

Article

RB Particle Filter Time Synchronization Algorithm Based on the DPM Model

Chunsheng Guo ^{1,*}, Jia Shen ¹, Yao Sun ² and Na Ying ¹

¹ College of Communication Engineering, Hangzhou Dianzi University, Hangzhou 310018, China; E-Mails: shenjia_hdu@163.com (J.S.); yingna@hdu.edu.cn (N.Y.)

² College of Automation Engineering, Hangzhou Dianzi University, Hangzhou 310018, China; E-Mail: sunyao@hdu.edu.cn

* Author to whom correspondence should be addressed; E-Mail: gcs@hdu.edu.cn; Tel./Fax: +86-571-8691-9123.

Academic Editor: Leonhard M. Reindl

Received: 29 July 2015 / Accepted: 28 August 2015 / Published: 3 September 2015

Abstract: Time synchronization is essential for node localization, target tracking, data fusion, and various other Wireless Sensor Network (WSN) applications. To improve the estimation accuracy of continuous clock offset and skew of mobile nodes in WSNs, we propose a novel time synchronization algorithm, the Rao-Blackwellised (RB) particle filter time synchronization algorithm based on the Dirichlet process mixture (DPM) model. In a state-space equation with a linear substructure, state variables are divided into linear and non-linear variables by the RB particle filter algorithm. These two variables can be estimated using Kalman filter and particle filter, respectively, which improves the computational efficiency more so than if only the particle filter was used. In addition, the DPM model is used to describe the distribution of non-deterministic delays and to automatically adjust the number of Gaussian mixture model components based on the observational data. This improves the estimation accuracy of clock offset and skew, which allows achieving the time synchronization. The time synchronization performance of this algorithm is also validated by computer simulations and experimental measurements. The results show that the proposed algorithm has a higher time synchronization precision than traditional time synchronization algorithms.

Keywords: wireless sensor networks; time synchronization; dirichlet process mixture model; rao-blackwellised particle filter

1. Introduction

Wireless Sensor Networks (WSNs) have broad applications in environmental monitoring, military reconnaissance, health care, and other fields [1–3]. Time synchronization is an important step in WSNs, which makes all nodes to coordinate operations in complex sensing tasks [4]. In order to deal with the effects of crystal oscillator frequency deviation and the relative motion between nodes, a random delay modeling and synchronous tracking technology are introduced to improve the accuracy of time synchronization in WSNs, which will benefit the subsequent applications, such as the node location and tracking applications [5–7].

Traditional synchronization technologies, such as Network Time Protocol (NTP) or Global Positioning System (GPS) [8,9], is not suitable for WSNs due to their size, cost, and impact on energy consumption. Therefore, many synchronization protocols for WSNs, such as Reference Broadcast Synchronization protocol (RBS) [10], Flooding Time Synchronization Protocol (FTSP) [11], Timing-Sync Protocol for Sensor Networks (TPSN) [12], and Delay Measurement Time Synchronization protocol (DMTS) [13], have been proposed in recent years. Among these protocols, TPSN shows a higher synchronization accuracy and better scalability as the delay is estimated using a two-way information exchange technique. However, non-deterministic delays cause difficulties in continuous clock offset and skew estimation during two-way timing message exchanges, which increase the computational burden of the time synchronization algorithm.

Currently, there are two main methods to solve non-deterministic delays in TPSN—selecting the appropriate delay distribution and calculating the adaptive delay distribution. In the first method, the symmetric exponential Maximum Likelihood (SEML) algorithm or the symmetric Gaussian Maximum Likelihood (SGML) algorithm [14–16] estimates clock offset by using Maximum Likelihood Estimation (MLE) in the hypothesis of symmetric exponential or Gaussian delay distributions, respectively. Chaudhari, Serpedin, *et al.* [17] introduced two unbiased estimation methods of clock offset: Best Linear Unbiased Estimation using Order Statistics (BLUS-OS) and Minimum Variance Unbiased Estimation (MVUE). Subsequently, they proved that the unbiased estimation of clock offset also coincides with MLE in the hypothesis of symmetric exponential delays. Obviously, when the application conditions are constantly changing, the clock offset and skew estimation in the determining delay distribution will lack the adaptability. In the second method, the Gaussian Mixture Model (GMM) is often used to estimate the unknown delay distribution; therefore, the estimation results of clock offset can adaptively adjust according to the change in the application conditions. Kim Lee, *et al.* [18,19] proposed estimation algorithms of clock offset based on Gaussian Mixture Kalman Particle Filter (GMKPF), and [20] introduced Iterative Gaussian Mixture Kalman Particle Filter (IGMKPF) in the estimation of clock offset, which improves the robustness of estimation results in arbitrary delay distributions. However, the predetermined number of Gaussian models weakens the flexibility of the delay distribution estimation, and affects the accuracy of time synchronization.

To solve the lack of the adaptive regulation method in modeling the delay distribution based on GMM, we conducted work on two fronts. First, we improve estimation accuracy of delay distributions based on the non-parametric Bayesian model. Next, we reduce the amount of computation in tracking non-Gaussian clock offset and skew based on the Rao-Blackwellised particle filter (RBPF). In order to overcome the limitations of the Gaussian mixture model, the Dirichlet process mixture (DPM) model is used to estimate the delay distribution, which can adjust the number of Gaussian models automatically based on the observational data. Subsequently, the DPM-based Rao-Blackwellised particle filter (DPM-RBPF) algorithm is proposed [21,22], where the system model is divided into linear and non-linear parts—it uses the particle filter and the Kalman filter to estimate the non-linear and the linear parts, respectively. Hence, the DPM-RBPF algorithm can efficiently track clock offset and skew in the non-Gaussian dynamic model. Time synchronization is achieved by improving the efficiency and accuracy of the continuous clock offset and skew estimation process.

The rest of this paper is organized as follows. In Section 2, the DPM-RBPF algorithm is stated in detail—it includes the two-way timing message exchange model, the DPM model description of the observation noise, and the RBPF algorithm. In Section 3, we compare the proposed algorithm with the other algorithms, and verify time synchronization performance of our algorithm. In Section 4, we draw conclusions and look ahead to the development direction of this research.

2. RBPF Time Synchronization Algorithm Based on the DPM Model

The time synchronization is the continuous clock offset and skew estimation, but it is crucial to deal with non-deterministic delay and packet loss. Therefore, this paper proposes a DPM-RBPF algorithm, which introduces a DPM model description of the observation noise and RBPF algorithm based on two-way timing message exchange. Consequently, the clock offset and skew tracking is achieved with the improved computational efficiency and accuracy.

2.1. State-Space Equation of Two-Way Timing Message Exchange

The two-way timing message exchange method is commonly employed in time synchronization. The specific exchange process of this method is illustrated in Figure 1.

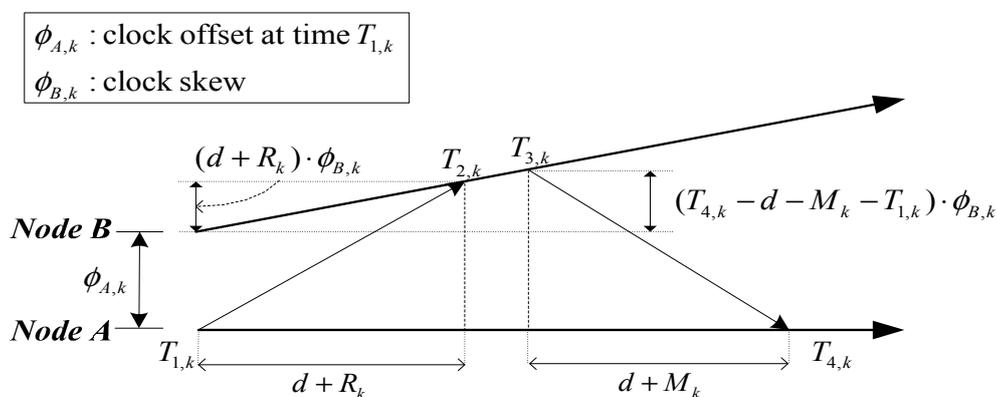


Figure 1. Two-way timing message exchange.

