electronics

Article

A Fast Estimation Algorithm for Parameters of Multiple Frequency-Hopping Signals Based on Compressed Spectrum Sensing and Maximum Likelihood

Yixing Li, Furong Wang, Gang Fan, Yang Liu and Ya Zhang *

College of Mechatronics Engineering, North University of China, Taiyuan 030051, China
* Correspondence: zy@nuc.edu.cn

Abstract: The parameter estimation of multiple frequency-hopping (multiple FH) signals with frequency-switching time is a great challenge under conditions in which the number of signals is unknown. Due to the increasing mobility of devices such as unmanned aerial vehicles (UAVs), speed of parameter estimation is even more demanding. To solve this problem, an algorithm for estimating parameters of multiple FH signals based on compressed spectrum sensing and maximum likelihood (CSML) theory is proposed in this paper. First, the received signal is split into segments of the same length, and the frequencies contained in each segment are extracted using compressed spectrum sensing and kurtosis threshold. Next, the frequencies contained in adjacent segments are compared to find the signal segment in which frequency hopping occurs and its corresponding frequency. Finally, a hopping-time fast estimation algorithm based on the maximum likelihood theory is used to estimate the hopping time. Simulation results show that the algorithm proposed in this paper can estimate the parameters of multiple FH signals quickly and accurately when the number of signals is unknown and that it is equally effective for multiple FH signals with frequency-switching time.

Keywords: frequency-hopping signal; maximum likelihood estimation; kurtosis threshold; enhanced two-hop model; compressed spectrum sensing

Citation: Li, Y.; Wang, F.; Fan, G.; Liu, Y.; Zhang, Y. A Fast Estimation Algorithm for Parameters of Multiple Frequency-Hopping Signals Based on Compressed Spectrum Sensing and Maximum Likelihood. *Electronics* **2023**, *12*, 1808. https://doi.org/10.3390/electronics12081808

Academic Editor: Simeone Marino

Received: 14 March 2023
Revised: 5 April 2023
Accepted: 7 April 2023
Published: 11 April 2023

1. Introduction

Radio communication is widely used in various fields as an important part of information technology. Initially, in radio communication, the signal is modulated by a transmitter into a constant-frequency radio frequency (RF) signal, which is then sent to a receiver. However, constant-frequency RF signals are very susceptible to interference from the complex surrounding electromagnetic environment and to eavesdropping by third parties [1,2]. In this case, frequency-hopping signals have gradually replaced constant-frequency RF signals due to their strong anti-interference capability and good confidentiality and are widely used for military communications and remote control [3–7]. Since the introduction of FH technology in the military, research on jamming, counter jamming, counter sabotage and sabotage for FH technology has never stopped [8]. The emergence of frequency-hopping signals with frequency-switching time is a good proof of this. In order to intercept and jam the frequency-hopping signals of non-cooperative targets, precise knowledge of the parameters of their transmitted FH signals is required [9]. Therefore, parameter estimation of frequency-hopping (FH) signals is an important task for information interception and autonomous electronic countermeasures [10,11]. The battlefield environment is rapidly changing, and the fast estimation of FH signals in the case of unknown frequency, an unknown number of signals or signals with frequency-switching time is of great importance. The parameter estimation of FH signals includes hopping time, hopping speed and frequency [12]. Researchers have conducted numerous studies on the estimation methods of FH signal parameters.

Time–frequency analysis algorithms, such as short-time Fourier transform (STFT) [13–15] and the Wigner–Ville distribution (WVD) [8,16], are classical algorithms for estimating parameters of FH signals. Although the STFT algorithm is less computationally intensive, it cannot meet the high accuracy requirements of both the frequency and time domains at the same time [17,18]. When used to estimate the parameters of a multiple FH signal, the WVD algorithm is disrupted by cross terms [19,20]. In Ref. [21], an algorithm for parameter estimation of FH signals based on adaptive smoothing of the Wigner–Ville distribution and instantaneous frequency (SWWVD-IF) was proposed. Although this algorithm can overcome the interference problem of cross terms, it has a high computational complexity. In Ref. [22], a parameter estimation algorithm for multiple frequency-hopping signals based on short-time Fourier transform and smoothed pseudo Wigner–Ville distribution (STFT&SPWVD) was proposed. The STFT&SPWVD algorithm combines the advantages of STFT, which avoids cross-term interference, and SPWVD, which provides higher resolution in the frequency–time domain. However, the computational complexity of this algorithm is slightly higher than that of the SPWVD algorithm. With the development of frequency-hopping technology, the frequency-hopping signal hops over a larger frequency band. The time–frequency analysis algorithm based on Nyquist's sampling theorem must obtain the entire signal at a high sampling rate [23]. This poses a serious challenge for signal reception equipment.

