Circuits Syst Signal Process (2013) 32:825–841
DOI 10.1007/s00034-012-9506-y

Adaptive Regularized Particle Filter for Synchronization of Chaotic Colpitts Circuits in an AWGN Channel

Shaohua Hong · Zhiguo Shi · Lin Wang · Yujie Gu · Kangsheng Chen

Received: 3 July 2011 / Revised: 28 September 2012 / Published online: 20 October 2012

Abstract For chaotic trajectories, when the system parameters are fixed, they are generally confined in a bounded state space. In this paper, we propose an adaptive regularized particle filter (RPF), which makes the best of this inherent characteristic, for identical synchronization of chaotic Colpitts circuits combating additive white Gaussian noise (AWGN) channel distortion. This proposed filter incorporates RPF that resamples from a continuous approximation of the posterior density to avoid sample impoverishment and then utilizes the revised Kullback–Leibler distance (KLD) sampling to adaptively select the number of particles used. Compared with the existing particle filters (PFs) with fixed large number of particles, this proposed adaptive RPF propagates less number of particles with similar performance and thus provides a much more efficient solution for this problem.

Keywords Adaptive regularized particle filter · Chaos · Colpitts circuit · Synchronization

S. Hong (✉) · L. Wang
Department of Communication Engineering, Xiamen University, Xiamen 361005, P.R. China
e-mail: hongsh@xmu.edu.cn

L. Wang
e-mail: wanglin@xmu.edu.cn

Z. Shi · K. Chen
Department of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310027, P.R. China

Z. Shi
e-mail: shizg@zju.edu.cn

K. Chen
e-mail: chenks@zju.edu.cn

Y. Gu
School of Engineering, Bar-Ilan University, Ramat-Gan 52900, Israel
e-mail: guyujie@hotmail.com

1 Introduction

Since the work of Pecora and Carroll on chaos synchronization [15], considerable interest has been displayed in this research topic for its potential applications in radar, signal encryption, communications, etc. [9, 18]. As one of the standard oscillators, the synchronization of Colpitts circuit [8] is a hot research topic [2, 11, 17, 19, 23]. Both mathematical analysis and simulation results have confirmed that perfect synchronization can be achieved under ideal synchronization conditions [11, 19], and it was found that with minor parameter mismatch, chaos synchronization can still be obtained and maintained [2, 23]. However, in chaos synchronization with an additive white Gaussian noise (AWGN) channel, the traditional algorithms using the same or partly the same circuit in the receiver as a response system show unacceptable performance [17].

Recently, recursive Bayesian approaches used to construct a receiver posed another way for this problem [5, 10, 20, 21, 24, 25], especially the particle filters (PFs) that are the state-of-the-art solution to nonlinear and non-Gaussian problems [1, 16]. In 2006, Kurian and Puthusserypady investigated the unscented Kalman filter (UKF) and PF for chaotic synchronization, where the Lorenz and Mackey–Glass (MG) systems as well as the Ikeda map (IM) were used for the numerical evaluation [10]. In 2008, Shi et al. studied the PF-based synchronization of chaotic Colpitts circuits with AWGN channel distortion [21], and further demonstrated experimentally the synchronization of chaotic Chua's circuit utilizing the PF-based algorithm [20]. In 2009, application of PF to noisy synchronization in polynomial chaotic maps was investigated and the results indicated that the PF outperformed all other Kalman structured observers in the case of noisy channels [24, 25].

It is well known that a common problem with PFs is the degeneracy phenomenon, where as time increases, only very few particles are substantial. Although resampling can reduce the effects of degeneracy, it also introduces a practical problem known as sample impoverishment. For the chaotic oscillation, since the uncertainty of process model is generally very small, this problem is particularly serious. In [21], a roughening scheme [4] was introduced after resampling to improve the diversity of particles, and simulation results showed that chaos synchronization of Colpitts circuits can be achieved and maintained in an AWGN channel using the PF with the roughening scheme. However, the constant tuning parameter in the roughening algorithm is empirical and difficult to be determined appropriately. Too large a value would blur the distribution, meanwhile too small a value would produce tight clusters of points and the roughening is useless. Moreover, since the complexity of the probability densities varies drastically over time, with a fixed number of particles one has to choose large sample sets so as to get acceptable performance, and the computational load is directly proportional to this number. To solve these problems, an adaptive regularized particle filter (RPF)-based synchronization of chaotic Colpitts circuits is proposed in this paper. Instead of resampling from a discrete approximation of the posterior density, this proposed filter introduces RPF that resamples from the continuous approximation and avoids the problem of loss of diversity among the particles. Additionally, the proposed adaptive RPF incorporates the revised Kullback–Leibler distance (KLD) sampling [22] to adaptively select the number of particles used. That

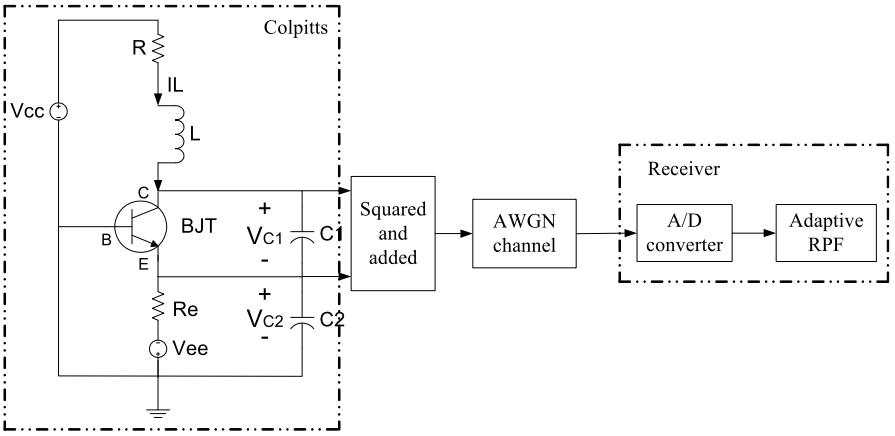

Fig. 1 Synchronization of chaotic Colpitts circuits in an AWGN channel

is to say, it propagates small number of particles when the density is concentrated in a small region of the state space and chooses large number of particles in cases of high uncertainties. Thus it propagates less number of particles and becomes much more efficient.

