises ($N(\mu d, \sigma_d^2)$) in the clock model, clock synchronization with random delay variation can be simulated. In the following three scenarios, the parameters of noise and BETS algorithms are set as follows.

scenario A: $\mu_d = 35\ \mu$s; scenario B: $\mu_d = 20\ \mu$s; scenario C: in the first 50 hops, $\mu_d = 35\ \mu$s, in last 50 hops $\mu_d = 20\ \mu$s. $\sigma_d = 15\ \mu$s in all scenarios. The parameter of BETS was set to $\hat{\mu}_{25}$, which is the mean of prior distribution measured at 25°C. In simulation, time message sampling will be taken for 10 times when adjacent nodes perform synchronization. The tests are performed for 5 times in each scenario and the results are averaged, which are shown in Fig. 6(a), (b), (c) and Table 2.

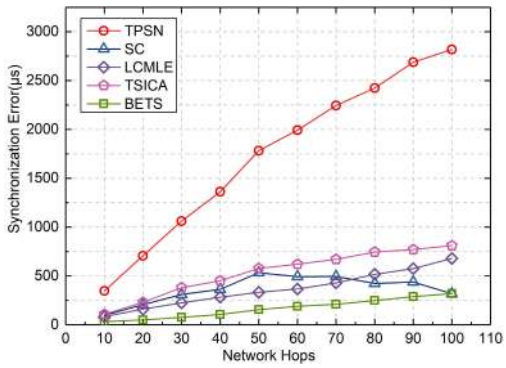
In scenario A, due to the lack of estimation and error compensation mechanism in TPSN, its accumulated synchronization errors increase linearly and the total synchronization errors after 100 hops reaches 3307 μs. TSICA will adjust the clock value before estimating the clock parameters, thus the synchronization error could be limited within a bounded range. However, when estimating the clock parameters, the TSCA fails to make full use of the statistical information to improve the synchronization accuracy. Therefore, its synchronization error accumulation is only lower than TPSN, which is 1318 μs.



(a). *Test results for scenario* A



(b). *Test results for scenario* B



(c). *Test results for scenario* C

FIGURE 6. (a). Test results for scenario A. (b). Test results for scenario B. (c). Test results for scenario C.

In static compensation, the accumulated error is 929 μs, which is about 72% less than that of TPSN. The accumulated error of LCMLE (832 μs) is close to that of static compensation, however, the error curve of LCMLE is much smoother, showing that its standard deviation of synchronization error is smaller. The accumulated error of BETS is 393 μs, which decreased by 53% compared with LCMLE, and its standard deviation of synchronization error is 2.2 μs, which is about half of that of the LCMLE (4.5 μs). Compared with other algorithms, the synchronization error accumulation and standard deviation of BETS are smaller.

In scenario B, since the static compensation only uses the constant $\hat{\mu}_{25}$ to compensate for the synchronization error, If the actual synchronization error was less than the preset

TABLE 2. Comparison of average single hop synchronization error.

		Scenario A	Scenario B	Scenario C
Mean (μs)	NC	33.1	23.2	28.7
	SC	9.3	-2.8	3.2
	TSICA	13.1	8.1	8.5
	LCMLE	8.3	6.2	6.8
	BETS	3.9	2.4	3.0
Standard deviation (μs)	NC	14.9	10.8	13.9
	SC	14.9	10.8	13.9
	TSICA	5.7	7.2	5.3
	LCMLE	4.5	8.2	6.7
	BETS	2.2	2.0	2.1

error compensation parameter, an over-compensation could occur. Although the average single-hop error of static compensation is the smallest in this scenario, but its standard deviation of error is larger. The BETS can effectively use the time message to dynamically adjust the error compensation parameters, which avoids the problem of overcompensation, and its synchronization error accumulation is less than that of LCMLE and TSICA.

In scenario C, when the synchronization error between the first 50 hops and the last 50 hops of the network changes, the static compensation and TSICA cannot handle this change, therefore, the synchronization error accumulation curve of fluctuates before and after 50 hops.

Since the precision of LCMLE is related to the processing delay of time message, its synchronization errors also change correspondingly, but its error fluctuation was smaller than static compensation and TSICA. Similar to scenario B, due to fine adaptability, the error fluctuation of BETS is minimal.