Provided that the original signal is sparse or can be sparsely represented, compressive sensing theory allows the signal to be acquired at a frequency lower than the sub-Nyquist sampling frequency [24,25]. This has led to the increasing application of compressive sensing theory to FH signal processing. Popular algorithms based on this theory include sparse linear regression (SLR) algorithms [26–28] and sparse Bayesian learning (SBL) algorithms [20,29,30]. These algorithms are very computationally intensive. In Ref. [31], a new algorithm combining orthogonal matching pursuit (OMP) and sparse linear regression named OSLR was introduced to reduce the computational effort of the algorithm. The OSLR algorithm has a lower time complexity than the SLR algorithm. However, it requires the number of signals of multiple FH signals as a priori information to achieve correct frequency estimation [32], and when the signals have frequency-switching time, the number of signals changes during the frequency-switching time, so it cannot estimate the parameters of the signals with frequency-switching time. In [32], a new parameter estimation algorithm (CSOMP-ASW) was introduced that combines compressed spectrum sensing, OMP, a frequency clustering algorithm and an adaptive sliding window algorithm. The CSOMP-ASW algorithm uses the compressed spectrum sensing algorithm to evaluate the sparsity of the signal, then uses the OMP algorithm and the clustering algorithm to estimate the frequency of the signal so that it is not fast enough to estimate the frequency.

In Ref. [33], a frequency-hopping synchronization algorithm based on maximum likelihood (ML) theory was proposed. This algorithm is computationally intensive and is only applicable to parameter estimation of a single FH signal without frequency-switching time. To reduce the computational complexity required to estimate parameters of multiple frequency-hopping signals under conditions in which the number of signals is known, in our previous work, we proposed a frequency-hopping signal parameter estimation algorithm called OML [34]. This algorithm estimates the frequency using the OMP algorithm and the hopping time using an algorithm traversing all time points based on the maximum likelihood theory, which partially improves the speed of hopping signal parameter estimation.

In summary, all the above research methods have different degrees of improvement in reducing the computational complexity and increasing the accuracy of parameter estimation, but there are still some shortcomings. In order to mitigate the limitations of the above methods and realize the fast estimation of frequency and hopping time of multiple FH signals under conditions in which the frequency is unknown, the number of signals is unknown or the signals have frequency-switching time, a new frequency-hopping algorithm (CSML) is proposed in this paper. First, based on the sparsity of multiple FH

signals in the frequency domain, a combination of compressed spectrum sensing and kurtosis thresholding is used to estimate the frequencies. The ML-based hopping-time fast estimation algorithm is then used to estimate the hopping time. The main contributions of this paper are as follows:

- An enhanced two-hop model (ETHM) is developed to characterize multiple FH signals; then, a likelihood function is derived from the model for the estimation of the parameters of the multiple FH signals. Based on this likelihood function, a hopping-time fast estimation algorithm is proposed that reduces the time required for estimation of the hopping time of multiple FH signals.
- The algorithm based on compressed spectrum sensing has low accuracy in performing frequency estimation. The reasons for this phenomenon are explored, and a zero-setting method and averaging methods are proposed to improve the effectiveness of this algorithm. Satisfactory performance is still obtained for the estimation of multiple FH signal frequencies without the involvement of other algorithms such as OMP. Fast estimation of the frequency of multiple FH signals without a priori information is achieved.