This paper is organized as follows. In Sect. 2, the problem formulation of the synchronization of chaotic Colpitts circuits subjected to an AWGN channel is presented. Section 3 describes the proposed adaptive RPF in considerable detail. The simulation results of synchronization process, synchronization performance and number of particles used over AWGN channel are presented in Sect. 4, and Sect. 5 concludes the paper.

2 Problem Formulation

The configuration of the synchronization of chaotic Colpitts circuits in an AWGN channel is shown in Fig. 1.

For the chaotic Colpitts oscillator, its state equations are as follows:

$$\begin{cases} \dfrac{dV_{C_1}(t)}{dt} = \dfrac{1}{C_1}\big(-f\big(-V_{C_2}(t)\big) + I_L(t)\big), \\ \dfrac{dV_{C_2}(t)}{dt} = \dfrac{1}{C_2}\left(I_L(t) - \dfrac{V_{C_2}(t) - V_{ee}}{R_e}\right), \\ \dfrac{dI_L(t)}{dt} = \dfrac{1}{L}\big(-V_{C_1}(t) - V_{C_2}(t) - I_L(t)R + V_{CC}\big), \end{cases} \tag{1}$$

where $f(.)$ is the driving-point characteristic of the nonlinear resistor of the bipolar junction transistor (BJT) [13], described by

$$f(x) = I_s\left[\exp\left(\frac{x}{V_T}\right) - 1\right] \approx I_s\left[\exp\left(\frac{x}{V_T}\right)\right], \quad \text{if } x \gg V_T. \tag{2}$$

In the Colpitts oscillator, this characteristic can be expressed as

$$I_E = f(V_{BE}) = f(-V_{C_2}) = I_S \exp\left(-\frac{V_{C_2}(t)}{V_T}\right). \tag{3}$$

To get more information about the states of chaotic Colpitts circuits in the receiver, the sum of $V_{C_1}^2(t)$ and $V_{C_2}^2(t)$ is selected as the transmitted signal. After it goes through the AWGN channel and reaches the receiver, it becomes

$$V_{\text{arrived}}(t) = V_{C_1}^2(t) + V_{C_2}^2(t) + v(t), \tag{4}$$

where $v(t)$ is the zero mean white Gaussian noise induced by the AWGN channel.

In the receiver, the analog signal is firstly sampled by an analog–digital converter (ADC) with time (sampling) interval T, and then the discrete-time signal is used as the observation for the adaptive RPF. According to Eq. (1), the discrete state dynamics for the adaptive RPF can be formulated as

$$\mathbf{x}_k = \boldsymbol{\Phi}\mathbf{x}_{k-1} + \mathbf{G}, \tag{5}$$

where

$$\mathbf{x}_k = \begin{pmatrix} V_{C_1}(kT) \\ V_{C_2}(kT) \\ I_L(kT) \end{pmatrix}, \qquad \boldsymbol{\Phi} = \begin{pmatrix} 1 & 0 & T/C_1 \\ 0 & 1 - T/(C_2R_e) & T/C_2 \\ -T/L & -T/L & 1 - RT/L \end{pmatrix},$$

$$\mathbf{G} = \begin{pmatrix} -f(-V_{C_2}((k-1)T))T/C_1 \\ V_{ee}T/(C_2R_e) \\ V_{CC}T/L \end{pmatrix}.$$

And the measurement model is given by

$$z_k = \left(\mathbf{x}_k[1]\right)^2 + \left(\mathbf{x}_k[2]\right)^2 + v_k, \tag{6}$$

where $\mathbf{x}_k[i]$ denotes the ith element of the vector $\mathbf{x}_k$.

In numerical simulations, the parameters of the chaotic Colpitts oscillator are set to be the same as those in [21]; that is, $C_1 = C_2 = 237$ nF, $L = 2.1$ mH, $R = 74.5\ \Omega$, $V_{CC} = 5$ V, $V_{ee} = -5$ V, $R_e = 2000\ \Omega$. The time interval T is 1 μs. With these parameters, the transmitter exhibits chaotic oscillation. It has been shown that under an ideal channel, perfect synchronization can be achieved using one-way coupling linear error feedback synchronization scheme [17]. Unfortunately, it is very difficult to achieve synchronization when the AWGN channel effect is exerted.

3 Adaptive Regularized Particle Filter Algorithm

Unlike the conventional analytical approximation methods, PFs base their operation on representing the target's posterior probability density function (PDF) of the state

by a set of random particles $\{\mathbf{x}_k^j\}_{j=1}^N$ with their associated weights $\{w_k^j\}_{j=1}^N$, where j is the particle index and N is the number of particles used, i.e.,

$$p(\mathbf{x}_k|\mathbf{z}_k) = \frac{1}{W_k}\sum_{j=1}^{N} w_k^j \delta\big(\mathbf{x}_k - \mathbf{x}_k^j\big), \tag{7}$$

where $W_k = \sum_{j=1}^N w_k^j$. In the implementation of PFs, there are three important operations:

(1) Sampling: generation of new particles $\mathbf{x}_k^j \sim q(\mathbf{x}_k|\mathbf{x}_{k-1}^j, \mathbf{z}_{1:k})$ for $j = 1, \ldots, N$, where $q(\mathbf{x}_k|\mathbf{x}_{k-1}^j, \mathbf{z}_{1:k})$ is the importance density. The most popular choice is the transitional prior $q(\mathbf{x}_k|\mathbf{x}_{k-1}^j, \mathbf{z}_k) = p(\mathbf{x}_k|\mathbf{x}_{k-1}^j)$, which is also used in this paper.
(2) Weight calculation: computation of the particle weights

$$w_k^j = w_{k-1}^j \frac{p(\mathbf{z}_k|\mathbf{x}_k^j)\, p(\mathbf{x}_k^j|\mathbf{x}_{k-1}^j)}{q(\mathbf{x}_k|\mathbf{x}_{k-1}^j, \mathbf{z}_{1:k})} \tag{8}$$

followed by normalization $w_k^j = w_k^j[\sum_{j=1}^N w_k^j]^{-1}$. Because the prior distribution is adopted as importance distribution in this paper, the importance weights will satisfy $w_k^j = w_{k-1}^j p(\mathbf{z}_k|\mathbf{x}_k^j)$, which reduces the complexity of the PFs.
(3) Resampling: drawing new particles $\{\tilde{\mathbf{x}}_k^j\}_{j=1}^N$ from the above set of particles $\{\mathbf{x}_k^j\}_{j=1}^N$ based on the particle weights according to a resampling algorithm.