In Figure 1, the k th time set of message departure and arrival $\{T_{1,k}, T_{2,k}, T_{3,k}, T_{4,k}\}$ can be obtained by two-way timing message exchange. $\phi_{A,k}$ represents clock offset of Node A's clock relative to Node B's clock at time $T_{1,k}$ in the k th exchange process. $\phi_{B,k}$ represents clock skew, $\phi_{B,k} = 0$ implies that Node A and Node B have the same clock frequency. d is the fixed transmission delay between Node A and Node B. R_k and M_k are the non-deterministic network delays from Node A to Node B and from Node B to Node A, respectively.

Given the above exchange model, the time difference between the uplink and the downlink transmission can be represented as follows:

$$\begin{aligned} U_k &= T_{2,k} - T_{1,k} = (d + R_k) \cdot \phi_{B,k} + \phi_{A,k} \\ V_k &= T_{4,k} - T_{3,k} = (d + M_k) \cdot \phi_{B,k} - (\phi_{B,k} - 1)(T_{4,k} - T_{1,k}) - \phi_{A,k} \end{aligned} \quad (1)$$

The observation model can be derived from Equation (1) as follows:

$$z_k = U_k - V_k = 2\phi_{A,k} + (T_{4,k} - T_{1,k}) \cdot \phi_{B,k} + (R_k - M_k) \cdot \phi_{B,k} - (T_{4,k} - T_{1,k}) \quad (2)$$

If the clock skew cannot be ignored (*i.e.*, $\phi_{B,k} \neq 0$), then the clock offset continue to change with time, as shown in Figure 1. Therefore, the clock offset and skew should be continuously estimated to improve the accuracy and reliability of synchronization. Here, we adopt the constant clock skew model as follows.

$$\phi_{A,k} = \Delta t \cdot \phi_{B,k-1} + \phi_{A,k-1} \quad (3)$$

where Δt is the time interval of two-way timing message exchange, *i.e.*, $\Delta t = T_{1,k} - T_{1,k-1}$.

By considering the clock offset $\phi_{A,k}$ and clock skew $\phi_{B,k}$ as a state variable $x_k = [\phi_{A,k} \ \phi_{B,k}]^T$, the state-space equation of two-way timing message exchange model is described as follows:

$$\begin{aligned} x_k &= \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \cdot x_{k-1} + v_{k-1} \\ z_k &= \begin{bmatrix} 2 & T_{4,k} - T_{1,k} \end{bmatrix} \cdot x_k + n_k \end{aligned} \quad (4)$$

where v_{k-1} is the Gaussian state noise with zero mean and covariance Q , which reflects the effects of environmental temperature, crystal ageing, power supply voltage fluctuation, *etc.* $n_k = (R_k - M_k) \cdot \phi_{B,k} - (T_{4,k} - T_{1,k})$ is the unknown distribution observation noise.

If the clock skew can be ignored (*i.e.*, $\phi_{B,k} = 0$), the initial clock offset is the only parameter to be estimated. Assuming the clock offset as the state variables, the state-space equation of the two-way timing message exchange can be described as follows [18]:

$$\begin{aligned} x_k &= x_{k-1} + v_{k-1} \\ z_k &= 2x_k + n_k \end{aligned} \quad (5)$$

where $n_k = T_{1,k} - T_{4,k}$ is the observation noise.

2.2. Observation Noise DPM Model

For the unknown observation noise, the DPM model [23] is used to estimate its probability density function. The DPM model, as a non-parametric Bayesian model, has been widely used in the

probability density estimation, data clustering [24], which can flexibly adjust model parameters and the component number. The graphical model of DPM model is shown in Figure 2.

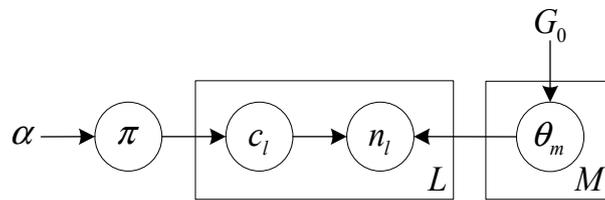


Figure 2. Dirichlet process mixture model.

In Figure 2, the observation noise n_l obeys the likelihood function F with the indicator c_l and the parameter set $\theta = \{\theta_1, \dots, \theta_m, \dots, \theta_M\}$, where θ_m is the parameter of the m th component posterior based on the prior G_0 with the parameter θ_0 . The parameter indicator c_l is obtained from a multinomial distribution π with a symmetric Dirichlet prior. Therefore, the formula of the hierarchical structure is expressed as follows:

$$n_l \sim F(n_l | \theta_{c_l}); \theta_{c_l=m} \sim G_m; c_l \sim Mul(\pi); \pi \sim Dir(\alpha / M, \dots, \alpha / M) \tag{6}$$

where α is the concentration parameter of the Dirichlet distribution. K is the dimensional number of Dirichlet distribution. M is the number of the posterior component.

In this work, the likelihood function $F(\cdot)$ is selected to be the Gaussian distribution, and the conjugate prior G_0 is described as the product of a Gaussian distribution and an inverse chi-squared distribution: $G_0(\mu, \sigma^2 | \theta_0) = \mathcal{N}(\mu | \mu_0, \sigma^2 / \lambda_0) \cdot \mathcal{X}^{-1}(\sigma^2 | \sigma_0^2, \nu_0)$, where $\theta_0 = \{\mu_0, \lambda_0, \sigma_0^2, \nu_0\}$. μ_0 and λ_0 are the mean and the scale parameter of the Gaussian distribution \mathcal{N} , respectively. σ_0^2 and ν_0 are the variance and the degree of freedom of the inverse chi-squared distribution \mathcal{X}^{-1} , respectively. Therefore, the posterior distribution is also a normal inverse chi-squared distribution. If the observation noise set $\mathbf{n} = \{n_1^m, n_2^m, \dots, n_{L_m}^m\}$ is received, where all L_m observation noise have the indicator $c_l = m$, the m th component posterior is expressed as following:

$$p(\mu, \sigma^2 | \theta_m) = \mathcal{N}(\mu | \mu_m, \sigma^2 / \lambda_m) \cdot \mathcal{X}^{-1}(\sigma^2 | \sigma_m^2, \nu_m)$$

where

$$\mu_m = (\lambda_0 \mu_0 + \sum_{i=1}^{L_m} n_i^m) / (\lambda_0 + L_m),$$

$$\lambda_m = \lambda_0 + L_m, \quad \nu_m = \nu_0 + L_m, \tag{7}$$

$$\sigma_m^2 = \sigma_0^2 + \sum_{i=1}^{L_m} (n_i^m)^2 + (\lambda_0 \mu_0 / \sqrt{\lambda_0})^2 - \left(\left(\lambda_0 \mu_0 + \sum_{i=1}^{L_m} n_i^m \right) / \sqrt{\lambda_0 + L_m} \right)^2$$