The remainder of this paper is organized as follows. In Section 2, an enhanced two-hop model is introduced. Section 3 provides details of the proposed CSML algorithm. Section 4 presents the simulations and results of the CSML algorithm. Finally, in Section 5, we present our conclusions.

Notation:
$\mathbf{A}^{\mathrm{T}}$—The transposition of $\mathbf{A}$;
$\mathbf{A}^{\mathrm{H}}$—The conjugate transposition of $\mathbf{A}$;
$\mathbf{A}^{-1}$—The inverse of $\mathbf{A}$;
$\mathbf{0}_Q$—The $Q \times 1$-dimensional zero vector;
$\mathbf{I}_Q$—The $Q \times 1$-dimensional unit vector;
$\|\mathbf{A}\|$—Euclidean norm of $\mathbf{A}$;
$\langle \mathbf{A} \rangle$—The inner product of $\mathbf{A}$;
$\hat{a}$—The estimated value of $\hat{a}$.

2. The Enhanced Two-Hop Model

In order to realize the parameter estimation of multiple FH signals with frequency-switching time, a multiple FH signal model is developed in this paper. To distinguish it from the two-hop model proposed in [33], it is named the enhanced two-hop model (ETHM). A schematic diagram of the ETHM is shown in Figure 1. In Figure 1, L is the total number of samples, and the rest of the notations are annotated under (1). The ETHM is composed of two FH signal models and two received signal models.

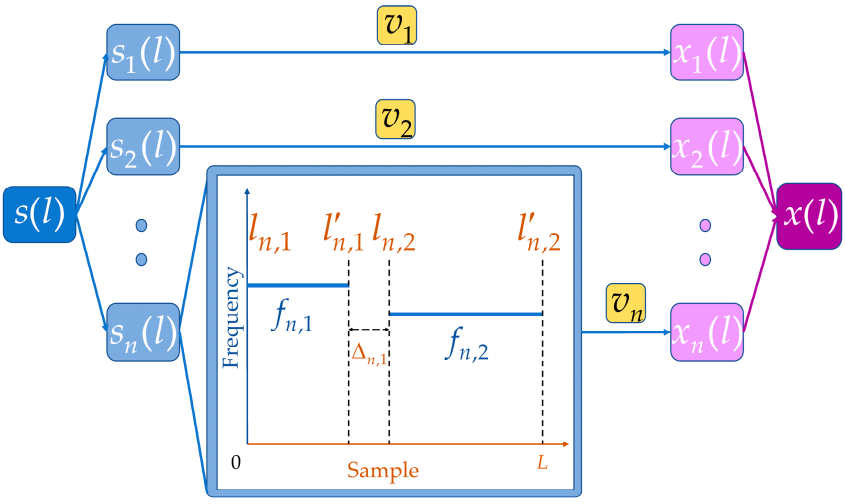

Figure 1. The enhanced two-hop model.

The nth FH signal can be denoted as:

$$s_n(l) = \begin{cases} e^{j2\pi f_{n,m} l/f_s} & , l_{n,m} \le l < l_{n,m+1} - \Delta_{n,m}, m = 1,2 \\ 0 & , \text{others} \end{cases}, \tag{1}$$

where m is the number of hops, $f_{n,m}$ is the frequency of the mth hop, $l_{n,m}$ is the starting time of the mth hop, $\Delta_{n,m}$ is the frequency-switching time between the mth hop and the $(m+1)$th hop, $l_{n,m+1} - \Delta_{n,m}$ is the stopping time of the mth hop (denoted as $l'_{n,m}$) and f_s is the sampling frequency.

The received signal corresponding to the nth FH signal can be expressed as:

$$x_n(l) = \begin{cases} a_{n,m} e^{j2\pi f_{n,m} l/f_s} + v_n(l) & , l_{n,m} \le l < l'_{n,m}, m = 1,2 \\ v_n(l) & , \text{others} \end{cases} \tag{2}$$

where $a_{n,m}$ is the amplitude corresponding to the mth hop of the nth FH signal, and v_n is the additive Gaussian white noise.

