

Research Article

Robust Blind Frequency and Transition Time Estimation for Frequency Hopping Systems

Kuo-Ching Fu and Yung-Fang Chen

Department of Communication Engineering, National Central University, no.300, Jung-da Road, Jung-li City, Taoyuan 32001, Taiwan

Correspondence should be addressed to Yung-Fang Chen, yfchen@ce.ncu.edu.tw

Received 27 April 2010; Revised 24 September 2010; Accepted 5 November 2010

Academic Editor: Kutluyil Dogancay

Copyright © 2010 K.-C. Fu and Y.-F. Chen. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In frequency hopping spread spectrum (FHSS) systems, two major problems are timing synchronization and frequency estimation. A blind estimation scheme is presented for estimating frequency and transition time without using reference signals. The scheme is robust in the sense that it can avoid the unbalanced sampling block problem that occurs in existing maximum likelihood-based schemes, which causes large errors in one of the estimates of frequency. The proposed scheme has a lower computational cost than the maximum likelihood-based greedy search method. The estimated parameters are also used for the subsequent time and frequency tracking. The simulation results demonstrate the efficacy of the proposed approach.

1. Introduction

Frequency hopping spread spectrum (FHSS) techniques are widely used in military communications for combating narrowband interference and for security purposes. The two parameters that are required for the estimation in FHSS are transition time and hopping frequency. Regular synchronization is divided into two stages—coarse acquisition and fine tracking [1]. Reference signals may be used to estimate the parameters [2–6], but they may not be available in all cases. Moreover, since the usage of reference signals requires bandwidth, it reduces the bandwidth efficiency. To improve spectral utilization, several researchers [7–13] have proposed some algorithms for blind estimation. Liang et al. [7] proposed a revisable jump Markov chain Monte Carlo (RJMCMC-) based algorithm for estimating frequency and timing parameters. However, it requires that the hyperparameter is known in advance. Liu et al. [8] used an antenna array and the expectation-maximization (EM) algorithm to estimate multiple FH signals, but the computational complexity was high. Mallat and Zhang [10] used the matching pursuit (MP) method that decomposes the signal into a linear expansion of time-frequency components; however, this algorithm needs to select a discrete subset of

possible dictionary functions for practical implementation [10]. Liu et al. [11] proposed a joint hop-timing and frequency estimation method that was based on the principle of dynamic programming (DP) coupled with 2D harmonic retrieval (HR) using antenna arrays. The complexity of the DP algorithm is roughly a fourth-order polynomial in the number of temporal signal snapshots. A stochastic modeling and particle filtering-based algorithm has been proposed by using a state-space model to solve nonlinear and non-Gaussian signals [12]. Ko et al. [13] proposed a blind maximum likelihood- (ML-) based iterative algorithm for frequency estimation and synchronization using a two-hop model; however, it yielded more than one solution, raising the problem of convergence to the solution that is associated with the hopping frequency. The authors [8] also pointed out that whether the approach of Ko et al. [13] can guarantee identifiability for the frequency estimation. Additionally, in the ML-based estimation approach, if the transition time in the processing data block between two hopping frequencies is close to the boundary value, then the data block is in an unbalance situation of sampled signals in the frequency components. In this scenario, the performance of one estimation of frequency is severely degraded.

This investigation presents a blind frequency estimation and timing synchronization algorithm. The approach is resistant to the aforementioned problem of unbalance. It reduces the computational load by using a proposed iterative method compared to a maximum likelihood-based greedy search method.

The rest of this paper is organized as follows. Section 2 introduces signal model and the problem formulation. In Section 3, the proposed algorithm for estimating frequencies and transition time is derived. Section 4 presents the computational complexity. Section 5 presents the computer-simulated results. Finally, Section 6 draws conclusions.

2. Signal Model and Problem Formulation

In this section, the signal model of FHSS is analyzed and the mathematical form of a likelihood function is derived. The two-hop signal model for frequency hopping can be expressed as

$$s(n) = \begin{cases} e^{j\omega_1 n T_s}, & n = 0, \dots, M-1, \\ e^{j\omega_2 (n-M) T_s}, & n = M, \dots, 2M-1, \end{cases} \quad (1)$$

where ω_1 and ω_2 are the hopping frequencies, T_s is the sampling period, and MT_s is the hopping period [13].

The received signal in an interval of MT_s can be written as

$$x(n) = \begin{cases} a_1 e^{j\omega_1 (M-K-1+n) T_s} + v(n), & n = 1, \dots, K, \\ a_2 e^{j\omega_2 (n-K-1) T_s} + v(n), & n = K+1, \dots, M, \end{cases} \quad (2)$$

where a_q , $q = 1, 2$ represent the channel gains of the q th hop through the transmitting path and $v(n)$ is an added white Gaussian noise (AWGN) with zero mean and variance σ^2 . The problem of determining K in (2) is thus equivalent to solving the timing synchronization problem.

Rewriting the received signal in vector form yields

$$\mathbf{x} = \mathbf{s} + \mathbf{v}, \quad (3)$$

where

$$\begin{aligned} \mathbf{x} &= [x(1), \dots, x(M)]^T, \\ \mathbf{s} &= [a_1 e^{j\omega_1 (M-K) T_s}, \dots, a_1 e^{j\omega_1 (M-1) T_s}, \\ &\quad a_2, \dots, a_2 e^{j\omega_2 (M-K-1) T_s}]^T, \\ \mathbf{v} &= [v(1), \dots, v(M)]^T. \end{aligned} \quad (4)$$

To simplify the analysis, \mathbf{x} can be partitioned into two components that correspond to individual hops as follows:

$$\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} a_1 \mathbf{s}_1 \\ a_2 \mathbf{s}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{bmatrix}, \quad (5)$$

where

$$\begin{aligned} \mathbf{x}_1 &= [x(1), \dots, x(K)]^T, \\ \mathbf{x}_2 &= [x(K+1), \dots, x(M)]^T, \\ \mathbf{s}_1 &= [e^{j\omega_1 (M-K) T_s}, \dots, e^{j\omega_1 (M-1) T_s}]^T, \\ &= e^{j\omega_1 (M-K) T_s} [1, \dots, e^{j\omega_2 (K-1) T_s}]^T \\ \mathbf{s}_2 &= [1, \dots, e^{j\omega_2 (M-K-1) T_s}]^T, \\ \mathbf{v}_1 &= [v(1), \dots, v(K)]^T, \\ \mathbf{v}_2 &= [v(K+1), \dots, v(M)]^T. \end{aligned} \quad (6)$$

The likelihood function of the received signal is

$$\begin{aligned} L(a_1, \omega_1, a_2, \omega_2, K) &= \frac{1}{(\sqrt{2\pi}\sigma)^M} e^{-(1/2\sigma^2)\|\mathbf{x}-\mathbf{s}\|^2} \\ &= \frac{1}{(\sqrt{2\pi}\sigma)^M} e^{-(1/2\sigma^2)(\|\mathbf{x}_1 - a_1 \mathbf{s}_1\|^2 + \|\mathbf{x}_2 - a_2 \mathbf{s}_2\|^2)}. \end{aligned} \quad (7)$$

The parameters $(a_1, \omega_1, a_2, \omega_2, K)$ can be estimated by maximizing (7), which is equivalent to minimizing the objective function

$$\begin{aligned} \varphi(a_1, \omega_1, a_2, \omega_2, K) &= \|\mathbf{x}_1 - a_1 \mathbf{s}_1\|^2 + \|\mathbf{x}_2 - a_2 \mathbf{s}_2\|^2 \\ &= \varphi_1(a_1, \omega_1, K) + \varphi_2(a_2, \omega_2, M-K), \end{aligned} \quad (8)$$

where

$$\varphi_1(a_1, \omega_1, K) = \|\mathbf{x}_1 - a_1 \mathbf{s}_1\|^2, \quad (9)$$

$$\varphi_2(a_2, \omega_2, M-K) = \|\mathbf{x}_2 - a_2 \mathbf{s}_2\|^2. \quad (10)$$

Since φ_1 and φ_2 of (8) are positive, minimizing (8) after some manipulation yields the estimated frequency $\hat{\omega}_1$:

$$\hat{\omega}_1 = \arg_{\omega_1} \left\{ \max \left(\frac{\|\hat{\mathbf{s}}_1^H \mathbf{x}_1\|^2}{K} \right) \right\}, \quad (11)$$

where

$$\hat{\mathbf{s}}_1 = e^{j\hat{\omega}_1 (M-K) T_s} [1, \dots, e^{j\hat{\omega}_1 (K-1) T_s}]^T. \quad (12)$$

Similarly, minimizing (10) yields $\hat{\omega}_2$:

$$\hat{\omega}_2 = \arg_{\omega_2} \left\{ \max \left(\frac{\|\hat{\mathbf{s}}_2^H \mathbf{x}_2\|^2}{(M-K)} \right) \right\}, \quad (13)$$

where

$$\hat{\mathbf{s}}_2 = [1, \dots, e^{j\hat{\omega}_2 (M-K-1) T_s}]^T. \quad (14)$$

Equations (11) and (13) indicate that frequencies $\hat{\omega}_1$ and $\hat{\omega}_2$ can be estimated by maximizing $\|\hat{\mathbf{s}}_1^H \mathbf{x}_1\|^2/K$ and $\|\hat{\mathbf{s}}_2^H \mathbf{x}_2\|^2/(M-K)$, respectively. The maximization of the two functions can be regarded as finding $\hat{\omega}_1$ and $\hat{\omega}_2$ that maximize the values of $\hat{\mathbf{s}}_1, \hat{\mathbf{s}}_2$ projected into the received signals \mathbf{x}_1 and \mathbf{x}_2 .