If the observation noise n_l is randomly selected, the indicator c_l and the parameter of the component posterior $\theta_m = \{\mu_m, \lambda_m, \sigma_m^2, \nu_m\}$ can be updated. Following the Chinese restaurant scheme of the Dirichlet process [21], the m th component posterior is formulated as follows:

$$p(c_l = m | \mathbf{c}_{-l}, n_k, \alpha, \theta_m) \propto p(c_l = m | \mathbf{c}_{-l}, \alpha) p(n_k | c_l = m, \theta_m)$$

$$p(c_l = m | \mathbf{c}_{-l}, \alpha) = \begin{cases} L_{-l,m} / (L + \alpha - 1) & \text{if } m \text{ has been appeared} \\ \alpha / (L + \alpha - 1) & \text{if } m \text{ is new} \end{cases} \tag{8}$$

where the indicator set \mathbf{c}_{-l} is defined as $\{c_1, \dots, c_{l-1}, c_{l+1}, \dots, c_L\}$. L denotes the number of indicators. $L_{-l,m}$ is the number of the indicator, which is equal to m in the indicator set \mathbf{c}_{-l} .

The likelihood function $p(n_k | c_l = m, \theta_m)$ is calculated as follows:

$$p(n_k | c_l = m, \theta_m) = \int \int_{\mu, \sigma^2} p(n_k | \mu, \sigma^2) p(\mu, \sigma^2 | \theta_m) d\mu d\sigma^2$$

$$= \pi^{-1/2} \left(\frac{\lambda_m + 1}{\lambda_m} \right)^{-1/2} \left(\frac{\sigma_m^2 + n_k^2 + \frac{(\lambda_m + 1)S_m^2 - \lambda_m(S_m + n_k)^2}{\lambda_m(\lambda_m + 1)}}{\sigma_m^2} \right)^{-(v_m + 1)/2} \frac{\Gamma\left(\frac{v_m + 1}{2}\right)}{\Gamma\left(\frac{v_m}{2}\right)} \quad (9)$$

where $S_m = \lambda_0 \mu_0 + \text{sum}_m$, sum_m represents the sum of observation noise, whose indicator is equal to m .

After the iterative update of the indicator and the parameter of the component posterior, the concentration parameter α is also sampled from the posterior distribution. Assume $\alpha \sim \text{Gamma}(a, b)$, a gamma prior with shape $a > 0$ and scale $b > 0$. In this paper, we adopt the auxiliary variable method for sampling α . By defining auxiliary variables w and s , where w is a variable with continuous values in $[0, 1]$, and s is a binary $\{0, 1\}$ variable, the component posterior $p(\alpha | w, s, m)$ is expressed as [25]:

$$p(\alpha | w, s, m) \propto \alpha^{a-1+m-s} e^{-\alpha(b-\log w)} \quad (10)$$

which is a gamma distribution with parameters $a + m - s$ and $b - \log w$. m ($m > 1$) is the number of normal components in the mixture model, after given α , the w , s and m may be sampled from the following distribution is given by:

$$p(w | \alpha) \propto w^\alpha (1-w)^{L-1}$$

$$p(s | \alpha) \propto (L/\alpha)^s \quad (11)$$

$$p(m | \alpha, L) \propto s(L, m) \alpha^m \Gamma(\alpha) / \Gamma(\alpha + L)$$

where w and s obey beta and binomial distributions, respectively. The $s(L, m)$ and $\Gamma(\cdot)$ are unsigned Stirling numbers of the first kind and gamma function, respectively. The number of normal components m can be calculate based on the concentration parameter α and the number of observations L , it means that the DPM model can flexibly adjust the number of gaussian components using the observation data.

2.3. State Tracking Based on DPM-RBPF

For the non-Gaussian observation noise in the state-space equation of clock offset or skew, the DPM-RBPF algorithm is used for time synchronization. The DPM model is used to describe the observation noise. RBPF is used to track clock offset or skew. RBPF is widely used in visual tracking, simultaneous localization and mapping (SLAM), and other fields [26,27]. It divides the state variables into linear and non-linear parts based on the Rao-Blackwell statistical theory, and improves the computational efficiency of the particle filter by integrating the Kalman filter.

As shown in Equation (4), by exchanging the times information between Node A and B for L times, we get the entire observation data $z_k = T_{2,k} + T_{3,k} - T_{1,k} - T_{4,k}$. The clock offset and skew defined as the state variables x_k , and the indicator of the model component c_k is defined as the hidden variable.

According to Bayes' theorem, the posterior probability density function $p(x_k, c_k | \alpha, \theta, z_{1:k}, x_{1:k-1})$ for the unknown variables $[x_k, c_k]$ can be decomposed into the linear section $p(x_k | c_k, \theta, z_k, x_{k-1})$ and the non-linear section $p(c_k | \alpha, \theta, z_k, x_{k-1})$ shown as follows:

$$p(x_k, c_k | \alpha, \theta, z_{1:k}, x_{1:k-1}) = p(x_k | c_k, \theta, z_k, x_{k-1}) \cdot p(c_k | \alpha, \theta, z_k, x_{k-1}) \quad (12)$$

Thus, the estimation of the unknown variables is divided into the estimation of the linear and non-linear variables independently through the marginalization of the probability density function. The Kalman filter is used to estimate the linear variable, and the particle filter is used to calculate the non-linear variable. Since the variable dimension of $p(c_k | \alpha, \theta, z_k, x_{k-1})$ is less than the variable dimension of $p(x_k, c_k | \alpha, \theta, z_k, x_{k-1})$, the DPM-RBPF algorithm needs fewer particles than the standard particle filter, which can effectively improve the computational efficiency of this algorithm [28]. The following section describes linear and non-linear estimation, respectively.

The posterior probability density of the non-linear hidden variable c_k can be approximated by a set of random particles with certain importance weight, such that the posterior $p(c_k | \alpha, \theta, z_k, x_{k-1})$ can be approximated as follows:

$$p(c_k | \alpha, \theta, z_k, x_{k-1}) = \sum_{i=1}^N w_k^i \cdot \delta(c_k - c_k^i) \quad (13)$$

where c_k^i is the i^{th} particles with the importance weight w_k^i , $\delta(\cdot)$ is the Dirac delta function. N is the number of particles.

Initially, the particle c_k^i is sampled according to the Equation (8), and its importance weight w_k^i is calculated using the sequential Monte Carlo sampling as follows:

$$w_k^i = w_{k-1}^i \frac{p(z_k | c_k, \theta, z_{k-1}, x_{k-1}) \cdot p(c_k | \theta, z_{k-1}, x_{k-1})}{q(c_k | \theta, z_k, x_{k-1})} \quad i = 1, 2, \dots, N \quad (14)$$

In order to simplify the calculation of the importance weight w_k^i , the proposal distribution $q(c_k | \theta, z_k, x_{k-1})$ is usually chosen to be equal to the conditional posterior $p(c_k | \theta, z_{k-1}, x_{k-1})$. Hence, the importance weight can be updated by using the formula $w_k^i = w_{k-1}^i p(z_k | c_k, \theta, z_{k-1}, x_{k-1})$. In a specific implementation process, the particles need to be re-sampled to prevent the degradation of particles, according to the weighting coefficient.