3.1 Regularized Particle Filter

Resampling is a method to reduce the degeneracy problem. The key idea is to discard the particles with low normalized importance weights and multiply the particles with high normalized importance weights to replace them. However, it in turn introduces the problem of loss of diversity among the particles, especially when the uncertainty of process model is small. This arises due to the fact that in the resampling step, samples are drawn from a discrete distribution rather than a continuous one. A modified PF known as the RPF, which resamples from a continuous approximation of the posterior density, was proposed as a potential solution to this problem [14]. Except for the resampling step, the RPF is identical to the PF. The details of the RPF are described in Table 1. For more details of discussion on the RPF, please refer to [14].

In Table 1, N_T is the pre-specified threshold of the sample size and h_{opt} is the Kernel bandwidth.

3.2 Adaptive Selection of the Number of Particles Used

Most existing approaches to implement PFs use a fixed number of particles during the entire state estimation process. It is highly inefficient in most situations, because the complexity of the posterior probability densities varies drastically over time and the complexity of PFs depends on the number of particles used for estimation. The

Table 1 Regularized particle filter [14]

$\left[\{\mathbf{x}_k^{j*}, w_k^j\}_{j=1}^N\right] = \mathrm{RPF}\left[\{\mathbf{x}_{k-1}^j, w_{k-1}^j\}_{j=1}^N, \mathbf{z}_k\right]$

- **Sampling**:
  - for $j = 1, \ldots, N$
    - draw $\mathbf{x}_k^j \sim q(\mathbf{x}_k | \mathbf{x}_{k-1}^j, \mathbf{z}_{1:k})$
  - end for
- **Weight calculation**:
  - Assign the particle a weight w_k^j according to (8)
  - Normalize:
    - for $j = 1, \ldots, N$
      - ✓ $w_k^j = w_k^j \left[\sum_{j=1}^N w_k^j\right]^{-1}$
    - end for
- **The effective sample size $\hat{N}_{\mathrm{eff}}$ calculating**:
  - $\hat{N}_{\mathrm{eff}} = \dfrac{1}{\sum_{j=1}^N (w_k^j)^2}$
- **Resampling:**
  - if $\hat{N}_{\mathrm{eff}} < N_T$
    - Calculate the empirical covariance matrix S_k of $\{\mathbf{x}_k^j, w_k^j\}_{j=1}^N$
    - Compute D_k such that $D_k D_k^T = S_k$
    - $\left[\left\{\mathbf{x}_k^j, w_k^j, -\right\}_{j=1}^N\right] = \mathrm{RESAMPLE}\left[\left\{\mathbf{x}_k^j, w_k^j\right\}_{j=1}^N\right]$
    - for $j = 1, \ldots, N$
      - ✓ Draw $\xi^j \sim K$ from the Epanechnikov kernel
      - ✓ $\mathbf{x}_k^{j*} = \mathbf{x}_k^j + h_{\mathrm{opt}} D_k \xi^j$
    - end for
  - end if

likelihood-based adaptation [3] can be utilized to adaptively select the number of particles used, the idea of which is to use the sum of non-normalized importance weights as a measure until the sum of the weights exceeds a pre-specified threshold. However, the pre-specified threshold is empirical and difficult to be set appropriately. Furthermore, the criterion only considers the quality of the match between the proposed and true distribution but ignores the other important factors, i.e., the complexity of the true density [22]. In this section, an adaptive scheme, which utilizes the revised KLD-Sampling [22] to adaptively select the size of sample sets, is incorporated to increase the efficiency.

3.2.1 KLD-Sampling

Assume that the true posterior density is given by a discrete, piecewise constant distribution $\mathbf{p} = (p_1, p_2, \ldots, p_m)$ over partitioned state spaces (called bins), where m is the number of bins. Let vector $\mathbf{n} = (n_1, n_2, \ldots, n_m)$ be the number of particles from each bin. Then maximum likelihood estimate (MLE) of the density $\mathbf{p}$ using the $\mathbf{n}$ samples can be given by $\hat{\mathbf{p}} = \mathbf{n}/N$. The KLD between the MLE and true posterior is

given by

$$K(\hat{\mathbf{p}}, \mathbf{p}) = \sum_{i=1}^{m} \hat{p}_i \log\left(\frac{\hat{p}_i}{p_i}\right). \tag{9}$$

Given pre-specified error threshold ε, the probability

$$\text{Prob}\left\{K(\hat{\mathbf{p}}, \mathbf{p}) < \varepsilon\right\} = 1 - \delta \tag{10}$$

holds if the total number of particles N meets the chi-square distribution

$$N = \frac{1}{2}\chi^2_{m-1,1-\delta}. \tag{11}$$

Using the Wilson–Hilferty transformation to approximate the quantiles of the $\chi^2_{m-1,1-\delta}$ distribution, the bound given by the KLD-Sampling for the number of particles is:

$$N = \frac{1}{2\varepsilon}\chi^2_{m-1,1-\delta} \approx \frac{(m-1)}{2\varepsilon}\left\{1 - \frac{2}{9(m-1)} + \sqrt{\frac{2}{9(m-1)}}z_{1-\delta}\right\}^3, \tag{12}$$

where $z_{1-\delta}$ is the upper $1-\delta$ quantile of the standard normal distribution and its values are readily available in standard statistical tables.

3.2.2 Revised KLD-Sampling

It can be found that a key feature of the bound given by the KLD-Sampling is that it does not require direct knowledge of the true posterior. It only requires knowledge about the number of bins with support, which is suitable for the PF-based synchronization algorithms for chaotic circuits, because when the system parameters are fixed, the chaotic trajectories are generally confined in a bounded state space. However, there is a problem with the KLD-Sampling; that is, the derivation of the bound has the implicit assumption that the samples come from the true distribution, whereas the samples in PFs come from an importance function. Moreover, the quality of the match between the true and the proposed distribution is one of the main elements that determines the accuracy of the filter, and hence the suitable number of particles required.