Finally, given the derived equations (11) and (13), the objective function of minimizing (8) is equivalent to the maximization of the objective function

$$\bar{\varphi}(\hat{\omega}_1, \hat{\omega}_2) = \bar{\varphi}_1 + \bar{\varphi}_2, \quad (15)$$

where

$$\begin{aligned} \bar{\varphi}_1 &= \frac{\|\hat{\mathbf{s}}_1^H \mathbf{x}_1\|^2}{K}, \\ \bar{\varphi}_2 &= \frac{\|\hat{\mathbf{s}}_2^H \mathbf{x}_2\|^2}{(M-K)}. \end{aligned} \quad (16)$$

Since the transition time K is not known in advance, every possible $K = 1, \dots, M$ and the estimates of the two frequencies $\hat{\omega}_1, \hat{\omega}_2$ should be tried in ML-based greedy search approaches. The ML estimation of K can be performed using

$$\hat{K} = \arg \max_K [\bar{\varphi}(\hat{\omega}_1, \hat{\omega}_2)]. \quad (17)$$

3. Proposed Estimation Algorithm Based on Maximum Likelihood Principle

The blind estimation scheme is developed in this section. The processing of the proposed scheme is divided into the synchronization phase and the tracking phase for parameter estimation. The details are as follows.

3.1. Synchronization Phase. Based on the analysis in Section 2, solving (8) is clearly a multivariable estimation problem (K, ω_1 and ω_2). To estimate frequencies ω_1 and ω_2 accurately, K must be estimated correctly. On the other hand, the accurate estimation of K depends on sufficiently accurate estimates of frequencies ω_1 and ω_2 . The most fundamental approach for solving this problem is a maximum likelihood-based greedy search approach, in which estimates are made by scanning the frequency and the transition time simultaneously, and finding the values of the frequency and time that maximize (8). However, the greedy search approach has an extremely large computational complexity although it may yield the optimal solution in some sense.

To deliver competitive performance but reduce the computational load, the proposed algorithm is developed as an iterative approach by modifying the concept of the alternative projection algorithm that was proposed by Ziskind and Wax [14]. Essentially, the approach converts a multivariable problem into a single variable problem and thereby reduces the computational load. The algorithm differs from that of Ziskind and Wax [14] and must be adapted to the FHSS problem. The proposed approach does not depend the complex matrix inverse calculation but

simply applies a basic vector computation (inner product), which further reduces the computational complexity.

In iterating the proposed scheme, the maximization is only conducted on one variable while the other variables are held constant. For instance, K may be fixed first and the ω_1 and ω_2 that maximize the objective function are computed; after ω_1 and ω_2 have been estimated, ω_1 and ω_2 are fixed and then K is estimated. After many iterations, the estimates of $\hat{\omega}_1, \hat{\omega}_2$, and \hat{K} can be obtained.

Since the transition time K is unknown in advance, an initial value of K , denoted as $\hat{K}^{(0)}$, is set, where the number in the superscript bracket stands for the iteration number and the overscript with the sign of a hat stands for "estimated value." For example, $\hat{K}^{(0)}$ denotes the estimated value of K in the initialization. Since K is unknown in advance, the initial estimate K can be selected based on the minimization of the initial estimation mean squared error in a statistical sense. K is a random variable with a uniform distribution. The expectation of the random variable is $\mu = E[K] = (M+1)/2$. Therefore, the initial estimate μ minimizes the mean squared error. The initial value $\hat{K}^{(0)}$ can be set to $\lfloor (M+1)/2 \rfloor$.

With the initial value of K known, the estimated values of $\omega_1^{(1)}$ and $\omega_2^{(1)}$ can be obtained by the maximization as

$$\begin{aligned} \hat{\omega}_1^{(1)} &= \max_{\omega_1} \frac{\|\mathbf{s}_1^H(\hat{K}^{(0)}, \omega_1) \mathbf{x}_1\|^2}{\hat{K}^{(0)}}, \\ \hat{\omega}_2^{(1)} &= \max_{\omega_2} \frac{\|\mathbf{s}_2^H(M - \hat{K}^{(0)}, \omega_2) \mathbf{x}_2\|^2}{(M - \hat{K}^{(0)})}. \end{aligned} \quad (18)$$

Next, the estimated frequencies $\hat{\omega}_1^{(1)}$ and $\hat{\omega}_2^{(1)}$ are fixed, and $\hat{K}^{(1)}$ is calculated by

$$\hat{K}^{(1)} = \arg \max_K [\bar{\varphi}^{(1)}(K^{(1)}, \hat{\omega}_1^{(1)}, \hat{\omega}_2^{(1)})], \quad (19)$$

where

$$\begin{aligned} \bar{\varphi}^{(1)}(K^{(1)}, \hat{\omega}_1^{(1)}, \hat{\omega}_2^{(1)}) &= \bar{\varphi}_1^{(1)}(K^{(1)}, \hat{\omega}_1^{(1)}) + \bar{\varphi}_2^{(1)}(M - K^{(1)}, \hat{\omega}_2^{(1)}), \end{aligned} \quad (20)$$

$$\bar{\varphi}_1^{(1)}(K^{(1)}, \hat{\omega}_1^{(1)}) = \frac{\|\mathbf{s}_1^H(K^{(1)}, \hat{\omega}_1^{(1)}) \mathbf{x}_1\|^2}{K^{(1)}}, \quad (21)$$

$$\bar{\varphi}_2^{(1)}(M - K^{(1)}, \hat{\omega}_2^{(1)}) = \frac{\|\mathbf{s}_2^H(M - K^{(1)}, \hat{\omega}_2^{(1)}) \mathbf{x}_2\|^2}{(M - K^{(1)})}.$$

Based on the above development, $\hat{K}, \hat{\omega}_1$, and $\hat{\omega}_2$ can be estimated by fixing the values of the other parameters in each iteration. Notably, the value given by (20) obtained by utilizing the proposed processing procedure is monotonically increasing during the iteration. After a few iterations, it converges to a local maximum, and the local maximum may or may not be the global maximum.

3.2. Robust Estimation in the Synchronization Phase. Although the estimates of K, ω_1 , and ω_2 can be obtained

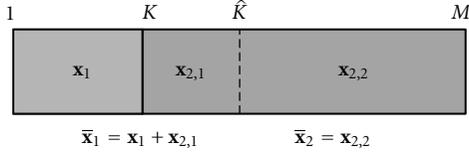


FIGURE 1: Situation with real K and estimated \hat{K} .

using the proposed scheme or other ML-based schemes, such as the exhaustive search method, if the transition time of the received signal is close to the boundary of the sampling data block, then one of the frequency estimation errors would be large, because the samples used in the estimation of one frequency are small. To eliminate this problem and improve performance, an attempt can be made to shift the sampling data block for processing in each iteration. This action is equivalent to adjusting the transition time K by shifting, and this value is expected to be close to $M/2$ as in a balanced situation. In each iteration, the sampling position of the data block is shifted by

$$\Delta K^{(i)} = K^{(i)} - \frac{M}{2}. \quad (22)$$

When the difference between the initial values of \hat{K} and true K is large, the estimation may be erroneous. For example, only one estimated frequency would be obtained because the duration of the signal with a single frequency component would dominate the whole sample block. The problem is remedied by making the following proposed modification. Since (11) and (13) are derived by assuming that the true K is known, both equations can be treated as two separated functions of only one frequency signal in each subblock such that ω_1, ω_2 can be estimated using (11) and (13), respectively. However, in practice, the transition time K in (11) and (13) is obtained from the estimation of \hat{K} . If $\hat{K} \neq K$, then one of the two subblocks contains both frequencies ω_1 and ω_2 and the other subblock contains only one frequency component. With reference to Figure 1, if $\hat{K} > K$, (11) and (13) become

$$\hat{\omega}_1 = \arg \left\{ \max_{\omega_1} \left(\frac{\|\hat{s}_1^H \bar{\mathbf{x}}_1\|^2}{\hat{K}} \right) \right\}, \quad (23)$$

$$\hat{\omega}_2 = \arg \left\{ \max_{\omega_2} \left(\frac{\|\hat{s}_2^H \bar{\mathbf{x}}_2\|^2}{(M - \hat{K})} \right) \right\}, \quad (24)$$

where

$$\begin{aligned} \bar{\mathbf{x}}_1 &= [\mathbf{x}_1, \mathbf{x}_{2,1}]^T, \\ \mathbf{x}_{2,1} &= [x(K+1), \dots, x(\hat{K})]^T, \\ \bar{\mathbf{x}}_2 &= [\mathbf{x}_{2,2}]^T, \\ \mathbf{x}_{2,2} &= [x(\hat{K}+1), \dots, x(M)]^T, \\ \mathbf{x}_2 &= [\mathbf{x}_{2,1}, \mathbf{x}_{2,2}]^T, \end{aligned} \quad (25)$$