After the nonlinear hidden variable c_k is sampled using the PF method, the linear state variable x_k is calculated by the Kalman filter method. For each particle c_k^i , we can recursively calculate the state variable x_k^i and the variance P_k^i by using a Kalman filter as shown below:

$$(x_k^i, P_k^i) = \text{KF}(c_k^i, \theta, x_{k-1}^i, P_{k-1}^i, z_k) \quad (15)$$

Therefore, the posterior probability density function $p(x_k | c_k, \theta, z_k, x_{k-1})$ is approximately expressed by using a mixture of densities shown as follows:

$$p(x_k | c_k, \theta, z_k, x_{k-1}) = \sum_{i=1}^N w_k^i \cdot \delta(x_k - x_k^i) \quad (16)$$

Finally, the minimum mean squared error (MMSE) estimation of linear state variable and noise covariance matrix can be written as follows:

$$\begin{aligned}\hat{x}_k &= \sum_{i=1}^N w_k^i x_k^i \\ \hat{P}_k &= \sum_{i=1}^N w_k^i \left[P_k^i + (x_k^i - \hat{x}_k)(x_k^i - \hat{x}_k)^T \right]\end{aligned}\quad (17)$$

2.4. Algorithmic Process

We propose the DPM-RBPF algorithm to improve the accuracy and efficiency of continuous clock offset and skew estimation in TPSN. It includes three key steps. First, the state-space equation is initialized and the DPM model is trained based on the approximate observation noise in Equation (5). Next, the particles of the indicator are re-sampled based on the updated weights. Furthermore, the DPM model is continuously retrained based on the new obtained observations and state values. Finally, the particles of the state variable are calculated using the Kalman filter and the state variable is estimated to complete time synchronization between two nodes. The process of DPM-RBPF algorithm is listed as shown below in Algorithm 1:

Algorithm 1. DPM-RBPF algorithm.

Step (1) Initialize the DPM model and the state-space equation.

- Set the concentration parameter α and the base distribution G_0 . According to the approximate observation noise data set, the DPM model is trained using Equations (7)–(11).
- Set the initial state variable $\hat{x}_0 = \hat{x}_{ML}$.
- Sample the N initial particles of the model component indicator from DPM model.
- Initialize the importance weight of particles uniformly.

Step (2) Resample the particles and update the DPM model.

- Calculate and normalize the importance weights using Equation (14).
- Generate a new set of particles $\{c_k^i\}_{i=1}^N$ with the corresponding importance weights $\{w_k^i\}_{i=1}^N$ using the resampling method.
- DPM model is updated based on the observations and estimated state variables in a time interval.

Step (3) Estimate the state variable.

- According to Equation (15), the state variables of all particles are updated. According to the Equation (17), the state variables are calculated.
 - If the online calculation is not complete, return to Step (2).
-

3. The Performance and the Analysis

In this section, the estimation accuracy and computational efficiency of the DPM-RBPF algorithm are compared with the Maximum Likelihood Estimation (MLE) [15,16], the Advanced Self-Correcting Time Synchronization (ASCTS) [29] and the Iterative Gaussian Mixture Kalman Particle Filter (IGMKPF) algorithm [20]. The performance of the proposed algorithm is verified via computer simulations and experimental measurements, and the influence of the number of observations, particles, and other parameters is analyzed in detail. Unless otherwise specified in the following experiments, it is assumed that the variance Q of state noise is equal to $1e^{-4}$, the number of the indicator particles is

set to 500, and the concentration parameter α is followed by gamma prior with the shape $a = 1$ and the scale $b = 1$.

3.1. Simulations' Comparison

Assuming that the non-deterministic delay model obeys Gaussian distribution and Laplace distribution, time synchronization results of MLE, ASCTS, IGMKPF and DPM-RBPF algorithm are compared. For the Gaussian distribution experiment, the parameters are set as follows: the non-deterministic delay R_k and M_k are assumed to follow the Gaussian distribution with zero mean and 0.01 variance. The initial value of the state \hat{x}_0 is set to 0.001. The average results of 30 trials of the three algorithms are shown in Figure 3a. The vertical coordinate is the mean square error (MSE), which is used to evaluate the performance of the clock offset and skew estimation. The horizontal coordinate is the number of two-way timing message exchanges. Each experiment was performed 30 times two-way timing message exchange to track the clock offset and skew. In the experiment of Laplace distribution, the parameters are set as follows: R_k and M_k are assumed to follow the Laplace distribution with zero location parameter and 0.01 scale parameter. The other settings are identical to the experiment of Gaussian distribution. The average results are shown in Figure 3b.

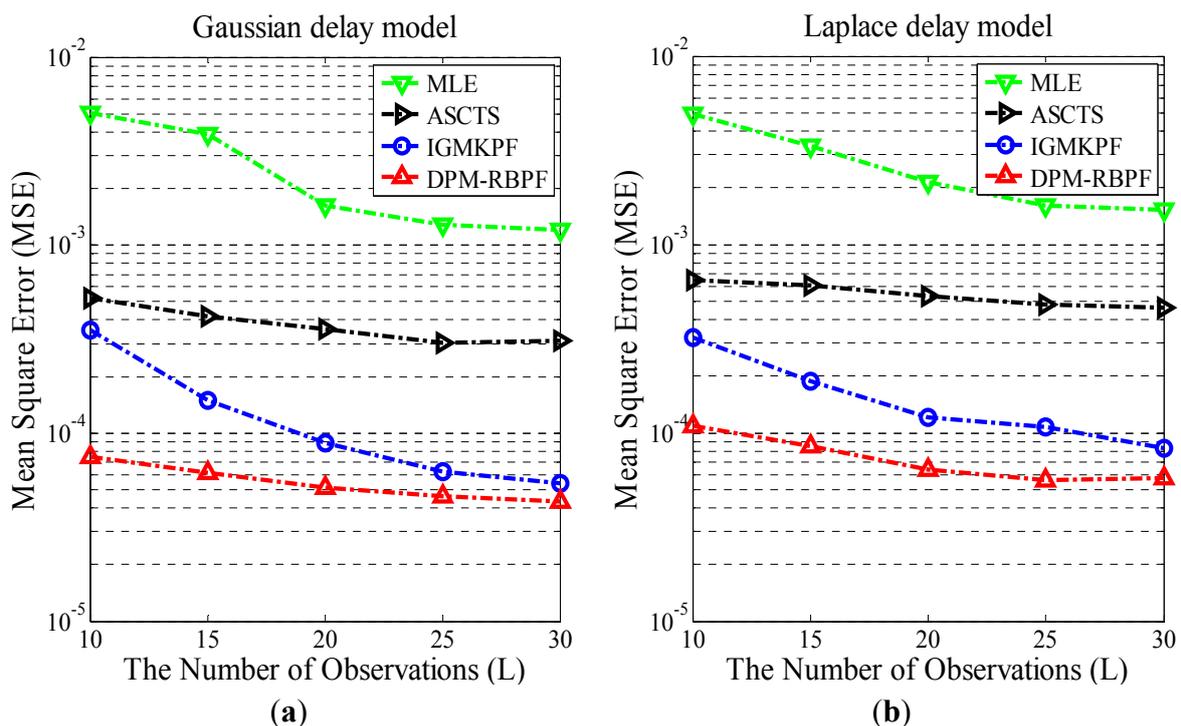


Figure 3. (a) MSE of clock offset estimators for symmetrical Gaussian delay; (b) MSE of clock offset estimators for symmetrical exponential delay.

As shown in Figure 3, with an increase in the number of observations, there is a gradual reduction in MSE. A smaller MSE implies that there is a higher precision of time synchronization. Comparing the four algorithms—MLE, ASCTS, IGMKPF and DPM-RBPF—DPM-RBPF is better than MLE, ASCTS and IGMKPF in MSE of the clock offset estimation. This indicates that the DPM model describes the observation noise more accurately. The hypothetical observation noise distribution model

used by MLE is probably different from the actual observation noise distribution model. Therefore, the performance of MLE is worse than the other two algorithms, which implement an adaptive observation noise model.