The multiple FH signals ($s(l)$), which are a mixture of N FH signals, can be indicated as:

$$\begin{aligned} s(l) &= \sum_{n=1}^{N} s_n(l) \\ &= \sum_{n=1}^{N} e^{j2\pi f_{n,m} l/f_s} \quad , l_{n,m} \le l < l'_{n,m}, m = 1,2. \end{aligned} \tag{3}$$

The received signal corresponding to $s(l)$ can be represented as:

$$x(l) = \sum_{n=1}^{N} x_n(l). \tag{4}$$

This paper only considers the case in which the spectrum of each hopping signal does not overlap.

3. CSML Algorithm

3.1. Likelihood Function

Under the condition in which the frequencies are known, the computational complexity of the ML-based algorithm is reduced by not traversing all frequencies and time points simultaneously. Furthermore, since the shape of the curve of the likelihood function is fixed, the correct hopping time can be obtained without traversing all time points. Therefore, the ML-based algorithm has the potential to estimate the hopping time quickly under the condition in which the frequency is known. The frequencies of multiple FH signals can also be provided by specialized frequency estimation algorithms (the frequency estimation algorithm used in this paper is based on the compressed spectrum sensing and kurtosis thresholding algorithm proposed in Section 3.2).

According to Equations (1) and (2), the received signal corresponding to the nth FH signal can be re-expressed as:

$$\mathbf{x}_n = \mathbf{s}_n + \mathbf{v}_n, \tag{5}$$

where:

$$\begin{aligned} \mathbf{x}_n &= [x_n(1), \ldots, x_n(L)]^{\mathrm{T}}, \\ \mathbf{s}_n &= \Big[a_{n,1} e^{j2\pi f_{n,1}/f_s}, \ldots, a_{n,1} e^{j2\pi f_{n,1}(l'_{n,1})/f_s}, 0, \ldots, 0, \\ &\qquad a_{n,2} e^{j2\pi f_{n,2} l_{n,2}/f_s}, \ldots, a_{n,2} e^{j2\pi f_{n,2} L/f_s}\Big]^{\mathrm{T}}, \\ \mathbf{v}_n &= [v_n(1), \ldots, v_n(L)]^{\mathrm{T}}. \end{aligned} \tag{6}$$

According to Equations (5) and (6), $\mathbf{x}_n$ can be divided into two parts as follows:

$$\begin{aligned} \mathbf{x}_{n,1} &= a_{n,1}\mathbf{s}_{n,1} + \mathbf{v}_{n,1}, \\ \mathbf{x}_{n,2} &= a_{n,2}\mathbf{s}_{n,2} + \mathbf{v}_{n,2}, \end{aligned} \tag{7}$$

where:

$$\begin{aligned} \mathbf{x}_{n,1} &= [x_n(1), \ldots, x_n(l'_{n,1})]^{\mathrm{T}}, \\ \mathbf{x}_{n,2} &= [x_n(l_{n,2}), \ldots, x_n(L)]^{\mathrm{T}}, \\ \mathbf{s}_{n,1} &= [e^{j2\pi f_{n,1}/f_s}, \ldots, e^{j2\pi f_{n,1}(l'_{n,1})/f_s}]^{\mathrm{T}}, \\ \mathbf{s}_{n,2} &= [e^{j2\pi f_{n,2}(l_{n,2})/f_s}, \ldots, e^{j2\pi f_{n,2}(L)/f_s}]^{\mathrm{T}}, \\ \mathbf{v}_{n,1} &= [v_n(1), \ldots, v_n(l'_{n,1})]^{\mathrm{T}}, \\ \mathbf{v}_{n,2} &= [v_n(l_{n,2}), \ldots, v_n(L)]^{\mathrm{T}}. \end{aligned} \tag{8}$$

According to Equations (7) and (8), the following likelihood function can be derived:

$$\begin{aligned} L(a_{n,1}, f_{n,1}, l'_{n,1}) &= \frac{1}{\left(\sqrt{2\pi}\sigma\right)^{l'_{n,1}}} e^{-\frac{1}{2\sigma^2}\|\mathbf{x}_{n,1} - a_{n,1}\mathbf{s}_{n,1}\|^2}, \\ L(a_{n,2}, f_{n,2}, l_{n,2}) &= \frac{1}{\left(\sqrt{2\pi}\sigma\right)^{L-l_{n,2}+1}} e^{-\frac{1}{2\sigma^2}\|\mathbf{x}_{n,2} - a_{n,2}\mathbf{s}_{n,2}\|^2}. \end{aligned} \tag{9}$$