To fix this problem of the KLD-Sampling, the equivalent number of particles from the importance and true densities is given by [22]

$$N_q = N_p \frac{\text{Var}_q(x)}{\text{Var}_p(x)}, \tag{13}$$

where N_q and N_p are the numbers of particles coming from the importance and the true distribution, respectively, $\text{Var}_q(x) = E_q((x - E_p(x))^2 w_q^2)$ and $\text{Var}_p(x) = E_p((x - E_p(x))^2)$ are the variances of particles drawn from the importance and the true distribution, respectively, $E_p(x) = \sum_{j=1}^{n} x_j w_q^j / \sum_{j=1}^{n} w_q^j$ and $w_q = p(x)/q(x)$ with the true distribution $p(x)$ and the importance distribution $q(x)$.

If we assume that the weights are independent of x, the variances can be stated in simpler terms as follows:

$$\begin{aligned}\mathrm{Var}_q(x) &= E_q\big(\big(x - E_p(x)\big)^2 w_q^2\big) \approx E_q\big(\big(x - E_p(x)\big)^2\big) \cdot E_q\big(w_q^2\big) \\ &= E_q\big(\big(x - E_p(x)\big)^2\big) \cdot \big(\mathrm{Var}_q(w_q) + E_q(w_q)^2\big) \\ &= E_q\big(\big(x - E_p(x)\big)^2\big) \cdot \big(\mathrm{Var}_q(w_q) + 1\big) \end{aligned} \tag{14}$$

and

$$\begin{aligned}\mathrm{Var}_p(x) &= E_p\big(\big(x - E_p(x)\big)^2\big) \approx E_q\big(w_q\big(x - E_p(x)\big)^2\big) \\ &= E_q(w_q) \cdot E_q\big(\big(x - E_p(x)\big)^2\big) = E_q\big(\big(x - E_p(x)\big)^2\big). \end{aligned} \tag{15}$$

Substituting (14) and (15) into (13), the bound given by the revised KLD-Sampling in the case that the samples are drawn from an importance function is:

$$N_q = \big(1 + \mathrm{Var}_q(w_q)\big) \cdot N_p \tag{16}$$

which coincides with the result in [12, 22]. Considering the weights $w_k = w_{k-1} \times \frac{p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1})}{q(\mathbf{x}_k|\mathbf{x}_{k-1},\mathbf{z}_{1:k})} \propto w_q$ inPFs, the adjustment factor $1 + \mathrm{Var}_q(w_q)$ can be obtained as follows:

$$\begin{aligned}1 + \mathrm{Var}_q(w_q) &= 1 + \mathrm{Var}_q(w_k)/E_q(w_k)^2 \\ &= E_q\big(w_k^2\big)/E_q(w_k)^2 \\ &= n \cdot \sum_{j=1}^{n}\big(w_k^j\big)^2 \Big/ \left(\sum_{j=1}^{n} w_k^j\right)^2. \end{aligned} \tag{17}$$

Substituting (17) into (16), we can obtain

$$N_q = n \cdot \sum_{j=1}^{n}\big(w_k^j\big)^2 \Big/ \left(\sum_{j=1}^{n} w_k^j\right)^2 \cdot \frac{(m-1)}{2\varepsilon}\left\{1 - \frac{2}{9(m-1)} + \sqrt{\frac{2}{9(m-1)}}z_{1-\delta}\right\}^3. \tag{18}$$

It is found that as in the case of the KLD-Sampling, the bound given by the revised KLD-Sampling can also be estimated incrementally as the particles and their weights are available.

3.3 The Proposed Adaptive RPF

For chaotic oscillators, when the system parameters are fixed, the chaotic trajectories are generally confined in a bounded state space Ω. Thus the samples outside the attractor region make no contribution for the posterior probability density and we can easily get the bin size for the revised KLD-Sampling. It is clear that the choice of

the size of bins is a compromise. If the size of bins is relatively small, the number of bins with support would be large and then more number of particles would be required. On the other hand, if the size of bins is too large, the number of bins with support may be small and then the number of particles used may be insufficient. In general, the scheme of setting the size of bins small and utilizing more number of particles shows a better performance. In this paper, the bounded state space is averagely partitioned into $10 \times 10 \times 10$ bins for simplification of simulation. Considering that the number of particles used is time-varying, resampling is applied at each time index and the threshold-based resampling algorithm [6, 7] is utilized to adaptively adjust the output number of particles in the resampling procedure. And to reduce the computing cost of generating from the regularized measure, the Epanechnikov kernel is replaced by the Gaussian kernel and the optimal bandwidth associated (when the underlying density is Gaussian with unit covariance matrix) is $h_{\text{opt}} = A(K) \cdot N^{-\frac{1}{n_{\mathbf{x}}+4}}$ with $A(K) = (4/(n_{\mathbf{x}}+2))^{\frac{1}{n_{\mathbf{x}}+4}}$, where $n_{\mathbf{x}}$ is the dimension of the variable. For the chaotic Colpitts circuits in this paper, $n_{\mathbf{x}} = 3$, and thus the optimal bandwidth is $h_{\text{opt}} = (4/5)^{\frac{1}{7}} \cdot N^{-\frac{1}{7}}$. The details of the proposed adaptive RPF for synchronization of chaotic Colpitts circuits over AWGN channel are described in Table 2.

In Table 2, $\widehat{\mathbf{x}}_k^s$ represents the substantial particle and $\widehat{w}_k^s$ is the weight of the substantial particle. It can be observed from Table 2 that the proposed adaptive RPF adaptively adjusts the sample size through the revised KLD-Sampling, and the output number of particles in the resampling procedure is also adaptive by using the threshold-based resampling algorithm [6, 7]. Therefore, the computational complexity is reduced and the efficiency is improved.

4 Simulation and Discussion

In this section, we will show how synchronization is achieved and how the number of particles used changes when the proposed adaptive RPF is employed as receiver. The effects of different SNRs, different ADC sampling intervals on the synchronization performance and the number of particles used are also investigated.