Gaussian distribution and Laplace distribution are symmetrical, but the delay noise is often asymmetric. As a result, the performances of the algorithms are compared under conditions of asymmetric delay noise, and the computational efficiency of DPM-RBPF and IGMKPF are tested. Assuming the non-deterministic delay model obeys an asymmetric distribution, R_k is the Gaussian distribution with zero mean and 0.01 variance, and M_k is the exponential distribution with 0.01 rate parameter. Figure 4a shows the comparison of the estimation results of different algorithms. Since the hypothetical noise distribution is inconsistent with the actual noise distribution, the performance of MLE under conditions of symmetrical Gaussian distribution or symmetrical exponential distribution (Laplace distribution) becomes worse. The performance of DPM-RBPF remains superior to the four comparative algorithms. Figure 4b shows the number of particles needed by DPM-RBPF, DPM-PF and IGMKPF under different MSE. For the same MSE, DPM-RBPF requires fewer particles than DPM-PF and IGMKPF. The calculation extent of DPM-RBPF is less than the other algorithms to achieve the same accuracy. This means that the RBPF can improve the computational efficiency.

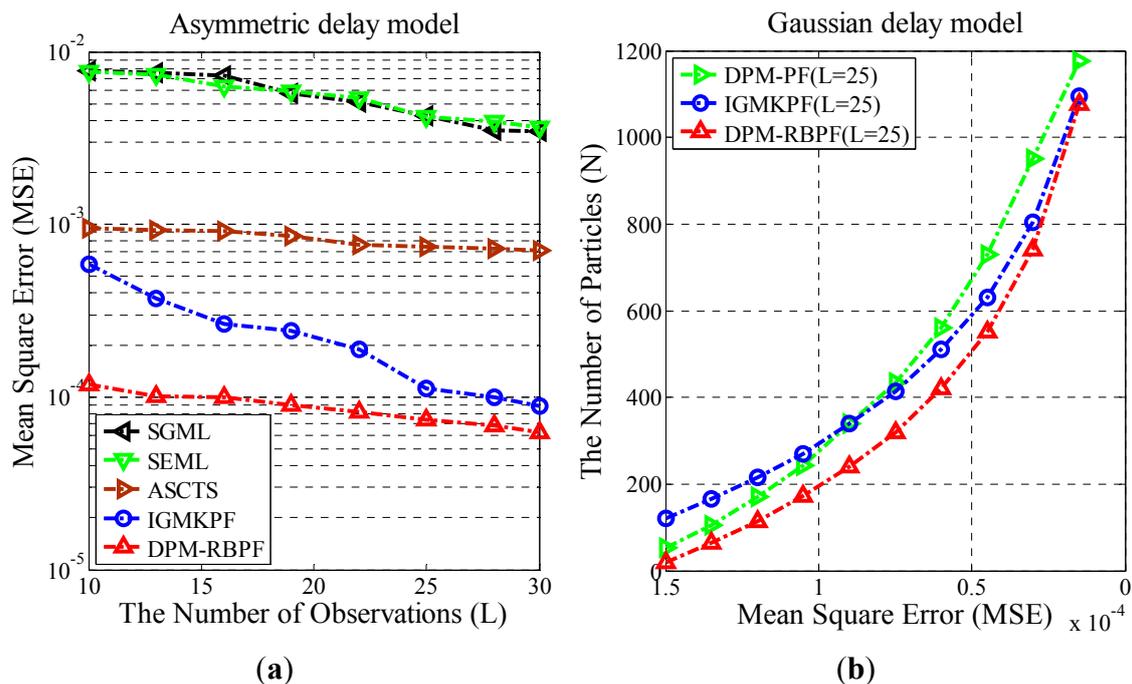


Figure 4. (a) MSE of clock offset estimators for mixing of a Gaussian and an exponential; (b) The relationship between the number of particles and MSE.

3.2. Parameters' Analysis

The impact of the relevant parameters on the DPM-RBPF algorithm is discussed in this section. Since sensor nodes are energy-constrained, the number of the time message exchanges between the nodes is limited, and the number of particles for the RBPF algorithm is also limited. In order to minimize energy consumption for a given synchronization accuracy, we evaluate the impact of the number of observations and the number of particles on the DPM-RBPF algorithm.

3.2.1. Relationship Analysis between the Number of Observations and MSE

Figure 5a shows the relationship between the number of observations L and the MSE of the clock offset when the numbers of particles are 100, 200, 300, 400, and 500, respectively. As can be seen from the figure, with fewer particles ($N = 100$ or $N = 200$), MSE only changes slightly with an increase in the number of observations. In other words, when the number of particles is too small, the DPM-RBPF algorithm cannot fully exploit the time information of observations. However, when the number of particles is greater than 300, we find that the MSE becomes smaller with an increase in the number observations. In order to guarantee that MSE is less than $7e^{-5}$, and to minimize energy consumption of nodes, we select $L = 20$ and $N \geq 400$.

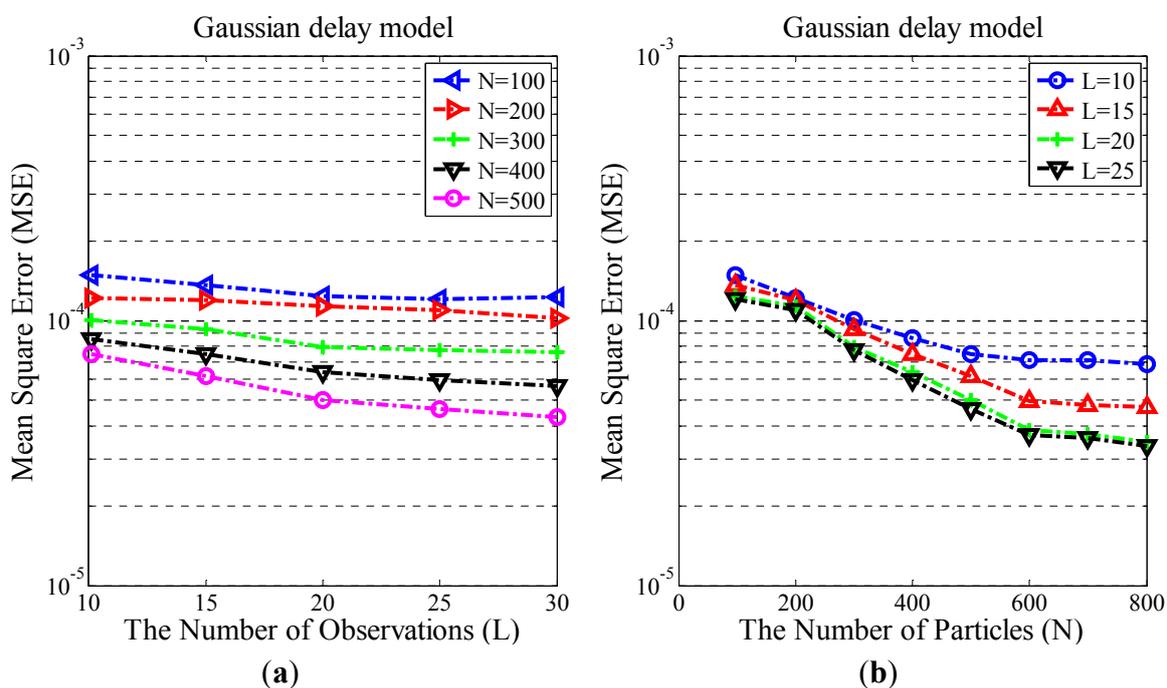


Figure 5. (a) The relationship between the Number of Observations and MSE; (b) The relationship between the Number of Particles and MSE.