Since $\bar{\mathbf{x}}_1$ in (23) contains both frequencies ω_1 and ω_2 , the expression in the $\max_{\omega_1} \{ \cdot \}$ operation of (23) becomes

$$\begin{aligned} & \max_{\omega_1} \left[\frac{\|\hat{s}_1^H \bar{\mathbf{x}}_1\|^2}{\hat{K}} \right] \\ &= \max_{\omega_1} \left[\frac{\|\hat{s}_1^H [\mathbf{x}_1, \mathbf{x}_{2,1}]^T\|^2}{\hat{K}} \right] \\ &= \max_{\omega_1} \left[\frac{\left\| \sum_{k_1=1}^K \hat{s}_1^H(k_1)x(k_1) + \sum_{k_2=K+1}^{\hat{K}} \hat{s}_1^H(k_2)x(k_2) \right\|^2}{\hat{K}} \right] \\ &= \max_{\omega_1} \left[\frac{\left\| \sum_{k_1=1}^K \hat{s}_1^H(k_1)(a_1s_1(k_1) + v_1(k_1)) \right. \right. \\ & \quad \left. \left. + \frac{\sum_{k_2=K+1}^{\hat{K}} \hat{s}_1^H(k_2)(a_2s_2(k_2) + v_2(k_2))}{\hat{K}} \right\|^2}{\hat{K}} \right]. \end{aligned} \quad (26)$$

According to (26), when $\hat{K} \neq K$, the expression contains the desired signal \mathbf{x}_1 , noise, and an interference portion $\mathbf{x}_{2,1}$. Therefore, if $\hat{\omega}_1$ is estimated using (26), then the estimation frequency error would depend on the difference $|\hat{K} - K|$ and the noise. Additionally, since (26) contains both ω_1 and ω_2 , two peaks that correspond to ω_1 and ω_2 are identified in frequency scanning. Therefore, the initial value \hat{K} would affect the performance of the estimation. To solve this problem, \hat{K} can be adjusted close to K . The task is achieved by performing the following operation.

If $\hat{K} > 2K$ (or $M - \hat{K} > 2(M - K)$), then the estimate of ω_1 from the signal subblock may become ω_2 (or ω_2 to be estimated erroneously as ω_1). The results of the estimation would have approximately the same value: $\hat{\omega}_1 \approx \hat{\omega}_2$, (if the channel gains of the two frequencies are assumed to be approximately the same as for regular applied systems). To solve the problem of $\hat{\omega}_1 \approx \hat{\omega}_2$, a method is proposed in which the value $K^{(0)}$ is adjusted close to $M/2$ by shifting forward and backward the received signal \mathbf{x} with $\Delta K^{(0)} = M/4$, when the frequency estimation condition $\hat{\omega}_1 \approx \hat{\omega}_2$ is met. (The reason for choosing $\Delta K^{(0)} = M/4$ will be explained below.)

Consider the situation $\hat{K} > 2K$. Since the goal is to obtain $\hat{K} = M/2$ for a balanced situation, $\hat{K} = M/2$ is substituted into $\hat{K} > 2K$ yielding the inequality $K < M/4$. Hence, the situation of $\hat{\omega}_1 \approx \hat{\omega}_2$ may occur if $K < M/4$. The transition time K of the received signal in a selected sample block is a random variable. When $0 \leq K \leq M/4$, the received signal block should be adjusted backward by $M/4$ samples. Following this adjustment, the transition time may fall in the range of $M/4 \leq K \leq M/2$; a similar adjustment can be applied when $M - \hat{K} > 2(M - K)$. Substituting $\hat{K} = M/2$ yields $3M/4 \leq K \leq M$. The received signal block should

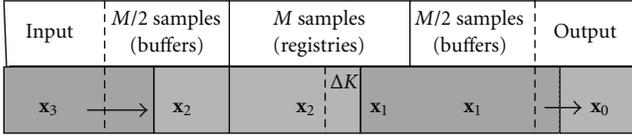


FIGURE 2: Received signal samples in registers.

be adjusted forward by $M/4$ samples, and the transition time may fall in the range $M/2 \leq K \leq 3M/4$.

Finally, when $\hat{\omega}_1 \approx \hat{\omega}_2$, $\hat{K} > 2K$ or $M - \hat{K} > 2(M - K)$ may occur. The forward and backward adjustments of the received signal block with $\Delta K^{(0)} = M/4$ yield the two shifted versions of \mathbf{x} :

$$\begin{aligned} \mathbf{x}_f^{(0)} &= [x(1 - \Delta K^{(0)}), \dots, x(M - \Delta K^{(0)})]^T, \\ \mathbf{x}_b^{(0)} &= [x(1 + \Delta K^{(0)}), \dots, x(M + \Delta K^{(0)})]^T. \end{aligned} \quad (27)$$

The block that has two peaks with similar power in the frequency domain may be chosen to determine whether $K^{(0)}$ is close to $M/2$ for the sample block. The decision rule is expressed as

$$\mathbf{x}_{\text{adj}}^{(0)} = P\mathbf{x}_{b_1} - P\mathbf{x}_{b_2} \underset{\mathbf{x}_b^{(0)}}{\overset{\mathbf{x}_f^{(0)}}{\geq}} P\mathbf{x}_{f_1} - P\mathbf{x}_{f_2}, \quad (28)$$

where $P\mathbf{x}_{b_1}$, $P\mathbf{x}_{b_2}$, $P\mathbf{x}_{f_1}$, and $P\mathbf{x}_{f_2}$ are the first and the second largest power values in blocks \mathbf{x}_b and \mathbf{x}_f , respectively. The estimates can be made by performing the DFT operations.

3.3. Tracking Phase. Once the timing of the received signal is determined, the frequency can be estimated by receiving every upcoming M sample. However, Owing to timing jitter and possible hostile communication scenarios, it is necessary to track the timing and frequencies of upcoming data samples. The following processing step is proposed. Figure 2 presents the process in which the received signal is fed into the registers. The registries contain M samples and two buffers, each associated with $M/2$ samples, which are used to adjust $\Delta K^{(i)}$.

Since the parameters $\hat{\omega}_1$, $\hat{\omega}_2$, and \hat{K} in the preceding block are obtained, $\hat{\omega}_2$ in the previous block becomes $\hat{\omega}_1$ in the upcoming data block. Hence, the estimated $\hat{\omega}_2$ and \hat{K} are adopted in the following estimation operation:

$$\mathbf{x}^{(0)} = [x(1), \dots, x(M)]^T, \quad (29)$$

$$\hat{K}^{(0)} = \frac{M}{2}, \quad (30)$$

$$\hat{\omega}_2^{(1)} = \arg \max_{\omega_2} \left\{ \max \left(\frac{\|\hat{\mathbf{s}}_2^H \mathbf{x}_2\|^2}{(M - \hat{K}^{(0)})} \right) \right\}, \quad (31)$$

$$\hat{K}^{(1)} = \arg \max_K [\bar{\varphi}^{(1)}(K^{(1)}, \hat{\omega}_1, \hat{\omega}_2^{(1)})], \quad (32)$$

$$\Delta K^{(1)} = \hat{K}^{(1)} - \hat{K}^{(0)} = \hat{K}^{(1)} - \frac{M}{2}, \quad (33)$$

$$\mathbf{x}^{(1)} = [x(1 + \Delta K), \dots, x(M + \Delta K)]^T. \quad (34)$$

Since \hat{K} is adjusted close to $M/2$ in the synchronization phase, in this phase, the received signal \mathbf{x} has a balanced block, and the iteration number $i = 1$ can be set. The previous estimate of $\hat{\omega}_2$ is used to estimate $\hat{\omega}_1$ in the upcoming block. Therefore, the estimation of $\hat{\omega}_1$ is eliminated. Notably, in (32), only $M/2$ samples are utilized to estimate the frequency to reduce the computational load and save time for the subsequent system operation. (In the hostile communication environment, e.g., a jammer/interfering operation may follow the estimation task.) However, if accuracy of the frequency estimation is paramount, then all M samples may be adopted to perform the frequency estimation. It would rely on the type of the application, and using whole M or $M/2$ sample for estimation is a trade-off between the computational complexity and the estimation accuracy.

Algorithm Summary. The steps in the proposed algorithm are summarized as follows.

Step 1 (synchronization phase). (1) Receive $2M$ signal samples $[x(-(1/2)M), \dots, x(1/2M)]^T$ and input the data to the registers and the buffers.

(2) Perform the preprocessing procedure to estimate frequencies and K .