To evaluate the synchronization performance, we define the synchronization error E and average attractor distance (AAD) D:

$$E_k = \sqrt{\big(\mathbf{e}_k[1]\big)^2 + \big(\mathbf{e}_k[2]\big)^2 + \big(\mathbf{e}_k[3]\big)^2}, \tag{19}$$

$$D = \lim_{k_x \to \infty} \frac{\sum_{k=k_0}^{k_x} E_k}{k_x - k_0 + 1}, \tag{20}$$

where $\mathbf{e}_k[1] = \hat{\mathbf{x}}[1] - \mathbf{x}[1]$, $\mathbf{e}_k[2] = \hat{\mathbf{x}}[2] - \mathbf{x}[2]$, $\mathbf{e}_k[3] = \hat{\mathbf{x}}[3] - \mathbf{x}[3]$, and k_0 denotes the time index when the transient parts of the signals have passed. The value of D will be zero when the transmitter and receiver are in perfect identical synchronization state. But if the synchronization performance is not very good, the variable D will be non-zero, and a bigger value of D means a worse synchronization performance.

Table 2 The proposed adaptive RPF

Input: bounds ε and δ, bin size Δ, maximum number of particles $N_{\max}$

- For $k = 1, 2, \ldots$
  - (1) **Initializing**:
    - $n = 1$, $m = 0$, $r = 1$, $N_q = 0$, $W_k = 0$, $W_k^2 = 0$
  - (2) **Generating samples**:
    - (a) Sampling:
      - ✓ if $(k > 1)$ //Draw state from previous belief after resampling
        - ▹ if $(n < s + 1)$
          - ◇ $\mathbf{x}_{k-1}^n = \widehat{\mathbf{x}}_{k-1}^r$
          - ◇ $w_{k-1}^n = \widehat{w}_{k-1}^r$
        - ▹ else //Draw from a continuous approximation of the posterior
          - ◇ draw $\xi \sim K$ from the Gaussian kernel
          - ◇ $\mathbf{x}_{k-1}^n = \widehat{\mathbf{x}}_{k-1}^r + h_{\text{opt}} D_{k-1} \xi$
          - ◇ $w_{k-1}^n = \widehat{w}_{k-1}^r$
        - ▹ end if
        - ▹ $r = \text{mod}(r, s) + 1$ //Repeat sequentially
      - ✓ end if
      - ✓ Sample $\mathbf{x}_k^n \sim p(\mathbf{x}_k | \mathbf{x}_{k-1}^n)$ using $\mathbf{x}_{k-1}^n$
    - (b) Weight computing:
      - ✓ if $\mathbf{x}_k^n \in$ bounded space Ω
        - ▹ $w_k^n = w_{k-1}^n p(\mathbf{z}_k | \mathbf{x}_k^n)$
      - ✓ else
        - ▹ repeat sampling step
      - ✓ end if
      - ✓ $W_k = W_k + w_k^n$ //Update normalization factor
      - ✓ $W_k^2 = W_k^2 + (w_k^n)^2$
    - (c) Number of bins with support updating:
      - ✓ if ($\mathbf{x}_k^n$ falls into empty b)
        - ▹ $m = m + 1$ //Update the number of bins
        - ▹ $b = non\text{-}empty$ //Mark bin
      - ✓ end if
      - ✓ if $(m \geq 2)$ //Update the number of desired particles
        - ▹ $N_q = \dfrac{n \cdot W_k^2}{(W_k)^2} \cdot \dfrac{(m-1)}{2\varepsilon} \left\{ 1 - \frac{2}{9(m-1)} + \sqrt{\frac{2}{9(m-1)}} z_{1-\delta} \right\}^3$
      - ✓ end if
    - (d) KLD bound checking:
      - ✓ if ($n \geq N_q$ or $n \geq N_{\max}$)
        - ▹ break
      - ✓ else
        - ▹ Update number $n = n + 1$, and repeat step (2)
      - ✓ end if

Table 2 (*Continued*)

- (3) **Normalizing**:
  - for $j = 1, \ldots, n$
    - $w_k^j = w_k^j / W_k$
  - end for
- (4) **Calculating Estimation**:
  - $\hat{\mathbf{x}} = E[\mathbf{x}_k | \mathbf{z}_{1:k}] = \sum_{j=1}^{n} w_k^j \times \mathbf{x}_k^j$
- (5) **Calculating the empirical covariance matrix** S_k **of** $\{\mathbf{x}_k^j, w_k^j\}_{j=1}^n$
- (6) **Computing** D_k **such that** $D_k D_k^T = S_k$
- (7) **Resampling**:
  - (a) Set threshold $T < 1/n$, $s = 0$
  - (b) Select substantial particles:
    - for $j = 1 : n$
      - if $(w_k^j > T)$
        - $s = s + 1$ //the number of substantial particles
        - $\bar{\mathbf{x}}_k^s = \mathbf{x}_k^j$
        - $\bar{w}_k^s = w_k^j$
      - end if
    - end for
- end For

4.1 Time Evolution of Synchronization

With the parameters listed in Sect. 2, the transmitter can exhibit chaotic oscillation. When the transmitted signal passes through an AWGN channel with SNR = 10 dB and reaches the receiver constructed by the proposed adaptive RPF described in Sect. 3 with error threshold $\varepsilon = 0.03$ and $\delta = 0.03$, and maximum number of particles $N_{\max} = 1000$, the simulation results are plotted in Fig. 2.

It follows from Fig. 2 that the transmitter's state can be estimated and tracked in a very short settling time, indicating that synchronization can be obtained and maintained utilizing the proposed adaptive RPF. This is to say, the resampling from a continuous approximation of the posterior density can avoid sample impoverishment in PF-based synchronization of chaotic Colpitts circuits. Figure 3 shows particle distributions after 1000 time steps using the proposed adaptive RPF. It is found that the particles disperse relatively evenly, and the distribution of particles can cover the true state.