3.2.2. Relationship Analysis between the Number of Particles and MSE

Figure 5b shows the relationship of the number of RB particle filter and the MSE when the number of observations is 10, 15, 20, and 25, respectively. From the figure, we can see that when the number of observations is constant, MSE increases as the number of particles reduces, and vice versa. The two curves $L = 20$ and $L = 25$ substantially coincide and MSE is similar. When the number of particles is larger than 600, MSE stabilizes. Considering synchronization accuracy and energy consumption comprehensively, we finally select $L = 20$ and $N = 500$.

3.3. Experimental Measurement

To evaluate the time synchronization performance, we set up a testing system with four mobile nodes based on Arduino Uno boards, as shown in Figure 6. Arduino Uno is a microcontroller board based on the ATmega328 processor with 16 MHz crystal clock oscillator. The on-chip memory

includes 32 KB of flash memory, 1 KB of EEPROM and 2 KB of SRAM. All Arduino Uno configured with CC3000 WiFi Shield establish the communication links to each other through a wireless router. The program of the proposed DPM-RBPF algorithm is running on the control computer with a 2.27 GHz CPU and 16 GB RAM.

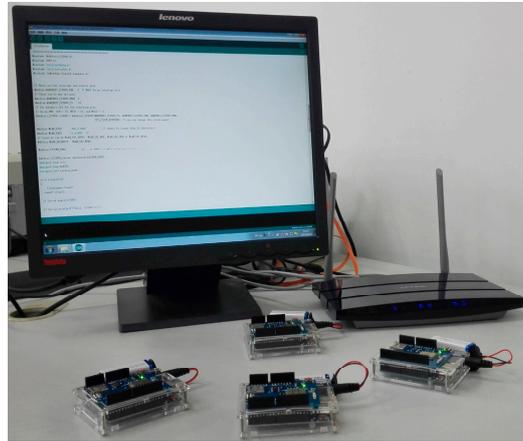


Figure 6. The experimental platform with four nodes (Arduino Uno board).

According to the two-way timing message exchange method, the control computer periodically sends the timestamps packet to four mobile nodes. When the mobile node receives a timestamps packet, it would read the clock at the time of receiving and sending the timestamps packet. These clocks are calibrated based on the clock offset in the timestamps packet, and fill back the timestamps packet. Then, the timestamps packet was immediately sent back to the control computer for the next clock skew tracking. The format of timestamps packet is $\{T_{0,k}, T_{1,k}, T_{2,k}, T_{3,k}, T_{4,k}\}$, which includes the clock offset $T_{0,k}$, the computer clock $T_{1,k}$ and $T_{4,k}$ at time of receiving and sending the timestamps packet, and the node clock $T_{2,k}$ and $T_{3,k}$ at the time of receiving and sending timestamps packet, where k is the number of two-way timing message exchanges. If there is no clock offset calibration (It means the $T_{0,k}$ in timestamp packet is set to 0), the clock in timestamps packet from k_0 to $k_0 + 3$ times is shown in the left section of Table 1. Due to the existence of the clock offset and skew between the control computer and the mobile node, $T_{1,k}$, $T_{4,k}$ and $T_{2,k}$, $T_{3,k}$ have a significant difference. If there is a clock offset calibration, as shown in the right section of Table 1, the clock values from $T_{1,k}$ to $T_{4,k}$ continue to increase, which is consistent with the chronological laws of the timestamps packet exchange.

Table 1. The comparison between no calibration and calibration (time interval 2 s).

| k | No calibration | | | | | Calibration (μ s) | | | | |
|-----------|----------------|-------------|-------------|-------------|-------------|------------------------|-------------|-------------|-------------|-------------|
| | $T_{0,k}$ | $T_{1,k}$ | $T_{2,k}$ | $T_{3,k}$ | $T_{4,k}$ | $T_{0,k}$ | $T_{1,k}$ | $T_{2,k}$ | $T_{3,k}$ | $T_{4,k}$ |
| k_0 | 0 | 118,104,732 | 100,814,673 | 100,816,003 | 118,225,238 | 277,031 | 118,104,732 | 118,161,017 | 118,162,347 | 118,225,238 |
| $k_0 + 1$ | 0 | 120,234,711 | 102,616,610 | 102,617,649 | 120,306,343 | 275,900 | 120,234,711 | 120,238,854 | 120,239,893 | 120,306,343 |
| $k_0 + 2$ | 0 | 122,324,748 | 104,408,959 | 104,410,527 | 122,395,988 | 307,936 | 122,324,748 | 122,339,139 | 122,340,707 | 122,395,988 |
| $k_0 + 3$ | 0 | 124,414,626 | 106,189,262 | 106,190,567 | 124,564,677 | 314,981 | 124,414,626 | 124,434,423 | 124,435,728 | 124,564,677 |

Using the build test system, the clock synchronization performances of ASCTS and DPM-RBPF algorithm were compared. The tracking curves of the clock offset and skew of four mobile nodes are shown in Figure 7. There is a certain clock offset and skew to every mobile node at the start time.

Through tracking the clock offset and skew, the clock offset gradually moves towards zero, which means the clock synchronization can be achieved. Compared with ASCTS, DPM-RBPF can quickly complete the clock synchronization, which calibrates the clock offset at 20, 22, 25, 39 steps, respectively, to four mobile nodes. The tracking curves of DPM-RBPF algorithm show a small fluctuation. This means that the proposed method has better tracking performance.