In the above three scenarios, LCMLE could reduce both the average and standard deviation of synchronization error compared with static compensation. However, its synchronization accuracy depends on the processing accuracy of time messages. Since there are some random delays (such as ISR processing delay) that could be hardly estimated, which limits the improvement of accuracy in LCMLE at hardware level, however. Nevertheless, its performance is still better than TSICA.

By utilizing the prior information of the synchronization error and adjusting the synchronization error compensation parameter, the BETS could overcome the influence of above-mentioned unfavorable factors and achieves superior accuracy.

The accuracy of clock synchronization algorithm is generally related to the number of samples, so we compared the synchronization errors of BETS, LCMLE and TSICA as the number of time messages is changed. The simulation settings are the same as those in scenario A. As can be seen from Fig.7, the BETS has lower synchronization error when sending and receiving the same number of time messages. When the number of time messages is 3, the synchronization error of LCMLE is 12.7 μs and those of TSICA and BETS are 11.3 μs and 8.9 μs, respectively. With the increase of number of time messages, the synchronization error of both algorithms decreases gradually. With the number of time messages

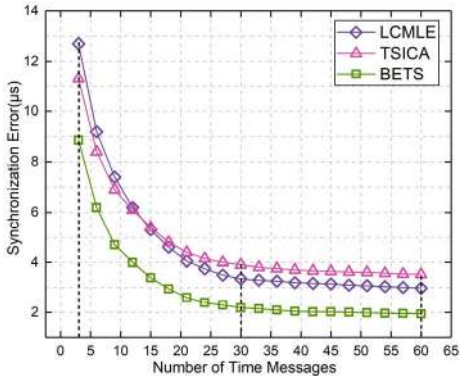


FIGURE 7. Comparison of the number of time messages.

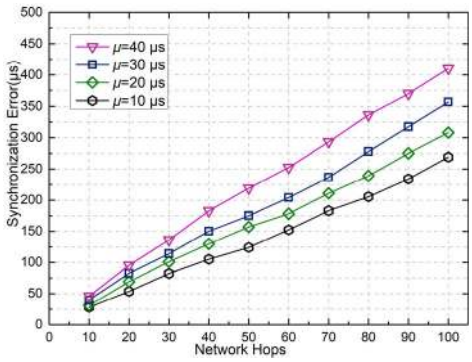


FIGURE 8. The effect of μ_d on synchronization error.

increases to 30, the synchronization error of LCMLE decreases to 3.4 μs, and those of TSICA and BETS are 3.9 μs and 2.2 μs, respectively. However, as the number of time messages exceeds 30, the synchronization accuracy barely improves, for example, When the number of time messages is 60, the synchronization error of LCMLE is 3.1 μs, and those of TSICA and BETS are 3.5μs and 1.9μs, respectively.

Fig. 8 shows the accumulated synchronization errors in BETS when μ_d increases from 10 μs to 40μs. Compared with LCMLE, although the BETS is adaptable to the variation of μ_d, the test results shows the limitation of dynamic adjustment ability of synchronization error. With the increase of μ_d, the synchronization error curve also moves up. In practical applications, the parameters closest to the current ambient temperature should be selected from the prior distribution family to maximize the compensation accuracy and reduce the influence on synchronization accuracy if the variation of μ_d is drastic.

C. OUTDOOR EXPERIMENT OF BETS

To evaluate the BETS in real scenarios, we performed outdoor test. Due to the limitation of experimental conditions, it is difficult to set up the large-scale network in practice. Thus, a linear WSNs with 12 sensor nodes is constructed in the experiment, which includes 11 hops. We deploy sensor nodes alone the river bank (Fig. 9) and the average distance between node pair is about 130 meters. On the day of conducting the

FIGURE 9. Outdoor deployment topology of sensor nodes.

experiment, the outdoor temperature ranged from 21°C to 30°C. Due to different start time, there are random initial clock differences among sensor nodes. When all nodes are started, node 0 will send a synchronization request message to node 1, as soon as node 1 has received this message, it synchronizes with node 0, and subsequent sensor nodes synchronizes in the same manner.