Thus, the parameters $l'_{n,1}$ and $l_{n,2}$ can be obtained by minimizing (10) and (11).

$$\varphi_{n,1}(a_{n,1}, f_{n,1}, l'_{n,1}) = \|\mathbf{x}_{n,1} - a_{n,1}\mathbf{s}_{n,1}\|^2. \tag{10}$$

$$\varphi_{n,2}(a_{n,2}, f_{n,2}, l_{n,2}) = \|\mathbf{x}_{n,2} - a_{n,2}\mathbf{s}_{n,2}\|^2. \tag{11}$$

After derivation, the parameters $l'_{n,1}$ and $l_{n,2}$ can be obtained by maximizing (A8), as shown in (12). The derivation procedure is shown in Appendix A.

$$\begin{aligned} \hat{l}'_{n,1} &= \underset{l'_{n,1}}{\operatorname{argmax}}[\overline{\varphi}_{n,1}(l'_{n,1})], \\ \hat{l}_{n,2} &= \underset{l_{n,2}}{\operatorname{argmax}}[\overline{\varphi}_{n,2}(l_{n,2})]. \end{aligned} \tag{12}$$

3.2. The Hopping-Time Fast Estimation Algorithm

As illustrated in (12), when estimating the hopping time of multiple FH signals, all the time points are traversed. This makes the CPU time consumed in the process of estimating the hopping time too long. To solve this problem, a fast estimation algorithm for hopping time is proposed in this subsection.

Assume that there is an FH signal with a length of 6400 ($L = 6400$) and a hopping time of 2600 ($l_{1,1} = 0, l'_{1,1} = 2600$). The variation of $\overline{\varphi}$ with l for this FH signal is shown in Figure 2a. The maximum $\overline{\varphi}$ occurs when l is equal to $l'_{1,1}$. When l is less than $l'_{1,1}$, the $\overline{\varphi}$ increases monotonically. When l is greater than $l'_{1,1}$, $\overline{\varphi}$ decreases monotonically. Due to the good monotonicity of $\overline{\varphi}$ before and after the hopping time $l'_{1,1}$, it is possible to select l uniformly at intervals of c in $[1, 6400]$; then, the $\overline{\varphi}$ corresponding to l is calculated. This narrows the existence of $l'_{1,1}$ to c points.

Figure 2b shows the variation of $\overline{\varphi}$ with l when $c_1 = 1000$ and $c_2 = 1$. The $\overline{\varphi}$ of these six sampling points is calculated (marked by black circles in Figure 2b, $c_1 = 1000$), and the two largest $\overline{\varphi}$ values correspond to $l = 2000$ and $l = 3000$, respectively. This shows that the hopping time $l'_{1,1}$ is in the range of $[2000, 3000]$. The hopping time $l'_{1,1}$ can be obtained by traversing l in the range of $[2000, 3000]$ (marked by the blue line in Figure 2b, $c_2 = 1$). With this algorithm, only 1005 calculations of $\overline{\varphi}$ are required to obtain an estimate of the hopping time $l'_{1,1}$, which is only one-sixth of the previous workload. The optimal value of parameter c will be discussed in Section 4.

The specific steps of the hopping-time fast estimation algorithm are described below:

Step 1: Select sampling points at intervals of c and calculate $\overline{\varphi}$ using (A8);

Step 2: Compare the $\overline{\varphi}$ calculated in step 1, and select the sampling points with the two largest $\overline{\varphi}$ values;

Step 3: Calculate the $\overline{\varphi}$ of the sampling points between the two points selected in step 2.

Step 4: Compare the $\overline{\varphi}$ calculated in step 3, and select the sampling point with the largest $\overline{\varphi}$ value as the hopping time.