(A) Set $\hat{K}^{(0)} = M/2$ and estimate the frequencies that maximize

$$\begin{aligned} \hat{\omega}_1^{(1)} &= \max_{\omega_1} \left(\frac{\|\mathbf{s}_1^H(\hat{K}^{(0)}, \omega_1) \mathbf{x}_1(\hat{K}^{(0)})\|^2}{\hat{K}^{(0)}} \right), \\ \hat{\omega}_2^{(1)} &= \max_{\omega_2} \left(\frac{\|\mathbf{s}_2^H(M - \hat{K}^{(0)}, \omega_2) \mathbf{x}_2(M - \hat{K}^{(0)})\|^2}{(M - \hat{K}^{(0)})} \right). \end{aligned} \quad (35)$$

(B) If $\hat{\omega}_1 \approx \hat{\omega}_2$, then shift $\mathbf{x}^{(0)}$ forward and backward, yielding two blocks,

$$\begin{aligned} \mathbf{x}_f^{(0)} &= [x(1 - \Delta K^{(0)}), \dots, x(M - \Delta K^{(0)})]^T, \\ \mathbf{x}_b^{(0)} &= [x(1 + \Delta K^{(0)}), \dots, x(M + \Delta K^{(0)})]^T, \end{aligned} \quad (36)$$

where $\Delta K^{(0)} = M/4$. Then, select one of the two blocks by applying the following decision rule:

$$\mathbf{x}_{\text{adj}}^{(0)} = P\mathbf{x}_{b_1} - P\mathbf{x}_{b_2} \underset{\mathbf{x}_b^{(0)}}{\overset{\mathbf{x}_f^{(0)}}{\geq}} P\mathbf{x}_{f_1} - P\mathbf{x}_{f_2}. \quad (37)$$

Go to (A).

(C) Estimate K using

$$\hat{K}^{(1)} = \arg \max_{K^{(1)}} [\bar{\varphi}^{(1)}(K^{(1)}, \hat{\omega}_1^{(1)}, \hat{\omega}_2^{(1)})], \quad (38)$$

where

$$\begin{aligned} \bar{\varphi}^{(1)}(K^{(1)}, \hat{\omega}_1^{(1)}, \hat{\omega}_2^{(1)}) \\ = \bar{\varphi}_1^{(1)}(K^{(1)}, \hat{\omega}_1^{(1)}) + \bar{\varphi}_2^{(1)}(M - K^{(1)}, \hat{\omega}_2^{(1)}). \end{aligned} \quad (39)$$

(D) Shift by $\Delta K = M/2 - \hat{K}^{(1)}$

$$\mathbf{x}^{(1)} = [x(1 + \Delta K), \dots, x(M + \Delta K)]^T. \quad (40)$$

(3) For $i = 2 \sim I$, perform iterations to estimate frequencies and K

(A) Set $\hat{K}^{(i-1)} = M/2$ and estimate the frequencies that maximize

$$\begin{aligned} \hat{\omega}_1^{(i)} &= \max_{\omega_1} \left(\frac{\|\mathbf{s}_1^H(\hat{K}^{(i-1)}, \omega_1) \mathbf{x}_1(\hat{K}^{(i-1)})\|^2}{\hat{K}^{(i-1)}} \right), \\ \hat{\omega}_2^{(i)} &= \max_{\omega_2} \left(\frac{\|\mathbf{s}_2^H(M - \hat{K}^{(i-1)}, \omega_2) \mathbf{x}_2(M - \hat{K}^{(i-1)})\|^2}{(M - \hat{K}^{(i-1)})} \right). \end{aligned} \quad (41)$$

(B) Estimate K using

$$\hat{K}^{(i)} = \arg \max_K [\bar{\varphi}^{(i)}(K^{(i)}, \hat{\omega}_1^{(i)}, \hat{\omega}_2^{(i)})], \quad (42)$$

where

$$\bar{\varphi}^{(i)}(K^{(i)}, \hat{\omega}_1^{(i)}, \hat{\omega}_2^{(i)}) = \bar{\varphi}_1^{(i)}(K^{(i)}, \hat{\omega}_1^{(i)}) + \bar{\varphi}_2^{(i)}(M - K^{(i)}, \hat{\omega}_2^{(i)}). \quad (43)$$

(C) Shift by $\Delta K = M/2 - \hat{K}^{(i)}$

$$\mathbf{x}^{(i)} = [x(1 + \Delta K), \dots, x(M + \Delta K)]^T. \quad (44)$$

Step 2 (tracking phase). (1) Input the next M samples to the registers.

(2) Use the estimated parameter $\hat{K}^{(0)} = \hat{K}$ and $\hat{\omega}_1$ (which is set to the estimated $\hat{\omega}_2$ of the previous block) to compute the following.

(A) Find the frequency estimate that maximizes

$$\hat{\omega}_2^{(1)} = \max_{\omega_2} \left(\frac{\|\mathbf{s}_2^H(M - \hat{K}^{(0)}, \omega_2) \mathbf{x}_2(M - \hat{K}^{(0)})\|^2}{(M - \hat{K}^{(0)})} \right). \quad (45)$$

(B) Next, fix the frequency estimates and find K that maximizes

$$\hat{K}^{(1)} = \arg \max_K [\bar{\varphi}^{(1)}(K^{(1)}, \hat{\omega}_1, \hat{\omega}_2^{(1)})]. \quad (46)$$

(C) Shift by $\Delta K = M/2 - \hat{K}^{(1)}$ to obtain

$$\mathbf{x}^{(1)} = [x(1 + \Delta K), \dots, x(M + \Delta K)]^T. \quad (47)$$

4. Analysis of Computational Complexity

The Big-Oh notation is a well-accepted approach for analyzing the computational complexities of algorithms and is adopted. The computational complexity is analyzed in detail as follows. Let N be the number of frequency scanning points, which is related to the frequency scanning resolution and is typically much larger than M .

In the ML greedy search scheme, every possible $K = 1, \dots, M$ and the estimates of the two frequencies $\hat{\omega}_1, \hat{\omega}_2$ should be tried. For each possible transition time that is used to evaluate $\max_K [\bar{\varphi}(\hat{\omega}_1, \hat{\omega}_2)]$ in (17), the computational complexity of the multiplication operations is $O(N\{(M - K) + K\}) = O(MN)$, where N is the number of points and the two subvectors of sizes $(M - K)$ and K are involved in the operation. The computational complexity of the addition operations is $O(N \times N) = O(N^2)$ by evaluating $\bar{\varphi}_1 + \bar{\varphi}_2$ with all paired combinations of the arguments $\hat{\omega}_1$ and $\hat{\omega}_2$. The selection of arguments in the maximum operation and the other operations has lower complexities and can be neglected when the Big-Oh notation is used. Accordingly, the total computational complexity of the multiplication operations is $O(M^2N)$ and that of the addition operations is $O(N^2M)$.

The proposed approach consists of the synchronization phase and the tracking phase. The multiplication and addition operations in the synchronization phase of the proposed approach are analyzed as follows. Referring to the algorithm summary, Step (A) has a computational complexity of $O(NM)$ and Step (B) has a computational complexity of $O(M^2)$. Step (B) of the preprocessing adjustment requires an extra $O(M \log M)$ complexity because of the two M -point FFT operations. The iterations stop after a fixed small number, which can be regarded as a constant I . Therefore, the total computational complexity is $O(M^2 + MN) \approx O(MN)$. The computational complexity is lower than that of the ML-based method. Since the transition time K of a received signal is random between 1 and M , wrong estimates can be made in the unbalanced situation. Thus, the adjustment scheme is proposed herein to prevent such a situation.

Similarly, the computational complexity of the scheme in the tracking phase is $O(MN)$. Although it has the same order of the computational complexity as the synchronization phase; however, K has been adjusted close to $M/2$, and only one frequency estimation task is performed. The computational complexity is further reduced, since we can estimate one frequency instead of two frequencies and search for K that is around $M/2$ instead of all possible M 's.