TPSN, TSICA and LCMLE are taken as the comparison, synchronization experiments are carried out at around 25°C, The parameter of BETS is set to $\hat{\mu}_{25}$, which is the mean of prior distribution measured at 25°C in the lab. The results are shown in Fig. 10. the accumulated error of TPSN is 403 μs, and those of TSICA and LCMLE are 152 μs and 108 μs respectively. From the fluctuation of error curve in the Fig. 10, it can be seen that the standard deviation of synchronization error of three other algorithms are also smaller compared with static compensation. Compared with LCMLE, the accumulated error of BETS (48.4μs) is significantly reduced. Thus, the outdoor test shows that the accumulated synchronization errors in multi-hop WSNs can be further reduced by selecting appropriate error compensation parameters.

From Fig. 7, it is shown that when the number of time message exceeds 30, the accuracy of synchronization will be hardly improved. Thus, in the actual experiment, the maximum number of time message sending and receiving is set to 30. Fig. 11 shows the relationship between the number of time messages and the accuracy of clock synchronization in practical experiments. It can be seen that the error curve of BETS lies in the lower left of Fig. 11, indicating that the BETS requires less time messages when achieving similar synchronization accuracy of LCMLE and TSICA. As the synchronization error is approximately 8μs, the BETS needs to send and receive 6 time messages, while the number for LCMLE and TSICA is 12. If the synchronization error needs to be reduced to 6μs, the BETS will have to send and receive 9 time messages, while the number for LCMLE and TSICA are 18 and 24, respectively.

Since the variation of ambient temperature may lead to a slight change in actual synchronization error, we have performed synchronization experiments at different ambient temperatures, and different prior distribution parameters are selected to investigate its impact on synchronization

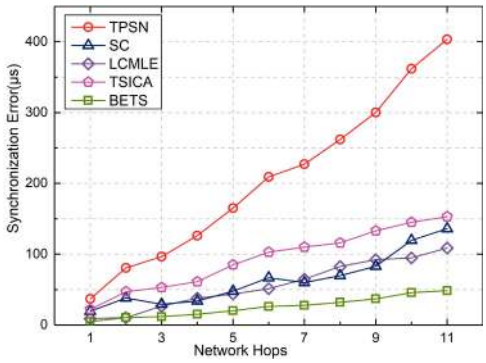


FIGURE 10. Test results for synchronization error.

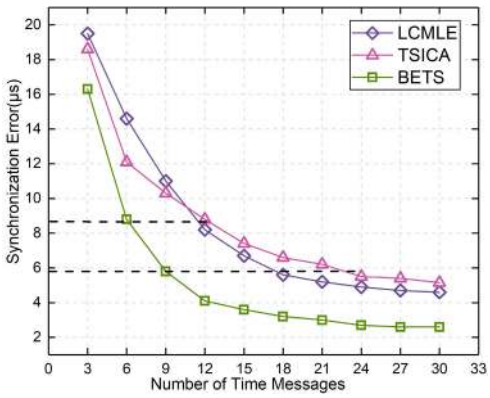


FIGURE 11. Comparison of the number of time messages.

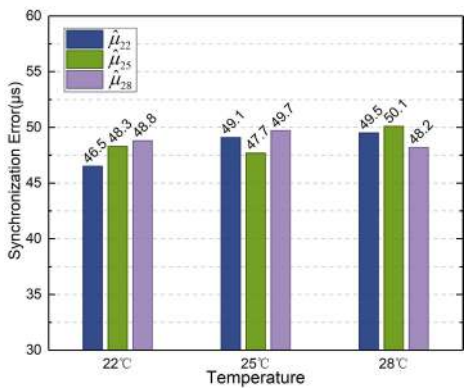


FIGURE 12. Synchronization error at different temperatures.

accuracy. Fig. 12 shows the synchronization results at 22°C, 25°C and 28°C with prior parameters set to $\hat{\mu}_{22}$, $\hat{\mu}_{25}$ and $\hat{\mu}_{28}$ at each temperature. The 11-hop cumulated synchronization error for each ambient temperature are labeled in Fig. 12 in μs. It can be seen that the minimal error will be achieved if the selected prior distribution parameter matches the current ambient temperature. For example, if the parameter of the BETS is set to $\hat{\mu}_{25}$ the least accumulated error(47.7μs) will be achieved at 25°C. No matter whether the ambient temperature rises or falls, the accumulation of synchronization error would increase slightly to 49.1μs (22°C) and 49.7μs (28°C), respectively. Therefore,

TABLE 3. The relationship between gradient and number of iterations.