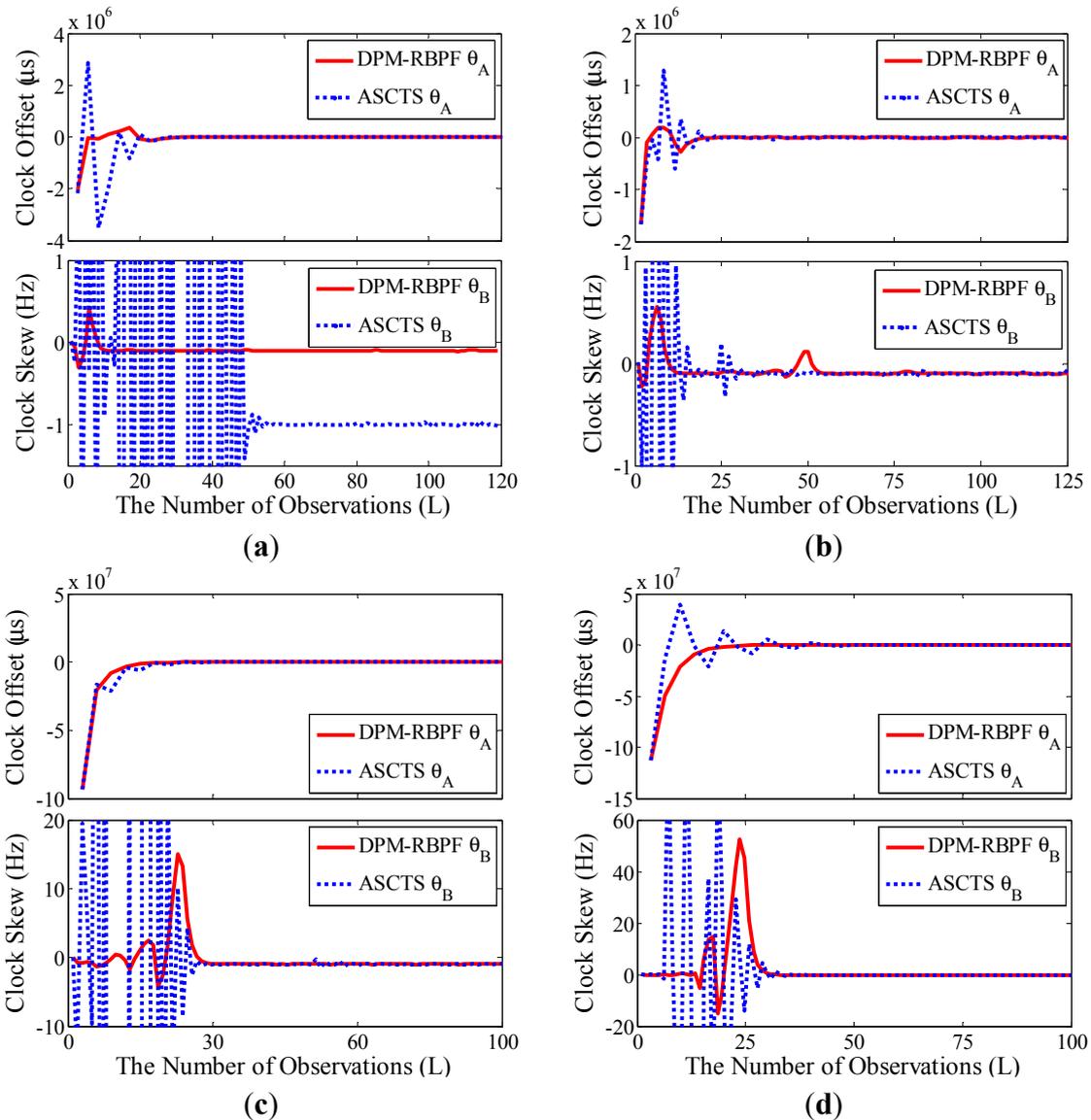


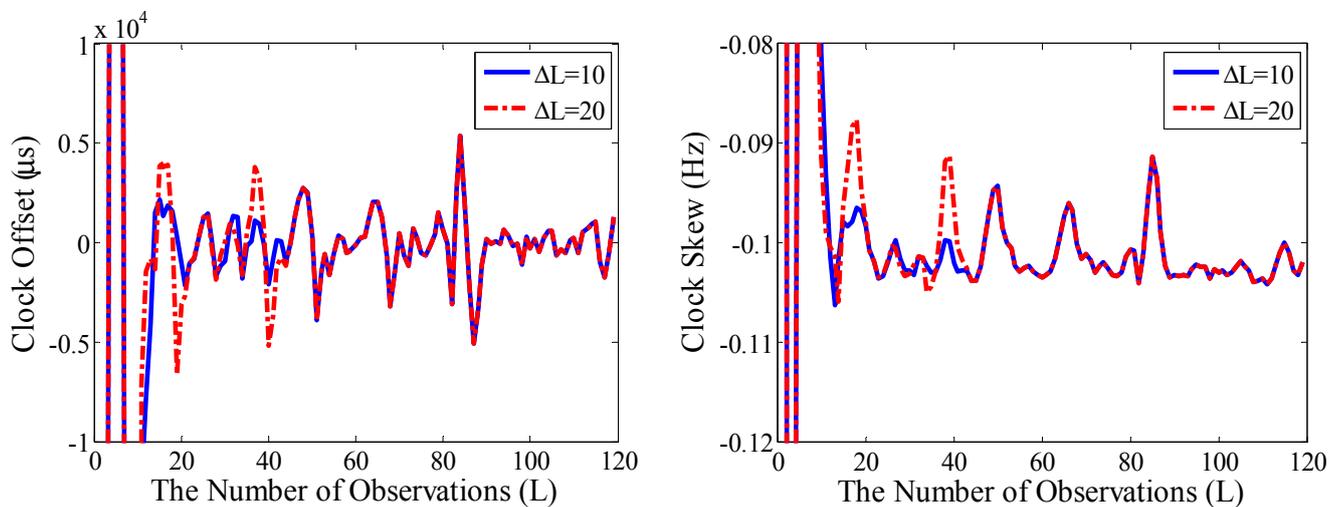
Figure 7. Tracking curves of ASCTS and DPM-RBPF (time interval 5 s). (a) Node A; (b) Node B; (c) Node C; (d) Node D.

Quantitative analysis of time synchronization is shown in Table 2. The mean and standard deviation of clock offset and skew in the steady state are listed. Since the observation noise is randomly distributed with nonzero mean, the estimation of clock offset in steady state has certain bias. This bias can usually be counteracted by system calibration. The standard deviation (STD) in Table 2 shows less tracking fluctuation in DPM-RBPF than ASCTS, which means that the time synchronization performance of DPM-RBPF is superior to ASCTS's.

Table 2. Quantitative analysis of time synchronization in steady state (time interval 5 s).

| | Node A | | Node B | | Node C | | Node D | |
|----------------------------------|--------|-------|--------|-------|--------|-------|--------|-------|
| | MEAN | STD | MEAN | STD | MEAN | STD | MEAN | STD |
| ASCTS $\theta_A(\mu\text{s})$ | -53.9 | 408.4 | -33.5 | 313.7 | -25.4 | 552.3 | 18.3 | 319.5 |
| ASCTS $\theta_B(\text{Hz})$ | -0.99 | 0.004 | -0.101 | 0.003 | -0.99 | 0.004 | -0.99 | 0.003 |
| DPM-RBPF $\theta_A(\mu\text{s})$ | 16.1 | 288.8 | 160.2 | 211.5 | 151.3 | 379.3 | 36.4 | 260.5 |
| DPM-RBPF $\theta_B(\text{Hz})$ | -0.10 | 0.002 | -0.099 | 0.002 | -0.99 | 0.001 | -0.99 | 0.002 |

In order to reveal the effect of the DPM model updated in the experiment, different observation intervals ΔL are selected. It means the DPM model is updated after obtaining a different number of observation noises, which are calculated based on the observations and the estimated state variables. When two different time intervals 10 and 20 are selected, the tracking curves of the clock offset and skew on node A are shown in Figure 8.

**Figure 8.** Tracking curves for different intervals (time interval 5 s).

As shown in Figure 8, since the update of the observation noise model is inconsistent, the tracking curves appear different within the 10–40 observation point range. Despite ignoring the clock skew in the initialization of the observation noise model, the tracking curves soon become identical after more than 40 points. It means that the updating DPM model can better adapt to environmental change and the DPM model is able to describe the unknown observation noise, which ensures the accurate tracking of clock offset and skew.

4. Conclusions

In this paper, we propose a novel time synchronization algorithm, DPM-RBPF, to improve the accuracy and efficiency of continuous clock offset and skew estimation in a dynamic sensor network. In the proposed algorithm, a unified state-space equation is constructed to track the clock offset and skew, and a DPM model is used to describe the distribution of observation noise, which can flexibly adjust the number of Gaussian components to enhance the adaptability of the observation noise model. In addition, simulation experiments indicate the advantages of the proposed algorithm in terms of

estimation precision and computational efficiency compared with the other algorithms—such as SGML, SEML, ASCTS and IGMKPF algorithms. We also discuss the parameters' selection process of the proposed algorithm. The results of computer simulations and test systems demonstrate that the proposed algorithm outperforms other algorithms in time synchronization accuracy.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 61372157), Zhejiang Provincial Natural Science Foundation of China (No. LY12F01003) and Zhejiang Province Public Service Technology Research Projects (No. 2014C31056).