Real-time computing refers to hardware and software systems that are subject to a real-time constraint. The Big-Oh notation is well accepted for analyzing the computational complexity of algorithms. The true computing time of an algorithm varies with the processor, architecture, memory, and operating system used to execute it. Additionally, the computing power of processors continues to increase owing to progress in VLSI technology. When the computational complexity of an algorithm is polynomial time instead of exponential time, that algorithm is feasible for real-time

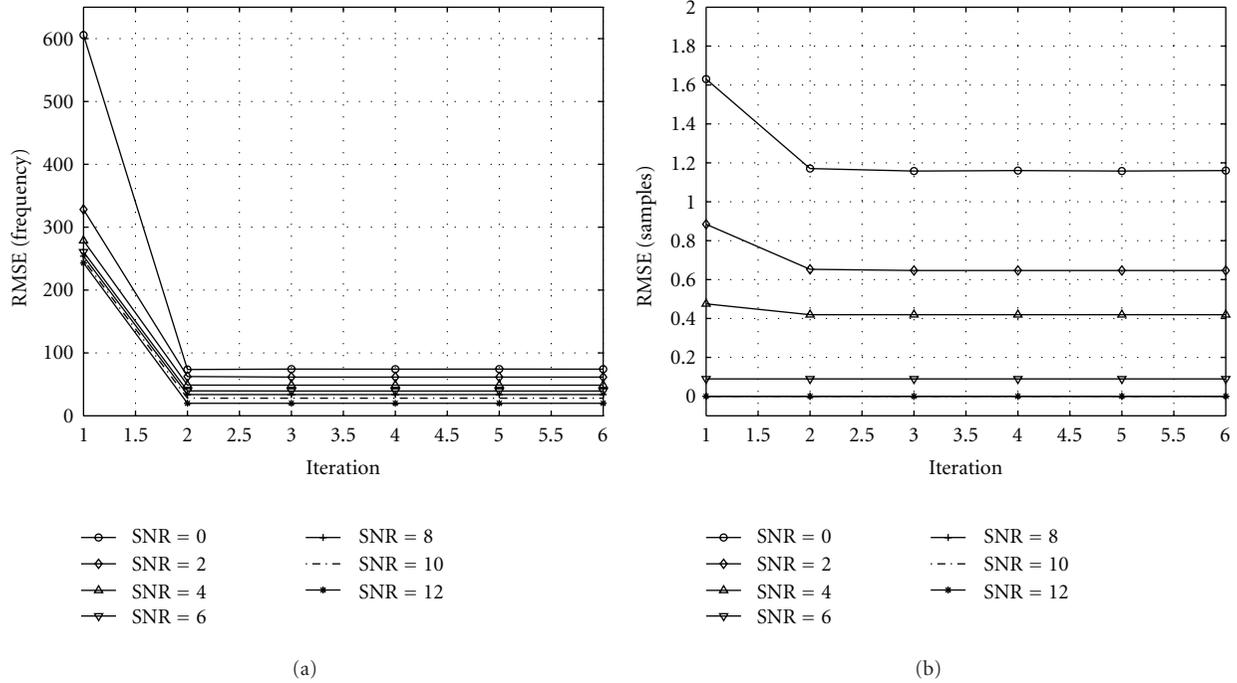


FIGURE 3: Number of iterations versus estimation error with $K = 120$. (a) Frequency estimation error. (b) K estimation error.

application. Therefore, the concept of computational complexity is adopted herein to judge whether an algorithm is a real-time algorithm. A similar evaluation approach can be found in the literature. The use of parallel processing can also reduce computing time. For example, if N processors are used, each may be assigned to evaluate the cost function of each frequency estimate, as in (18), which can be done separately and in parallel. As analyzed in this investigation, the computational complexity of the proposed scheme is $O(NM)$, which is polynomial time rather than exponential time. An algorithm with a complexity of $O(MN)$ can be treated as feasible for real-time application.

5. Simulations Results

In this section, numerous simulations are conducted to demonstrate the efficiency of the proposed scheme. The hopping frequencies are set to $[30k \ 13k \ 23k \ 40k]$ Hz, and the channel gains are set to $[0.75 + 0.02i \ 0.8 - 0.05i \ 0.83 + 0.10i \ 0.79 + 0.02i]$. The sampling frequency is $f_s = 100$ kHz and the hopping period is $M = 128$. The scanning frequency resolution is 50 Hz. To illustrate the convergence property of the proposed algorithm, Figure 3 presents the frequency error and the error in the estimation of the transition time K versus the number of iterations for $K = 120$. The simulation results indicate that it converges after about two iterations.

To evaluate performance, the transition time is set to three conditions, $K = 56$, $K = 120$, and random K , generated randomly between 1 and M with a uniform

distribution. The Cramer Rao lower bound (CRLB) for the balance block ($K = M/2$) is also provided for comparison. (It is derived in the appendix.)

Figure 4 compares the performance of the proposed method with that of the ML-based greedy search method. It reveals that the proposed algorithm has comparable performance to that of the ML algorithm for a balanced block with $K = 56$. Figure 4(a) shows that the proposed method performs better than the ML algorithm in estimating the f_1 frequency, because the proposed algorithm adjusts the sampling position to $M/2 = 64$, whereas the ML algorithm has a sample length of 56 for the estimation. The two algorithms perform similarly in this balanced block situation. However, the proposed algorithm outperforms the ML-based greedy search method in the following unbalanced block situation and in cases of random K .

Figure 5 presents the performance of the proposed scheme and the ML-based algorithm for a transition time close to the boundary of the data block, $K = 120$. As indicated in the result, because of the lack of samples, the ML-based algorithm yields large estimation errors, revealed in Figure 5(b), while the proposed algorithm performs stably as in the case of $K = 56$. In particular, in Figure 5(a), the estimation error of the ML-based method is lower than the CRLB because the data length $K = 120$ exceeds that, $K = 64$, used in the CRLB.

In practice, transition time is a random variable for a sampling data block. The transition time K is set randomly between 1 and M with a uniform distribution. As indicated in Figure 6, the performance of the proposed algorithm is similar to that with $K = 56$.

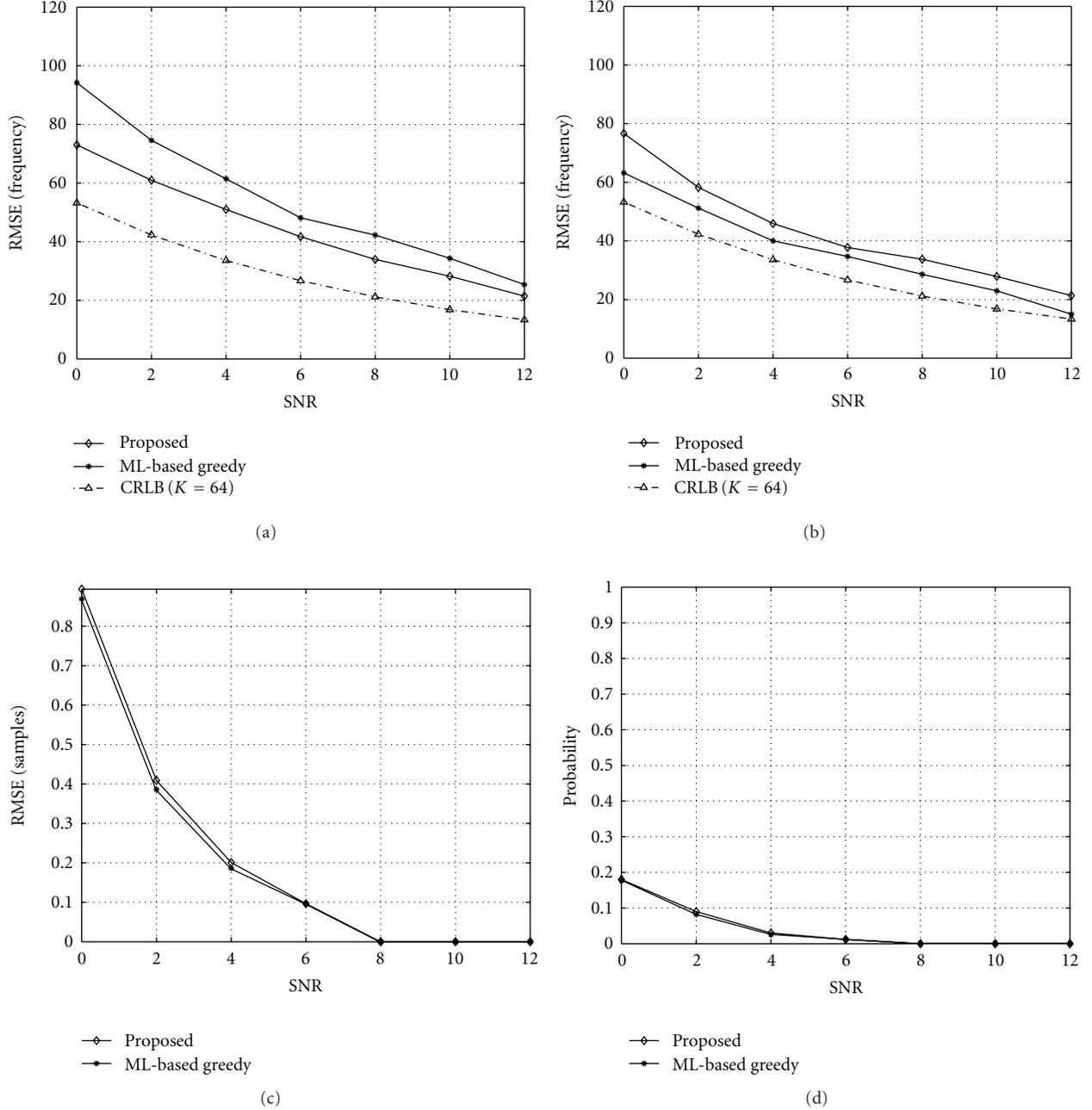
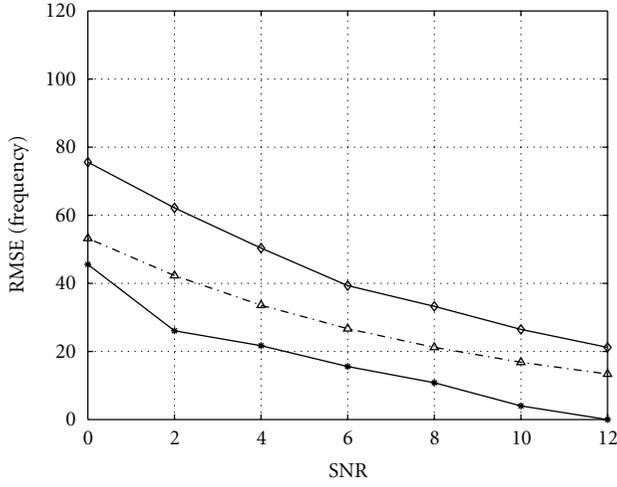


FIGURE 4: Root mean square estimation error versus SNR with $K = 56$. (a) RMSE of f_1 estimate, (b) RMSE of f_2 estimate, (c) RMSE of K estimate, and (d) error probability of K .