4.2 Synchronization Performance vs. SNR, Sampling Interval

It is well known that synchronization performance is very sensitive to noise in the traditional drive–response system. Moreover, in the digital-filter-based synchronization approaches, it is inevitable to lose some information during the ADC conversion, thus the sampling interval of the ADC affects the synchronization performance. In this

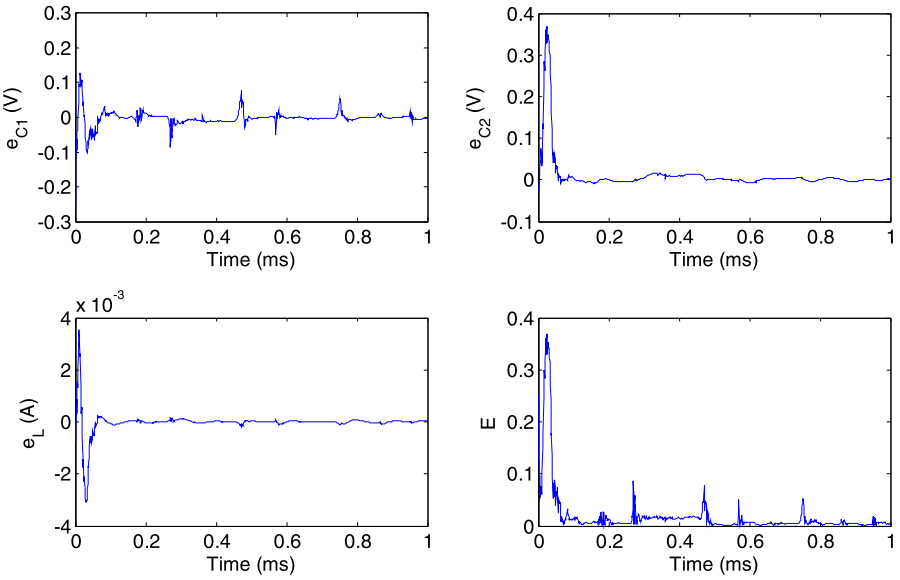

Fig. 2 With the proposed adaptive RPF, error signals: (**a**) $e_{C_1} = V_{C_1} - \hat{V}_{C_1}$; (**b**) $e_{C_2} = V_{C_2} - \hat{V}_{C_2}$; (**c**) $e_L = I - \hat{I}$; (**d**) the synchronization error E

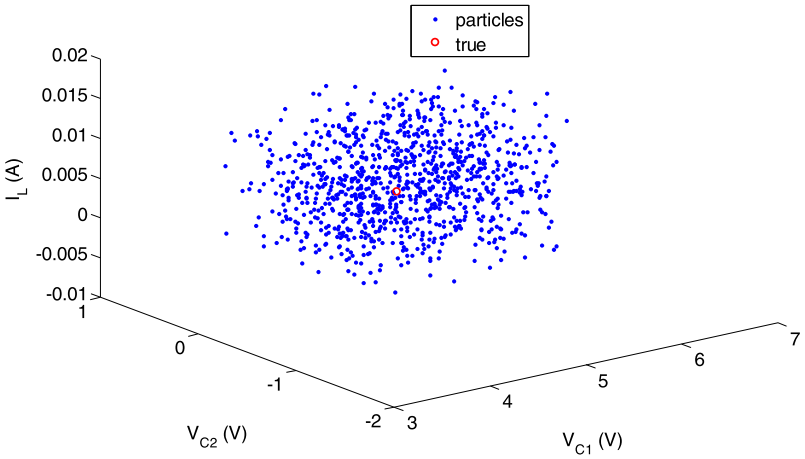

Fig. 3 Particle distributions after 1000 time steps using the proposed adaptive RPF

section, we will study the effect of different SNRs and different sampling intervals on the synchronization performance by observing the value of AAD. For comparison, the PF with roughening scheme [21], the RPF with a fixed large number of samples, and the adaptive RPF using the KLD-Sampling are also studied.

Figure 4 shows the simulation results of AAD vs. SNR. All the data here are obtained through 500 Monte Carlo simulations. Note that even when the AAD has a

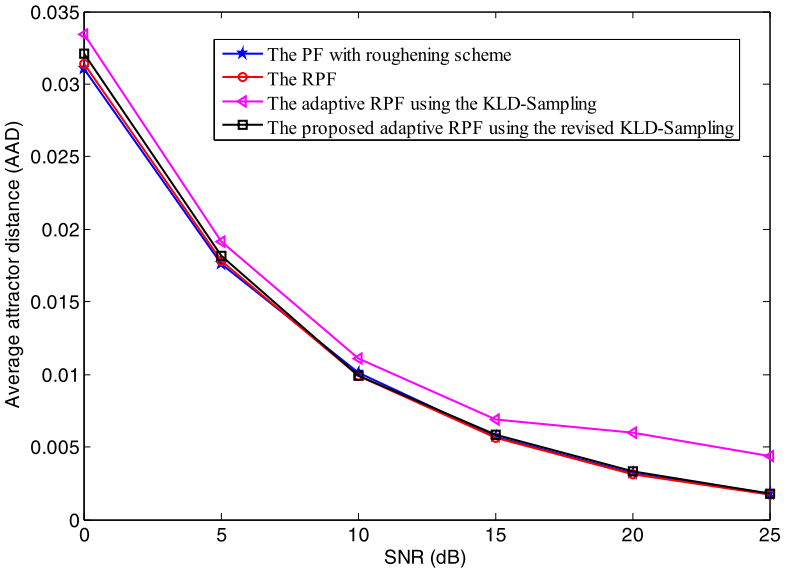

Fig. 4 AAD vs. SNR

value of 0.05, from the time-domain waveform of the drive and response we can see that synchronization performance is quite good. It is seen from Fig. 4 that compared with the PF with roughening scheme [21] and the RPF with a fixed 1000 number of particles, the proposed adaptive RPF using the revised KLD-Sampling shows a similar performance. And the performances of all these three schemes are better than that of the adaptive RPF using the KLD-Sampling especially at high SNRs. It is because that the KLD-Sampling ignores the quality of the match between the true (posterior) and the proposed (prior) distribution and when the SNR is very high, the likelihood function is too narrow and the quality of the match is lower. It can be also observed from Fig. 4 that with the decrement of SNR the synchronization performance degrades at a rather slow rate. Thus it can be inferred that the proposed adaptive RPF is effective when the AWGN distortion is considered.

The AAD vs. the ADC sampling interval is plotted in Fig. 5. In this simulation, the SNR is assumed to be 20 dB. It shows again that the proposed adaptive RPF has almost similar performance to those of the PF with roughening scheme [21] and the RPF with a fixed 1000 number of particles, while the adaptive RPF using the KLD-Sampling exhibits worse synchronization performance. It is seen from Fig. 5 that with the decrease of the ADC sampling interval the synchronization performance improves quickly and the performance of the adaptive scheme using the KLD-Sampling is close to that of the proposed adaptive scheme. The reason can be that the shorter the ADC sampling interval, the higher precision the discrete data sampled from the analog signal.