Gradient value ($\times 10^{-8}$)	Number of iterations
1.0	17
1.5	11
2.0	6
2.5	4
3.0	5
3.5	9
4.0	15

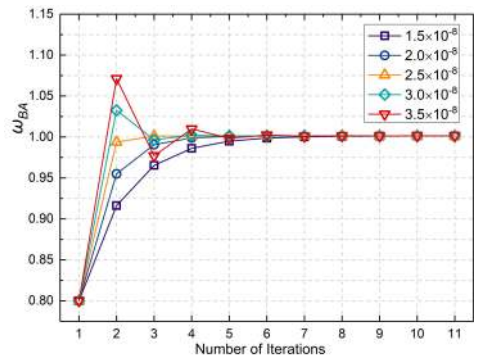


FIGURE 13. Convergence with different gradient settings.

in order to achieve the optimal synchronization accuracy, sensor nodes should select appropriate parameters to set BETS algorithm according to the ambient temperature measured in real time.

D. RELATIVE CLOCK DRIFT RATE ESTIMATION

In order to calculate the clock synchronization execution period, this section estimates the relative clock drift rate with gradient descent method. The experiment was carried out on sensor node hardware. Node A sends time messages to node B every 1ms for 10 times. Finally, the time stamps recorded by nodes will be uploaded to PC for analysis.

The performance of the algorithm is closely related to the selection of parameters. Fig. 13 shows the convergence of gradient descent method when the convergence condition is $\omega_{BA}^{+} - \omega_{BA} \leq 1 \times 10^{-6}$, where ω_{BA}^{+} is the updated value of ω_{BA} after each iteration. When the gradient $\alpha < 2.5 \times 10^{-8}$, ω_{BA} will converge unidirectionally after each iteration; when $\alpha > 2.5 \times 10^{-8}$, ω_{BA} will oscillate as it converges, and the number of iterations will increase at the same time. Further experiments showed that if $\alpha > 4.5 \times 10^{-8}$, ω_{BA} will unable to converge. Table 3 shows the relationship between number of iterations and value of gradient. When $\alpha = 4 \times 10^{-8}$, 15 iterations are needed. Whereas if $\alpha = 2.5 \times 10^{-8}$, only 4 iterations will ensure the convergence, when α is larger or smaller than this optimal value, the number of iterations will increase.

Considering the convergence condition and clock drift rate in actual senor nodes, we chose the following parameters to estimate the relative clock drift rate: the initial value of ω_{BA} is 0.8, the gradient $\alpha = 2.5 \times 10^{-8}$, and the convergence

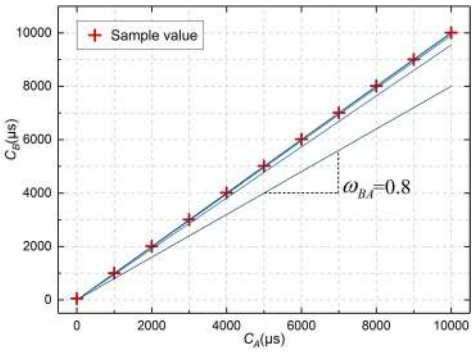


FIGURE 14. The relationship between the number of iterations and ω_{BA} ($\alpha = 2.5 \times 10.8$).

condition is $\omega_{BA}^{+} - \omega_{BA} \leq 1 \times 10^{-6}$. Fig. 13 shows the variation of ω_{BA} after each iteration. The horizontal and vertical axes represent the clock values of nodes A and B, respectively. For the convenience of analyzing, the time when node B received the first time message was set to 0. After each iteration, ω_{BA} (the slope of the line in Fig. 14) continues to increase to fit the sampled time data. After 4 iterations, the convergence condition was reached, and finally $\omega_{BA} = 1.000008$. From the conversion, we can know that the clock difference caused by clock drift between sensor nodes A and B will increase by about 28.8 ms per hour.

Defining the relative drift rate between adjacent node i and node $i+1$ as $\omega_{i,i+1}(0 \leq i \leq n)$, and the estimation for $\omega_{i,i+1}$ will be performed along the synchronization route. Finally, $\omega_{\max} = \mathrm{Max}\{\omega_{i,i+1}\}_{i=0}^{n}$ can be obtained by sorting algorithm. Finally, clock synchronization period will be given by (11) when the allowable maximum clock difference $\Delta\varphi_{max}$ is specified according to practical application scenarios.