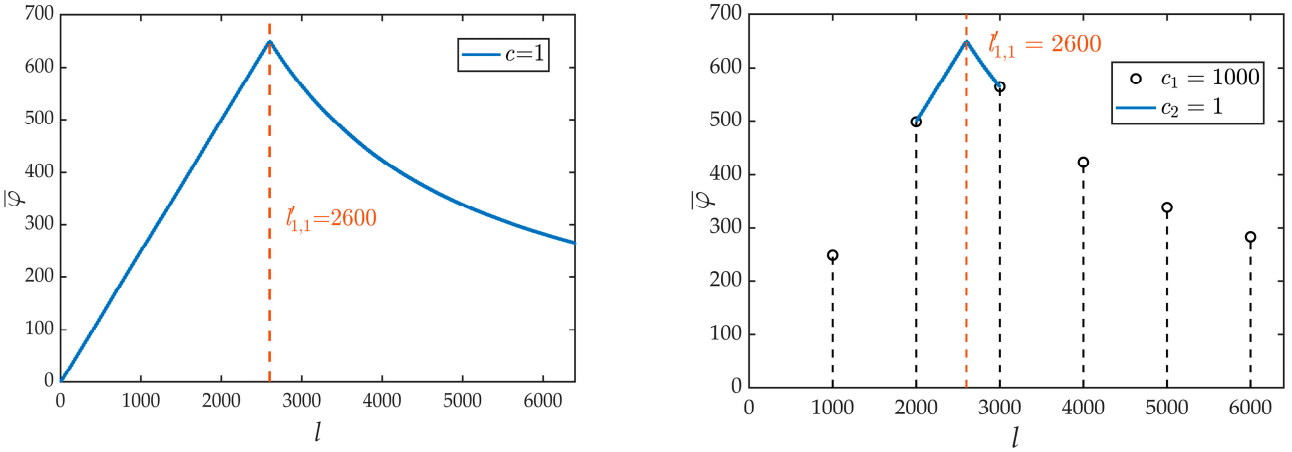

Figure 2. The variation of $\overline{\varphi}$ with l. (**a**) The variation of $\overline{\varphi}$ with l for an FH signal with an L of 6400 and $l'_{1,1}$ of 2600 (the curve of the calculated $\overline{\varphi}$ with l when $c = 1$). (**b**) The curve of the calculated $\overline{\varphi}$ with l when $c_1 = 1000$ and $c_2 = 1$.

3.3. The Frequency Estimation Algorithm

As in [32], the sparsity of the FH signal in the frequency domain is used to obtain the FH signal spectrum in the compressed domain. The relationship between the compressed sampled signal ($\mathbf{y}$) and the received signal ($\mathbf{x}$) can be expressed as:

$$\mathbf{y} = \mathbf{\Phi x} = \mathbf{\Phi \Psi b} = \mathbf{\Theta b}, \tag{13}$$

where $\mathbf{y} \in \mathbb{R}^Y$ is the compressed sampled signal of the received signal ($\mathbf{x} \in \mathbb{R}^L$), $\mathbf{\Phi} \in \mathbb{R}^{Y \times G_0}$ is the Gaussian random matrix, $\mathbf{\Psi} \in \mathbb{R}^{G_0 \times G_0}$ is the Fourier orthogonal matrix, $\mathbf{\Theta}$ is the compressed sensing matrix, and $\mathbf{b}$ is the factor matrix of the received signal ($\mathbf{x}$) corresponding to $\mathbf{\Psi}$. The compression ratio is $\lambda = G_0/Y$, and G_0 is equal to L. The gth column of $\mathbf{\Psi}$ can be expressed by the following equation:

$$\Psi_g = [1, e^{j\pi g/G_0}, e^{j\pi 2g/G_0}, e^{j\pi 3g/G_0}, \ldots, e^{j\pi(G_0-1)g/G_0}]. \tag{14}$$

Its corresponding frequency can be expressed as:

$$F_g = f_s(g-1)/G_0. \tag{15}$$

According to (13), the inner product of $\mathbf{y}$ and $\mathbf{\Theta}$ can be expressed in the form of (16). $\mathbf{p}$ is the magnitude of the projection of the received signal ($\mathbf{x}$) on the Fourier orthogonal matrix; therefore, the spectrum of $\mathbf{x}$ can be obtained. Figure 3a illustrates the spectrum of the received signal at a signal-to-noise ratio (SNR) of 30 dB; Figure 3b shows the spectrum of the noise signal.

$$\mathbf{p} = |\langle \mathbf{y}, \mathbf{\Theta} \rangle|, \tag{16}$$

where $\mathbf{p} = [p_1, p_2, \ldots, p_{G_0}]^{\mathrm{T}}$.