From Figs. 4 and 5 one can conclude that the proposed adaptive RPF using the revised KLD-Sampling shows similar performance to those of the existing PFs with

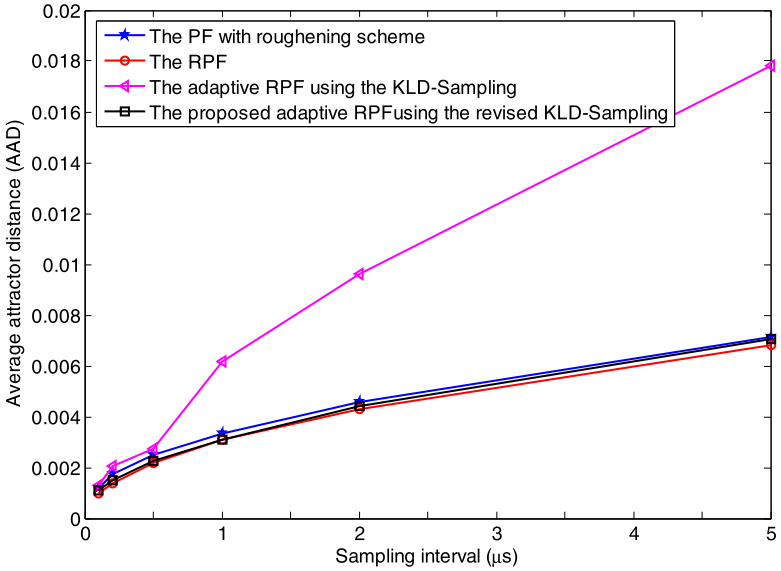

Fig. 5 AAD vs. ADC sampling interval

fixed large sample set sizes, and all of them are better than the adaptive scheme using the KLD-Sampling for this problem.

4.3 Number of Particles Used vs. SNR, Sampling Interval

In this section, the number of particles used will be studied when the adaptive RPF is employed as the receiver.

Figure 6 shows the sample set sizes during the estimation process using the proposed adaptive RPF for this problem in SNR = 10 dB. It is found that when the chaotic state is highly uncertain, such as at the beginning of a filtering, a large number of particles are desired to accurately represent its objective; while only a small number of particles suffice to accurately track its state in case of small uncertainties. The corresponding computational time in Matlab 7.7.0 on a Core 2 Duo 3.16-GHz PC with 4 GB RAM is shown in Table 3. Upon the inspection of Table 3, the proposed adaptive RPF is much faster than the existing PFs with fixed large number of particles. It is because that the proposed filter adaptively adjusts the number of particles used and propagates less number of particles, which consequently reduces the computational time. Obviously, compared with the existing PFs with a fixed large number of particles, the proposed adaptive RPF is much more efficient.

For different SNRs, the numbers of particles used in both of the adaptive filters are shown in Table 4. It is found that the proposed adaptive RPF using the revised KLD-Sampling requires more number of particles than the adaptive scheme using the KLD-Sampling. It is because that the proposed scheme considers the quality of the match between the true and proposed distribution, which is one of the main elements determining the accuracy of the filter, and hence the number of particles required.

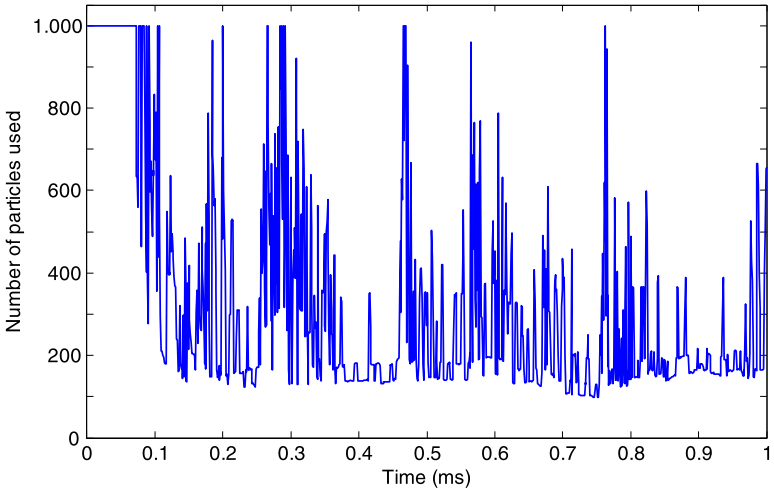

Fig. 6 Time evolution of number of particles used

Table 3 Computational time

Filters	Computational time (s)
The PF with roughening scheme [21]	84.8
The RPF	88.3
The proposed adaptive RPF using the revised KLD-Sampling	35.6

Table 4 The average number of particles used vs. SNR

Filters	SNR (dB)					
	0	5	10	15	20	25
The adaptive RPF using the KLD-Sampling	359	221	142	95	80	69
The proposed adaptive RPF using the revised KLD-Sampling	746	529	353	247	196	171

However, compared with the PF with roughening scheme [21] and the RPF with fixed 1000 number of particles during the entire state estimation process, the proposed adaptive RPF propagates less number of particles. We can also find that in both of the adaptive schemes, the average number of particles used decreases with the increase of the SNR. The reason can be that the SNR is smaller; that is, the channel noise is rather bigger and the density has higher uncertainties, thus larger number of particles is required.

The corresponding numbers of particles used vs. ADC sampling interval for this problem in SNR = 20 dB are shown in Table 5. One can observe again that the number of particles used in the proposed adaptive scheme is more than that in the adap-

Table 5 The average number of particles used vs. ADC sampling interval

Filters	Sampling interval (μs)					
	0.1	0.2	0.5	1	2	5
The adaptive RPF using the KLD-Sampling	65	67	71	80	85	117
The proposed adaptive RPF using the revised KLD-Sampling	154	158	174	196	235	328

tive scheme using the KLD-Sampling. And the proposed adaptive RPF propagates less number of particles than the existing PFs with fixed 1000 number of particles. It is seen from Table 5 that the average number of particles used increases with the increase of the ADC sampling interval. The reason can be that the ADC sampling interval is shorter, namely the PF state sampling interval is shorter and the particles coming from the importance function are more precise, thus the number of bins with support is correspondingly smaller and the bound given by the adaptive scheme for the number of particles is lower.