To demonstrate the tracking capability of the proposed processing scheme, the following simulation is performed with a sequence of ten frequencies. The hopping frequencies are [30k 13k 23k 40k 21k 45k 17k 31k 10k 37k 12k 29k] Hz. The complex channel gains for these frequencies are [0.79 + 0.02i, 0.80 - 0.05i, 0.83 + 0.10i, 0.65 + 0.55i, 0.78 + 0.04i, 0.77 - 0.32i, 0.73 - 0.34i, 0.81 + 0.03i, 0.68 - 0.42i, 0.75 - 0.22i, 0.81 + 0.20i, 0.76 + 0.33i]. The channel gains are generated using the expression $(0.8 + a_1) + (0.8 + a_2)i$, where a_1 and a_2 are normally distributed with zero mean and a variance of 0.05.

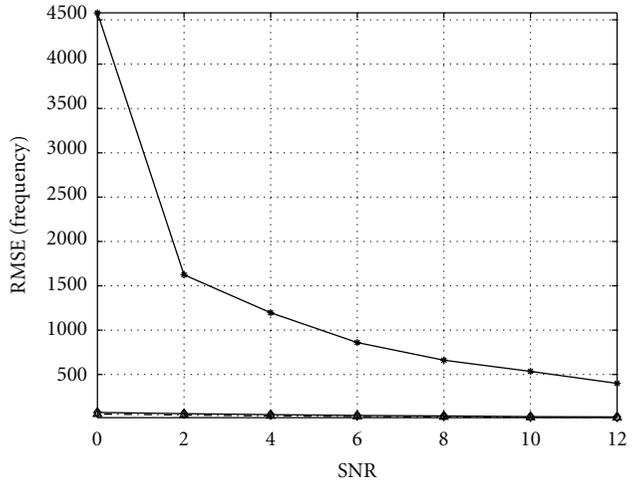
The initial transition time K is set randomly between 1 and M . As shown in Figure 7, the frequency estimation errors are less than 100 Hz over the range of the evaluation.

Figures 8(a) and 8(b) compare the experimental complexities. The CPU is Intel(R) Core(TM)2 CPU 6320 at 1.86 GHz, and the computing time of the proposed algorithm is much lower than that of the ML-based greedy search scheme. Notably, the computing time can be reduced by optimizing the program codes and designing special computer architectures for implementing the proposed scheme.



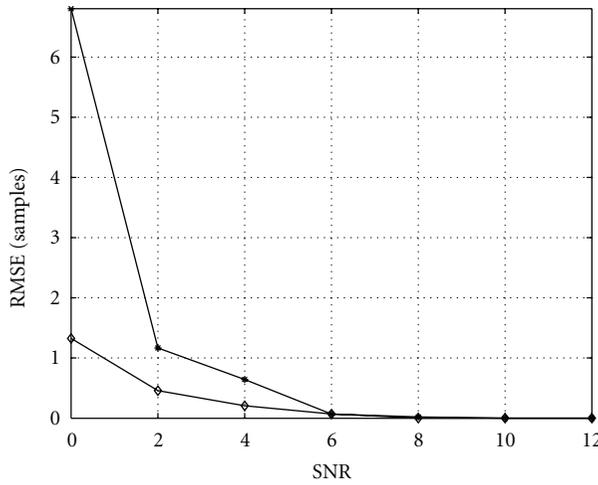
—◇— ML-based greedy
 —■— Proposed
 - -△- CRLB ($K = 64$)

(a)



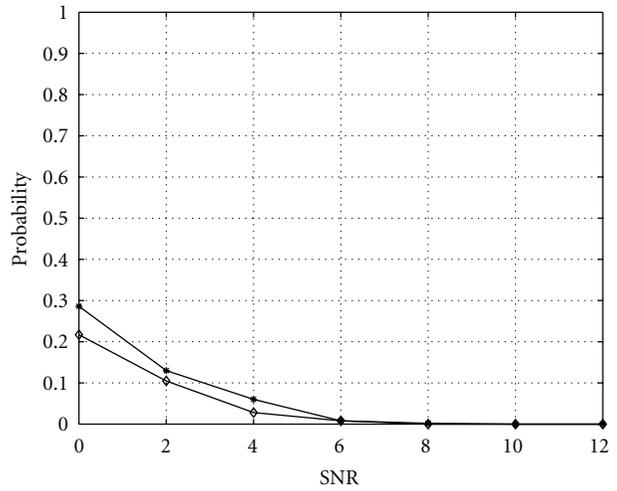
—◇— ML-based greedy
 —■— Proposed
 - -△- CRLB ($K = 64$)

(b)



—◇— Proposed
 —■— ML-based greedy

(c)



—◇— Proposed
 —■— ML-based greedy

(d)

FIGURE 5: Root mean square estimation error versus SNR with $K = 120$. (a) RMSE of f_1 estimate, (b) RMSE of f_2 estimate, (c) RMSE of K estimate, and (d) error probability of K .

6. Conclusions

A robust blind frequency and timing estimation algorithm is developed for frequency hopping systems. The proposed scheme has a lower computational load than the ML-based greedy search algorithm. The multivariable search problem is reduced to a single variable search problem. The algorithm does not require the simultaneous search of all times and frequencies, and its performance is comparable with that of the ML-based greedy search algorithm. The

problem of unbalanced situations (where K is close to the boundary) is solved using the proposed algorithm. The simulation results indicate that the performance is relatively independent of transition time K , whereas the pure ML-based algorithm fails to estimate the parameters. The proposed algorithm can be adapted for tracking. The tracking performance is also demonstrated by utilizing the estimated parameters of \hat{K} and $\hat{\omega}_1$ in the previous data block and the computational task of estimating $\hat{\omega}_1$ is omitted to reduce complexity.