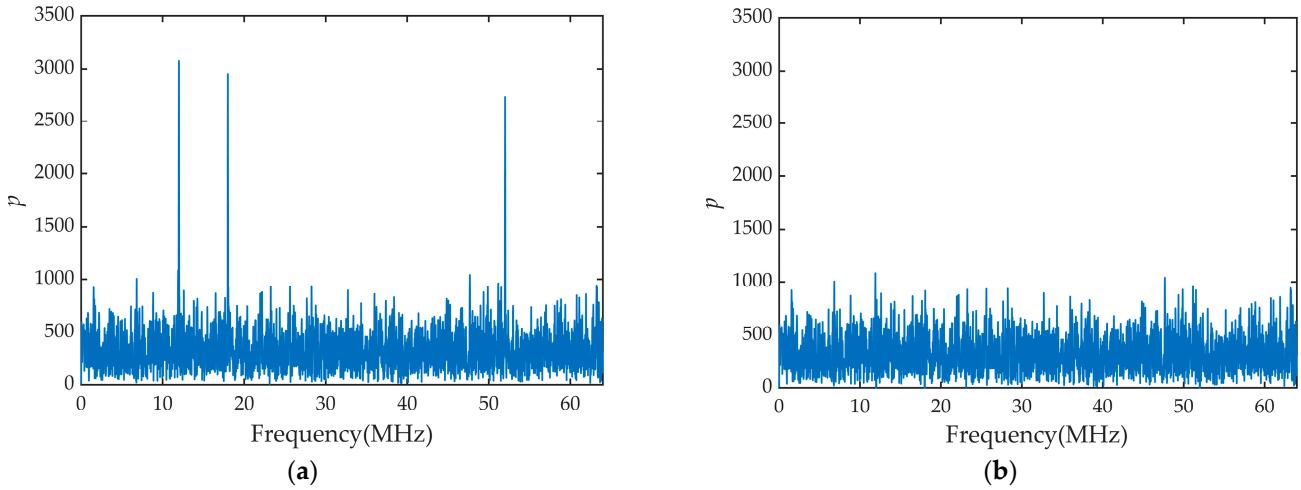

Figure 3. Spectrum of the signal. (**a**) The obtained spectrum refers to the case of the received signal when SNR = 30 dB. (**b**) The spectrum of the noise signal.

As shown in Figure 3a, the frequencies corresponding to the three peaks are 12 MHz, 18 MHz and 54 MHz, which are the frequencies of the multiple FH signals and can be obtained by comparing the magnitude of p in the spectrum. Due to the presence of noise, the spectrum contains a large number of non-zero values. It is risky to discriminate whether the frequency is the frequency of multiple FH signals by setting a threshold on the value of p.

The kurtosis value (ε), defined by (17) [35], is sensitive to the presence of peaks in the data; the spectrum in Figure 3a contains three peaks, and the kurtosis value (ε) of this spectrum is equal to 19.7, while the spectrum in Figure 3b does not contain peaks, and the kurtosis value (ε) of this spectrum is 1.6.

$$\varepsilon = \frac{\sum_{i=1}^{G_0}(p_i - \overline{p})^4 / G_0}{\left(\sum_{i=1}^{G_0}(p_i - \overline{p})^2 / G_0\right)^2}, \tag{17}$$

where $\overline{p} = \sum_{i=1}^{G_0} p_i / G_0$.