VI. CONCLUSION

To solve the problem of error accumulation of clock synchronization in multi-hop WSNs, this paper presents a Bayesian estimation-based time Synchronization (BETS) algorithm. The precise compensation of clock synchronization error is realized by combining the prior information of synchronization error and time message sampled from the network deployment site. Experiments based on the simulation and hardware platform show that the BETS could improves the synchronization accuracy effectively, and the number of time messages sent and received is reduced compared with existing algorithms, which is suitable for energy-constrained WSNs. In order to optimize the execution cycle of the synchronization, the gradient descent method is used to estimate the relative clock drift rate, and the convergence of the algorithm is tested with different parameter settings. In this paper, the Gaussian synchronization error is compensated. However, if the network contains complex routing or network congestion occurs, the estimation of clock synchronization error will be more complex. Therefore, it is necessary to study the synchronization error compensation method in the above scenario in the future work.

REFERENCES

[1] L. Shi and A. O. Fapojuwo, "TDMA scheduling with optimized energy efficiency and minimum delay in clustered wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 9, no. 7, pp. 927–940, Jul. 2010.

[2] I. Jawhar, N. Mohamed, and K. Shuaib, "An efficient framework and networking protocol for linear wireless sensor networks," *Ad Hoc Sensor Wireless Netw.*, vol. 7, nos. 1–2, pp. 3–21, 2009.

[3] X. Bai, Z. Wang, L. Sheng, and Z. Wang, "Reliable data fusion of hierarchical wireless sensor networks with asynchronous measurement for greenhouse monitoring," *IEEE Trans. Control Syst. Technol.*, vol. 27, no. 3, pp. 1036–1046, May 2019.

[4] W. Dargie and C. Poellabauer, *Fundamentals of Wireless Sensor Networks: Theory and Practice*. Hoboken, NJ, USA: Wiley, 2010, pp. 236–237.

[5] S. Boubiche, D. E. Boubiche, A. Bilami, and H. Toral-Cruz, "Big data challenges and data aggregation strategies in wireless sensor networks," *IEEE Access*, vol. 6, pp. 20558–20571, 2018.

[6] F. Akyildiz and M. C. Vuran, *Wireless Sensor Networks*. Hoboken, NJ, USA: Wiley,2010.

[7] X. Liu, A. Liu, T. Wang, K. Ota, M. Dong, Y. Liu, and Z. Cai, "Adaptive data and verified message disjoint security routing for gathering big data in energy harvesting networks," *J. Parallel Distrib. Comput.*, vol. 135, pp. 140–155, Jan. 2020.

[8] S. Ping, "Delay measurement time synchronization for wireless sensor networks," *Int. Res. Berkeley Lab*, vol. 6, pp. 1–10, Jun. 2003.

[9] S. Ganeriwal, R. Kumar, and M. B. Srivastava, "Timing-sync protocol for sensor networks," in *Proc. 1st Int. Conf. Embedded Netw. Sensor Syst.*, 2003, pp. 138–149.

[10] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," *ACM SIGOPS Operating Syst. Rev.*, vol. 36, no. 1, pp. 147–163, Dec. 2002.

[11] M. Maróti, B. Kusy, G. Simon, and Á. Lédeczi, "The flooding time synchronization protocol," in *Proc. 2nd Int. Conf. Embedded Netw. Sensor Syst. (SenSys)*, 2004, pp. 39–49.

[12] H. Wang, L. Shao, M. Li, and P. Wang, "Estimation of frequency offset for time synchronization with immediate clock adjustment in multi-hop wireless sensor networks," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 2239–2246, Dec. 2017.

[13] X. Huan, K. S. Kim, S. Lee, E. G. Lim, and A. Marshall, "A beaconless asymmetric energy-efficient time synchronization scheme for resource-constrained multi-hop wireless sensor networks," *IEEE Trans. Commun.*, vol. 68, no. 3, pp. 1716–1730, Mar. 2020.

[14] N. M. Freris, S. R. Graham, and P. R. Kumar, "Fundamental limits on synchronizing clocks over networks," *IEEE Trans. Autom. Control*, vol. 56, no. 6, pp. 1352–1364, Jun. 2011.

[15] J. J. Pérez-Solano and S. Felici-Castell, "Improving time synchronization in wireless sensor networks using Bayesian inference," *J. Netw. Comput. Appl.*, vol. 82, pp. 47–55, Mar. 2017.