Author Contributions

All authors have made significantly contributions to this manuscript. ChunSheng Guo, Jia Shen and Yao Sun carried out a literature review. ChunSheng Guo, Jia Shen and Na Ying proposed the algorithm, designed the experiments, and wrote the manuscript. ChunSheng Guo and Jia Shen performed the data analysis. All authors also contributed to modification of the manuscript.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Chong, C.Y.; Kumar, S.P. Sensor networks: Evolution, opportunities, and challenges. *IEEE Proceed.* **2003**, *8*, 1247–1256.
2. Jafari, R.; Encarnacao, A.; Zahoory, A.; Dabiri, F.; Noshadi, H.; Sarrafzadeh, M. Wireless sensor networks for health monitoring. In Proceedings of the 2th Annual International Conference on Mobile and Ubiquitous Systems, San Diego, CA, USA, 1 January 2005; pp. 479–481.
3. Elson, J.; Romer, K. Wireless sensor networks: A new regime for time synchronization. In Proceedings of the First Workshop on Hot Topics In Networks, Princeton, NJ, USA, 28–29 October 2002; pp. 149–154.
4. Sivrikaya, F.; Yener, B. Time synchronization in sensor networks: A survey. *IEEE Netw.* **2004**, *4*, 45–50.
5. Chen, H.Y.; Gao, F.F.; Martins, M.; Huang, P.; Liang J.L. Accurate and Efficient Node Localization for Mobile Sensor Networks. *Mobile Netw. Appl.* **2013**, *1*, 141–147.
6. Chen, H.Y.; Liu, B.; Huang, P.; Liang J.L.; Gu, Y. Mobility-Assisted Node Localization Based on TOA Measurements Without Time Synchronization in Wireless Sensor Networks. *Mobile Netw. Appl.* **2012**, *1*, 90–99.
7. Liu, B.; Chen, H.Y.; Zhong, Z.G.; Poor, H.V. Asymmetrical Round Trip Based Synchronization-Free Localization in Large-Scale Underwater Sensor Networks. *IEEE Trans. Wirel. Commun.* **2010**, *11*, 3532–3542.
8. Cho, H.; Kim, J.; Baek, Y. Enhanced Precision Time Synchronization for Wireless Sensor Networks. *Sensors* **2011**, *11*, 7625–7643.

9. Bulusu, N.; Jha, S. *Wireless Sensor Networks: A Systems Perspective*, Artech House: Norwood, MA, USA, 2005.
10. Elson, J.; Girod, L.; Estrin, D. Fine-Grained Network Time Synchronization using Reference Broadcasts. In Proceedings of the Fifth Symposium on Operating Systems Design and Implementation (OSDI2002), Boston, MA, USA, 9–11 December 2002; pp. 147–163.
11. Maroti, M.; Kusy, B.; Simon, G.; Ledeczi, A. The flooding time synchronization protocol. In Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, Baltimore, MA, USA, 3–5 November 2004; pp. 39–49.
12. Ganeriwal, S.; Kumar, R.; Srivastava, M.B. Timing-Sync Protocol for Sensor Networks. In Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, Los Angeles, CA, USA, 5–7 November 2003; pp. 138–149.
13. Ping, S. *Delay Measurement Time Synchronization for Wireless Sensor Networks*; Technical Report for Intel Research Berkeley Lab, Pittsburgh, PA, USA, June 2003; pp. 1–10.
14. Jeske, D.R. On maximum-likelihood estimation of clock offset. *IEEE Trans. Commun.* **2005**, *1*, 53–54.
15. Noh, K.L.; Chaudhari, Q.M.; Serpedin, E.; Suter, B.W. Novel Clock Phase Offset and Skew Estimation Using Two-Way Timing Message Exchanges for Wireless Sensor Networks. *IEEE Trans. Commun.* **2007**, *4*, 766–777.
16. Chaudhari, Q.M.; Serpedin, E.; Qaraqe, K. On Maximum Likelihood Estimation of Clock Offset and Skew in Networks with Exponential Delays. *IEEE Trans. Signal Process.* **2008**, *4*, 1685–1697.
17. Chaudhari, Q.M.; Serpedin, E.; Qaraqe, K. On Minimum Variance Unbiased Estimation of Clock Offset in a Two-Way Message Exchange Mechanism. *IEEE Trans. Inf. Theory* **2010**, *6*, 2893–2904.
18. Kim, J.S.; Lee, J.; Serpedin, E.; Qaraqe, K. A robust estimation scheme for clock phase offsets in wireless sensor networks in the presence of non-Gaussian random delays. *Signal Process.* **2009**, *6*, 1155–1161.
19. Kim, J.S.; Lee, J.; Serpedin, E.; Qaraqe, K. A Robust Approach for Clock Offsets Estimation in Wireless Sensor Networks. *EURASIP J. Adv. Sig. Proc.* **2010**, *2010*, doi:10.1155/2010/132381.
20. Kim, J.S.; Lee, J.; Serpedin, E.; Qaraqe, K. Robust Clock Synchronization in Wireless Sensor Networks through Noise Density Estimation. *IEEE Trans. Signal Process.* **2011**, *7*, 3035–3047.
21. Teh, Y.W.; Jordan, M.I.; Beal, M.J.; Blei, D.M. Hierarchical Dirichlet Processes. *J. Am. Statist. Assoc.* **2004**, *1*, 6–10.
22. Schon, T.; Gustafsson, F.; Nordlund, P.J. Marginalized particle filters for mixed linear/nonlinear state-space models. *IEEE Trans. Signal Process.* **2005**, *7*, 2279–2289.
23. Ota, K.; Duflos, E.; Vanheeghe, P.; Yanagida, M. Speech recognition with speech density estimation by the Dirichlet Process Mixture. In Proceedings of the IEEE International Conference on Speech and Signal Processing (ICASSP), Las Vegas, NV, USA, 31 March–4 April 2008; pp. 1553–1556.
24. Cherian, A.; Morellas, V.; Papanikolopoulos, N.; Bedros, S.J. Dirichlet process mixture models on symmetric positive definite matrices for appearance clustering in video surveillance applications. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 20–25 June 2011; pp. 3417–3424.

25. Rabaoui, A.; Viandier, N.; Marais, J.; Duflos, E. On selecting the hyperparameters of the DPM models for the density estimation of observation errors. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, Czech Republic, 22–27 May 2011; pp. 4092–4095.
26. Miller, I.; Campbell, M.; Huttenlocher, D. Efficient Unbiased Tracking of Multiple Dynamic Obstacles under Large Viewpoint Changes. *IEEE Trans. Robot.* **2011**, *1*, 29–46.
27. Maddern, W.; Milford, M.; Wyeth, G. Continuous appearance-based trajectory SLAM. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, 9–13 May 2011; pp. 3595–3600.
28. Karlsson, R.; Schon, T.; Gustafsson, F. Complexity analysis of the marginalized particle filter. *IEEE Trans. Signal Process.* **2005**, *11*, 4408–4411.
29. Liu, B.; Ren, F.Y.; Shen, J.Y.; Chen, H.Y. Advanced self-correcting time synchronization in wireless sensor networks. *IEEE Commun. Lett.* **2010**, *4*, 309–311.

© 2015 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/4.0/>).