From the simulation results above it can be found that for chaos synchronization of Colpitts circuits in an AWGN channel, the proposed adaptive RPF, which utilizes the revised KLD-Sampling to adaptively select the number of particles used, can show similar performance to those of the existing PFs with fixed large number of particles but propagates less number of particles and thus is much more efficient.

5 Conclusion

In this paper we proposed an adaptive RPF utilizing the revised KLD-Sampling for synchronization of chaotic Colpitts circuits combating AWGN channel distortion. This proposed adaptive RPF makes the most of the inherent characteristic of chaotic trajectories, which are generally confined in a bounded state space with the fixed system parameters. Simulation results demonstrate that for the chaotic Colpitts oscillator, the proposed adaptive RPF avoids the sample impoverishment problem and is more efficient than the existing schemes with fixed large sample set sizes while keeping the performance approximately the same. The proposed adaptive RPF-based synchronization scheme has tremendous potential for developing practical chaotic communication systems using Colpitts circuits. In future work, we will demonstrate experimentally the synchronization of chaotic Colpitts circuits utilizing this proposed adaptive scheme and develop practical chaotic communication systems using the chaotic Colpitts circuits.

References

1. M.S. Arulampalam, S. Maskell, N. Gordon, T. Clapp, A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. IEEE Trans. Signal Process. **50**(2), 174–187 (2002)
2. A. Baziliauskas, R. Krivickas, A. Tamaševičius, Coupled chaotic Colpitts oscillators: identical and mismatched cases. Nonlinear Dyn. **44**, 151–158 (2006)

3. D. Fox, W. Burgard, F. Dellaert, S. Thrun, Monte Carlo localization: efficient position estimation for mobile robots, in *Proc. National Conf. Art. Intel. (AAAI)* (1999), pp. 343–349
4. N. Gordon, D. Salmond, Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEE Proc., F, Radar Signal Process. **140**(2), 107–113 (1993)
5. S.-H. Hong, Z.-G. Shi, K.-S. Chen, EKF-based dual synchronization of chaotic Colpitts circuit and Chua's circuit. Kybernetika **44**(4), 482–491 (2008)
6. S.-H. Hong, Z.-G. Shi, K.-S. Chen, Compact resampling algorithm and hardware architecture for particle filters, in *IEEE Int. Conf. Commun. Circ. Syst. (ICCCAS'08)* (2008), pp. 886–890
7. S.-H. Hong, Z.-G. Shi, K.-S. Chen, A low-power memory-efficient resampling architecture for particle filters. Circuits Syst. Signal Process. **29**, 155–167 (2010)
8. M.P. Kennedy, Chaos in the Colpitts oscillator. IEEE Trans. Circuits Syst. I **41**(11), 771–774 (1994)
9. G. Kolumban, M.P. Kennedy, L.O. Chua, The role of synchronization in digital communications using chaos—part II: chaotic modulation and chaotic synchronization. IEEE Trans. Circuits Syst. I **45**(11), 1129–1140 (1998)
10. A.P. Kurian, S. Puthusserypady, Unscented Kalman filter and particle filter for chaotic synchronization, in *IEEE Asian Pacific Conf. Circ. Syst. (APCAS)* (2006), pp. 1830–1834
11. G.-H. Li, Chaos and synchronization of Colpitts oscillators. Microw. Opt. Technol. Lett. **39**(6), 446–449 (2003)
12. J. Liu, Metropolized independent sampling with comparisons to rejection sampling and importance sampling. Stat. Comput. **6**, 113–119 (1996)
13. G.M. Maggio, O.E. Feo, M.P. Kennedy, Nonlinear analysis of the Colpitts oscillator and applications to design. IEEE Trans. Circuits Syst. I **46**(9), 1118–1130 (1999)
14. C. Musso, N. Oudjane, F. LeGland, Improving regularized particle filter, in *Sequential Monte Carlo Methods in Practice*, ed. by A. Doucet, J.F.G. de Freitas, N. Gordon (Springer, New York, 2001)
15. L.M. Pecora, T.L. Carroll, Synchronization in chaotic systems. Phys. Rev. Lett. **64**(8), 821–824 (1990)
16. B. Ristic, S. Arulampalam, N. Gordon, *Beyond the Kalman Filter: Particle Filter for Tracking Applications* (Artech House, Boston, 2004)
17. C. Shen, Z.-G. Shi, L.-X. Ran, Adaptive synchronization of chaotic Colpitts circuits against parameter mismatches and channel distortions. J. Zhejiang Univ. Sci. a **2**(S-II), 228–236 (2006)
18. Z.-G. Shi, L.-X. Ran, Microwave chaotic Colpitts oscillator: design, implementation and applications. J. Electromagn. Waves Appl. **20**(10), 1335–1349 (2006)
19. Z.-G. Shi, J.-T. Huangfu, L.-X. Ran, Performance comparison of two synchronization schemes for Colpitts circuit based chaotic communication system over noisy channel, in *5th WCICA 6* (2004), pp. 1276–1279
20. Z.-G. Shi, S.-H. Hong, K.-S. Chen, Experimental study on tracking the state of analog Chua's circuit with particle filter for chaos synchronization. Phys. Lett. A **372**, 5575–5580 (2008)
21. Z.-G. Shi, S.-H. Hong, J.-M. Chen, K.-S. Chen, Y.-X. Sun, Particle filter-based synchronization of chaotic Colpitts circuits combating AWGN channel distortion. Circuits Syst. Signal Process. **27**(6), 833–845 (2008)
22. A. Soto, Self adaptive particle filter, in *Proc. Int. Joint Conf. Art. Intel (IJCAI'05)* (2005)
23. A. Uchida, K. Takahashi, M. Kawano, S. Yoshimori, Synchronization of chaos in one-way coupled Colpitts oscillator. IEICE Trans. Fundam. Electron. Commun. Comput. Sci. **E85**(A9), 2072–2077 (2002)
24. M. Yahia, P. Acco, Particle filter applied to polynomial chaotic maps, in *IEEE European Conf. Circ. Theory and Design (ECCTD)* (2009), pp. 535–538
25. M. Yahia, P. Acco, M. Benslama, Particle filter applied to noisy synchronization in polynomial chaotic maps. Int. J. Inf. Commun. Eng. **5**, 354–358 (2009)