[16] Q. Gao, K. J. Blow, and D. J. Holding, "Simple algorithm for improving time synchronisation in wireless sensor networks," *Electron. Lett.*, vol. 40, no. 14, pp. 889–891, 2004.

[17] T. Yang, Y. Zhang, W. Li, and A. Y. Zomaya, "Decentralized networked load frequency control in interconnected power systems based on stochastic jump system theory," *IEEE Trans. Smart Grid*, early access, Mar. 2, 2020, doi: 10.1109/TSG.2020.2978029.

[18] Q. M. Chaudhari, E. Serpedin, and K. Qaraqe, "On minimum variance unbiased estimation of clock offset in a two-way message exchange mechanism," *IEEE Trans. Inf. Theory*, vol. 56, no. 6, pp. 2893–2904, Jun. 2010.

[19] K.-L. Noh, Q. M. Chaudhari, E. Serpedin, and B. W. Suter, "Novel clock phase offset and skew estimation using two-way timing message exchanges for wireless sensor networks," *IEEE Trans. Commun.*, vol. 55, no. 4, pp. 766–777, Apr. 2007.

[20] M. Leng and Y.-C. Wu, "Low-complexity maximum-likelihood estimator for clock synchronization of wireless sensor nodes under exponential delays," *IEEE Trans. Signal Process.*, vol. 59, no. 10, pp. 4860–4870, Oct. 2011.

[21] M. Xu, W. Xu, T. Han, and Z. Lin, "Energy-efficient time synchronization in wireless sensor networks via temperature-aware compensation," *ACM Trans. Sensor Netw.*, vol. 12, no. 2, pp. 1–29, May 2016.

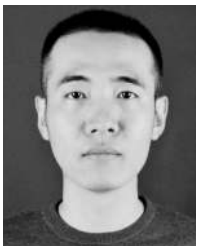
[22] *Guide to the Expression of Uncertainty in Measurement (GUM)*, document JCGM 100:2008, 2008.

[23] B. Fremgen, *Probability and Statistics for Engineers and Scientists*. Upper Saddle River, NJ, USA: Prentice-Hall, 2011, pp. 172–181.

[24] B. Etzlinger, H. Wymeersch, and A. Springer, "Cooperative synchronization in wireless networks," *IEEE Trans. Signal Process.*, vol. 62, no. 11, pp. 2837–2849, Jun. 2014.
[25] B. Yang and M. H. Liao, "A statistic clock synchronization protocol for sensor networks," *J. Harbin Inst. Technol.*, vol. 39, no. 1, pp. 98–101, Jan. 2007.
[26] S. S. Mao, *Bayesian Statistics*. Beijing, China: China Statistics Press, 2012, pp. 69–70.
[27] H. L. Van Trees and K. L. Bell, *Bayesian Bounds for Parameter Estimation and Nonlinear Filtering/Tracking*. New York, NY, USA: Wiley, 2007.
[28] *SI4438 Datasheet*, Silicon Laboratories Co., 2014, pp. 34–36.

TING YANG is currently a Professor with the School of Electrical Engineering and Automation, Tianjin University, Tianjin, China. He has published more than 80 articles in major journals and refereed conference proceedings. He is the chief author or coauthor of four books. His scientific interests include distribution automation, information, and communication systems in smart grid.

Prof. Yang is the Fellow of Theory and Advanced Technology of the Electrical Engineering committee, the Fellow of the Circuit and System committee, and a member of the International Society for Industry and Applied Mathematics.



YUQING NIU received the master's degree in electronic information engineering from the Taiyuan University of Technology, China. He is currently pursuing the Ph.D. degree with Tianjin University.

His research interests include the Internet of Things, wireless sensor networks, and wireless communication technology.

JIEXIAO YU was born in Tianjin, China. She received the B.S. degree in electronic information engineering from Nankai University, in 2002, and the M.S. degree in theory and new technology of electrical engineering and the Ph.D. degree in circuits and systems from Tianjin University, Tianjin, China, in 2005 and 2009, respectively. She is currently an Associate Professor with the School of Electrical and Information Engineering (previously known as the School of Electronic Information Engineering), Tianjin University. Her current research interests include wireless localization, RFID, WSN, and array signal processing.