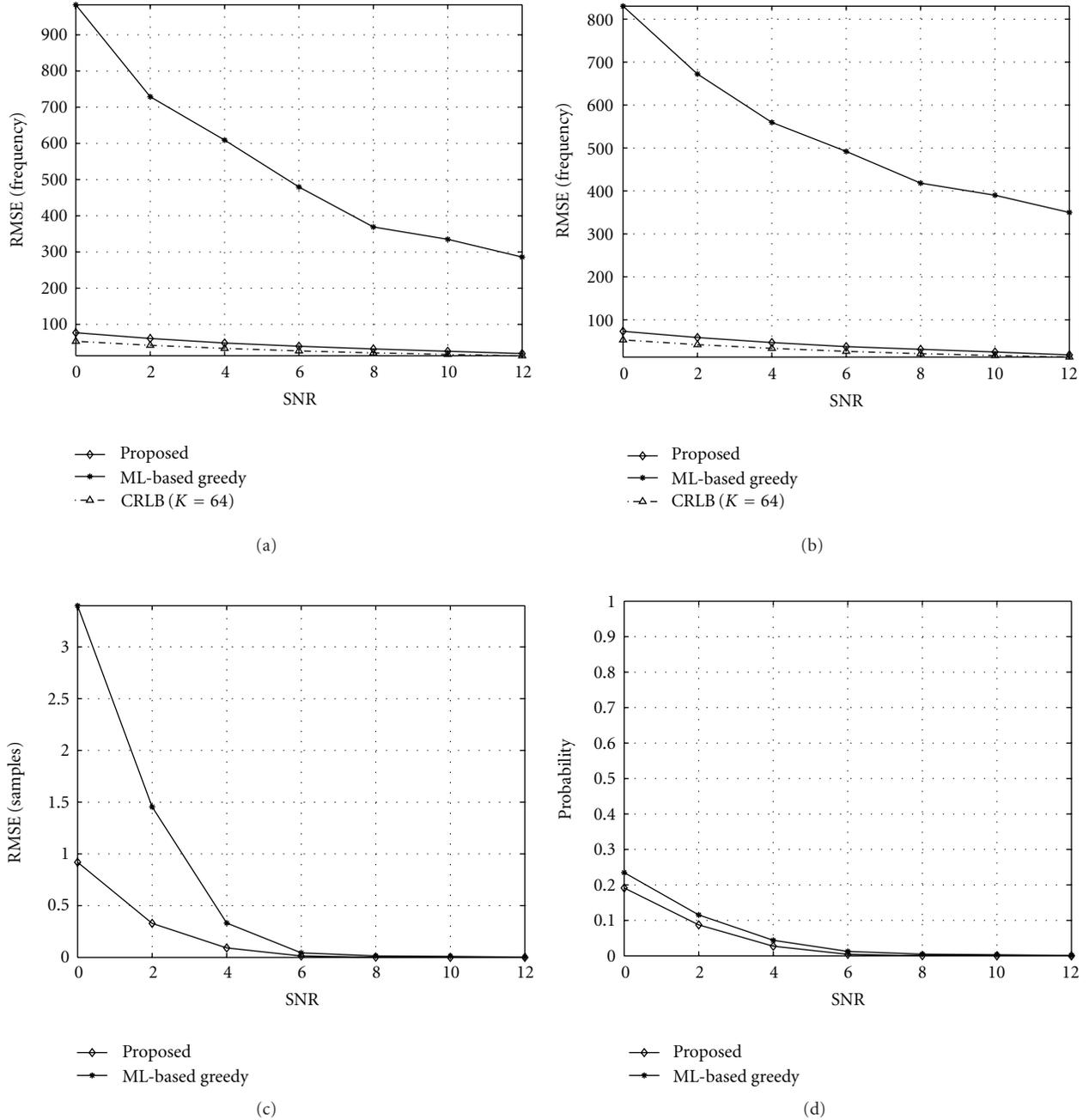


FIGURE 6: Root mean square estimation error versus SNR with $K = \text{random}$. (a) RMSE of f_1 estimate, (b) RMSE of f_2 estimate, (c) RMSE of K estimate, and (d) error probability of K .

Appendix

The likelihood function of the received signal is

$$\begin{aligned}
 L(a_1, \omega_1, a_2, \omega_2, K) &= \frac{1}{(\sqrt{2\pi}\sigma)^M} e^{-(1/2\sigma^2)\|\mathbf{x}-\mathbf{s}\|^2} \\
 &= \frac{1}{(\sqrt{2\pi}\sigma)^M} e^{-(1/2\sigma^2)(\|\mathbf{x}_1-a_1\mathbf{s}_1\|^2+\|\mathbf{x}_2-a_2\mathbf{s}_2\|^2)}.
 \end{aligned}
 \tag{A.1}$$

The random sizes of the subvectors that are associated with

the random parameter K make the derivation of the CRLBs for all estimates very complicated. Alternatively, if the goal is to put the received block into a balanced situation, $K = M/2$ is assumed. The M -sample block is divided into two $M/2$ -sample blocks. Hence, the likelihood function can be expressed as

$$\begin{aligned}
 L_1(a_1, \omega_1) &= \frac{1}{(\sqrt{2\pi}\sigma)^{M/2}} e^{-(1/2\sigma^2)\|\mathbf{x}_1-a_1\mathbf{s}_1\|^2}, \\
 L_2(a_2, \omega_2) &= \frac{1}{(\sqrt{2\pi}\sigma)^{M/2}} e^{-(1/2\sigma^2)\|\mathbf{x}_2-a_2\mathbf{s}_2\|^2}.
 \end{aligned}
 \tag{A.2}$$

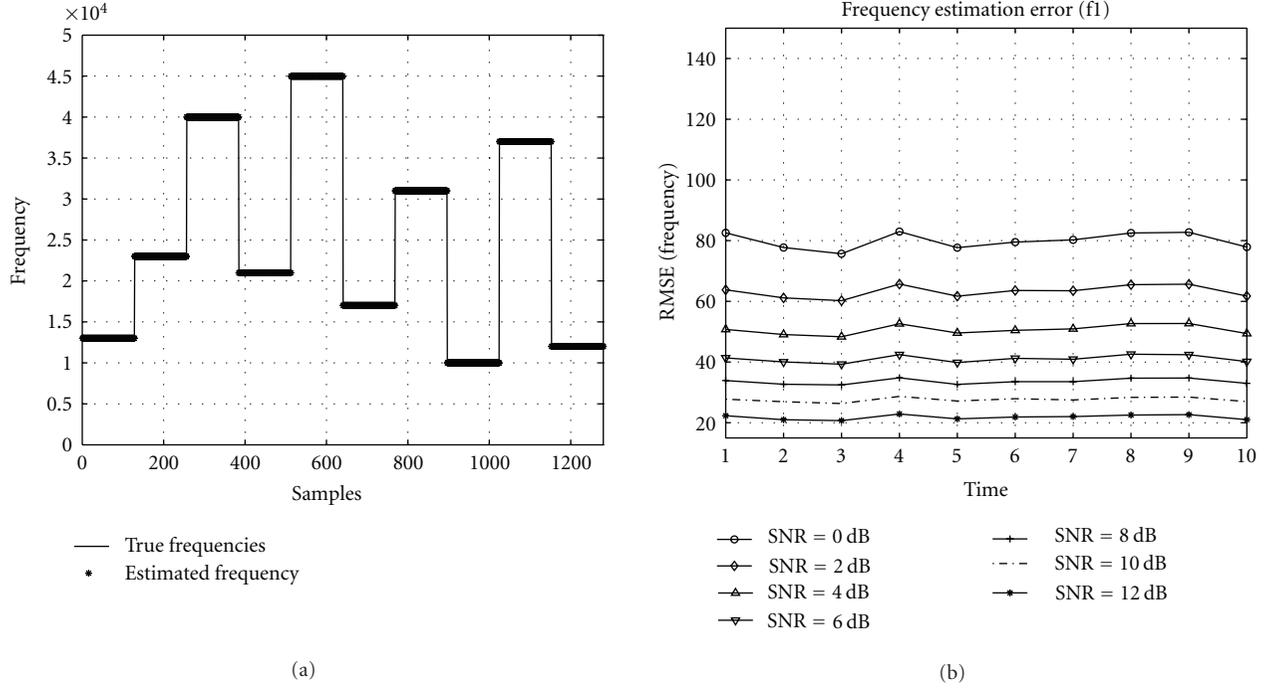


FIGURE 7: Root mean square estimation error versus tracking frequency with initial $K = random$. (a) Frequency tracking and (b) RMSE of frequency tracking.

The log-likelihood functions are

$$\ln L_1 = \text{const} - \frac{M}{2} \ln \sigma - \frac{1}{2\sigma^2} \|\mathbf{x}_1 - a_1 \mathbf{s}_1\|^2, \quad (\text{A.3})$$

$$\ln L_2 = \text{const} - \frac{M}{2} \ln \sigma - \frac{1}{2\sigma^2} \|\mathbf{x}_2 - a_2 \mathbf{s}_2\|^2. \quad (\text{A.4})$$

Equations (A.3) and (A.4) have similar forms. The CRLB that is associated with (A.3) is derived as follows. (That associated with (A.4) can be derived similarly). Differentiation of (A.3) with respect to ω_1 yields

$$\begin{aligned} \frac{\partial \ln L_1}{\partial \omega_1} &= \frac{\partial}{\partial \omega_1} \left(\text{const} - \frac{M}{2} \ln \sigma - \frac{1}{2\sigma^2} \|\mathbf{x}_1 - a_1 \mathbf{s}_1\|^2 \right) \\ &= -\frac{1}{2\sigma^2} \left(\frac{\partial (\mathbf{x}_1 - a_1 \mathbf{s}_1)^H}{\partial \omega_1} (\mathbf{x}_1 - a_1 \mathbf{s}_1) \right. \\ &\quad \left. + (\mathbf{x}_1 - a_1 \mathbf{s}_1)^H \frac{\partial (\mathbf{x}_1 - a_1 \mathbf{s}_1)}{\partial \omega_1} \right) \\ &= -\frac{1}{2\sigma^2} \left(-a_1^* (-j \mathbf{s}_1^H \mathbf{C}) (\mathbf{x}_1 - a_1 \mathbf{s}_1) \right. \\ &\quad \left. + (\mathbf{x}_1 - a_1 \mathbf{s}_1)^H (-a_1 j \mathbf{C} \mathbf{s}_1) \right) \end{aligned}$$

$$\begin{aligned} &= -\frac{1}{2\sigma^2} \left(ja_1^* \mathbf{s}_1^H \mathbf{C} \mathbf{x}_1 - ja_1 a_1^* \mathbf{s}_1^H \mathbf{C} \mathbf{s}_1 - ja_1 \mathbf{x}_1^H \mathbf{C} \mathbf{s}_1 \right. \\ &\quad \left. + ja_1 a_1^* \mathbf{s}_1^H \mathbf{C} \mathbf{s}_1 \right) \\ &= -\frac{1}{2\sigma^2} \left(ja_1^* \mathbf{s}_1^H \mathbf{C} \mathbf{x}_1 - ja_1 \mathbf{x}_1^H \mathbf{C} \mathbf{s}_1 \right) \\ &= \frac{1}{2\sigma^2} \left(ja_1 \mathbf{x}_1^H \mathbf{C} \mathbf{s}_1 - ja_1^* \mathbf{s}_1^H \mathbf{C} \mathbf{x}_1 \right), \end{aligned} \quad (\text{A.5})$$

where

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & K \end{bmatrix}, \quad K = \frac{M}{2}. \quad (\text{A.6})$$

Further differentiating (A.5) yields

$$\begin{aligned} \frac{\partial^2 \ln L_1}{\partial \omega_1^2} &= \frac{\partial}{\partial \omega_1} \frac{1}{2\sigma^2} \left(ja_1 \mathbf{x}_1^H \mathbf{C} \mathbf{s}_1 - ja_1^* \mathbf{s}_1^H \mathbf{C} \mathbf{x}_1 \right) \\ &= \frac{1}{2\sigma^2} \left(ja_1 \mathbf{x}_1^H \mathbf{C} (j \mathbf{C} \mathbf{s}_1) - ja_1^* (-j \mathbf{s}_1^H \mathbf{C}) \mathbf{C} \mathbf{x}_1 \right) \quad (\text{A.7}) \\ &= \frac{-1}{2\sigma^2} \left(a_1 \mathbf{x}_1^H \mathbf{C} \mathbf{C} \mathbf{s}_1 + a_1^* \mathbf{s}_1^H \mathbf{C} \mathbf{C} \mathbf{x}_1 \right). \end{aligned}$$

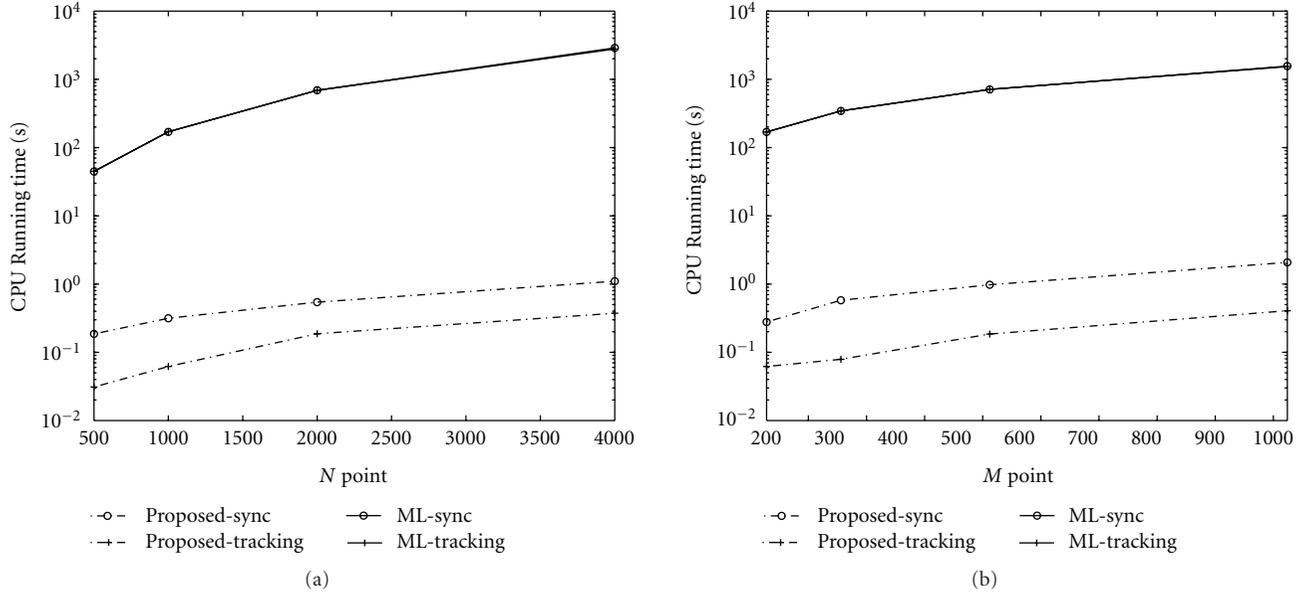


FIGURE 8: Comparison of CPU time. (a) Fixed $M = 128$ and $N = 500 \sim 4000$ and (b) fixed $N = 1000$ and $M = 128 \sim 1024$.

Substituting (A.7) into the Fisher information function yields

$$\begin{aligned}
 I(\omega_1) &= -E \left[\frac{\partial^2 \ln L_1}{\partial \omega_1^2} \right] \\
 &= \frac{1}{2\sigma^2} E \left[a_1 \mathbf{x}_1^H \mathbf{C} \mathbf{C} \mathbf{s}_1 + a_1^* \mathbf{s}_1^H \mathbf{C} \mathbf{C} \mathbf{x}_1 \right] \\
 &= \frac{1}{2\sigma^2} \left[a_1 a_1^* \mathbf{s}_1^H \mathbf{C} \mathbf{C} \mathbf{s}_1 + a_1^* \mathbf{s}_1^H \mathbf{C} \mathbf{C} a_1 \mathbf{s}_1 \right] \\
 &= \frac{a_1^2 \mathbf{s}_1^H \mathbf{C} \mathbf{C} \mathbf{s}_1}{\sigma^2} \\
 &= \frac{a_1^2}{\sigma^2} \sum_{k=1}^K k^2 \\
 &= \frac{a_1^2}{\sigma^2} \left(\frac{K(K+1)(2K+1)}{6} \right).
 \end{aligned} \tag{A.8}$$

Finally, the CRLB is given by

$$\begin{aligned}
 \text{var}_{\text{CRLB}} &= \frac{1}{I(\omega_1)} \\
 &= -\frac{\sigma^2}{a_1^2} \left(\frac{6}{K(K+1)(2K+1)} \right).
 \end{aligned} \tag{A.9}$$

Acknowledgment

The authors would like to thank the National Science Council of the Republic of China, Taiwan, for financially supporting this research under Contract no. NSC_99-2221-E-008 -039.

References

- [1] H. Zhao and Q. Wang, "On frequency hop synchronization in multipath rayleigh fading," *IEEE Transactions on Vehicular Technology*, vol. 47, no. 3, pp. 1049–1065, 1998.
- [2] J. Min and H. Samuelli, "Synchronization techniques for a frequency-hopped wireless transceiver," in *Proceedings of the 46th IEEE Vehicular Technology Conference*, pp. 183–187, May 1996.
- [3] J. S. Min and H. Samuelli, "Analysis and design of a frequency-hopped spread-spectrum transceiver for wireless personal communications," *IEEE Transactions on Vehicular Technology*, vol. 49, no. 5, pp. 1719–1731, 2000.
- [4] Y. M. Siu, W. S. Chan, and S. W. Leung, "A SFH spread spectrum synchronization algorithm for data broadcasting," *IEEE Transactions on Broadcasting*, vol. 47, no. 1, pp. 71–75, 2001.
- [5] Y. Qiu, Z. Gan, and Y. Pan, "Research on downlink synchronization of a frequency-hopping satellite communication system," in *Proceedings of the International Conference on Communication Technology Proceedings (ICCT '98)*, vol. 1, pp. S17051–S17054, October 1998.
- [6] W. Li, J. Wang, and Y. Yao, "Synchronization design of frequency-hopping communication system," in *Proceedings of the International Conference on Communication Technology Proceedings (ICCT '98)*, vol. 1, pp. 115–119, October 1998.
- [7] J. Liang, L. Gao, and S. Yang, "Frequency estimation and synchronization of frequency hopping signals based on reversible jump MCMC," in *Proceedings of the International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS '05)*, pp. 589–592, December 2005.
- [8] X. Liu, J. Li, and X. Ma, "An EM algorithm for blind hop timing estimation of multiple FH signals using an array system with bandwidth mismatch," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 5, pp. 2545–2554, 2007.

- [9] H. Fan, Y. Guo, and X. Feng, "Blind parameter estimation of frequency hopping signals based on matching pursuit," in *Proceedings of the 4th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '08)*, October 2008.
- [10] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.
- [11] X. Liu, N. D. Sidiropoulos, and A. Swami, "Joint hop timing and frequency estimation for collision resolution in FH networks," *IEEE Transactions on Wireless Communications*, vol. 4, no. 6, pp. 3063–3073, 2005.
- [12] A. Valyrakis, E. E. Tsakonas, N. D. Sidiropoulos, and A. Swami, "Stochastic modeling and particle filtering algorithms for tracking a frequency-hopped signal," *IEEE Transactions on Signal Processing*, vol. 57, no. 8, pp. 3108–3118, 2009.
- [13] C. C. Ko, W. Zhi, and F. Chin, "ML-based frequency estimation and synchronization of frequency hopping signals," *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 403–410, 2005.
- [14] I. Ziskind and M. Wax, "Maximum likelihood localization of multiple sources by alternating projection," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, no. 10, pp. 1553–1560, 1988.