Taking Figure 3 as an example, the frequency corresponding to the largest p value is taken as the frequency of the multiple FH signals and recorded, and the largest p value is set to zero (zero-setting method) or to the average of p on both sides of it (averaging method). After three operations, the spectrum of the received signal changes from that shown in Figure 3a to that shown in Figure 3b, and the kurtosis value (ε)changes from 19.1 to 1.6. Therefore, a suitable kurtosis value can be selected as the stopping condition of the frequency estimation algorithm, known as the kurtosis threshold (μ).

During the operation, it was found that the p (marked by the yellow triangle in Figure 4a,b) on both sides of the maximum p was also large, as shown in Figure 4a,b. This is the reason for the low accuracy of the frequency estimation algorithm based on compressed spectrum sensing. Therefore, the p on both sides of the maximum p should be processed using the zero-setting method or the averaging method.

Assuming that the p at 12 MHz is p_{601} (the 601st value in the vector **P**), the zero-setting method can be expressed as $[p_{601-\alpha}, \ldots, p_{601+\alpha}]^{\mathrm{T}} = \mathbf{0}_{2\alpha+1}$, where α is the neighborhood radius. Figure 5a shows the partial enlargement of the spectrogram at 12 MHz after using the zero-setting method when the neighborhood radius (α) is equal to 6. The averaging method can be expressed as $[p_{601-\alpha}, \ldots, p_{601+\alpha}]^{\mathrm{T}} = I_{2\alpha+1} \times (p_{600-\alpha} + p_{602+\alpha})/2$. Figure 5b illustrates the partial enlargement of the spectrogram at 12 MHz after using the averaging method when the neighborhood radius (α) is equal to 6. The optimal values of the kurtosis threshold (μ) and the neighborhood radius (α) will be further investigated in Section 4.

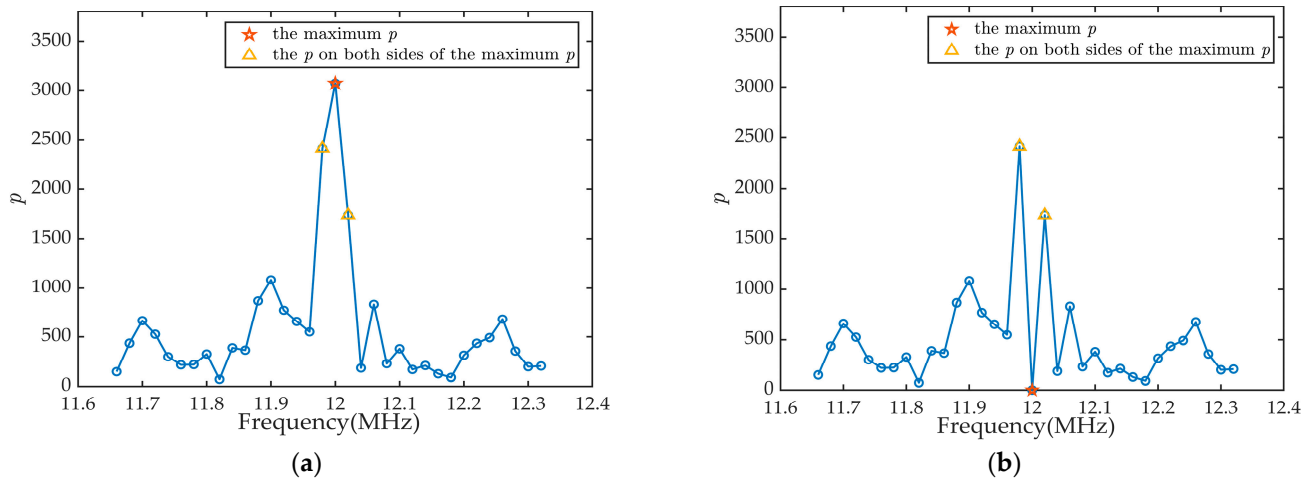

Figure 4. Partial enlargement of the spectrum at 12 MHz. (**a**) Partial enlargement of the spectrum shown in Figure 3a at 12 MHz. (**b**) Partial enlargement of the spectrum shown in Figure 3a at 12 MHz After using the zero-setting method.

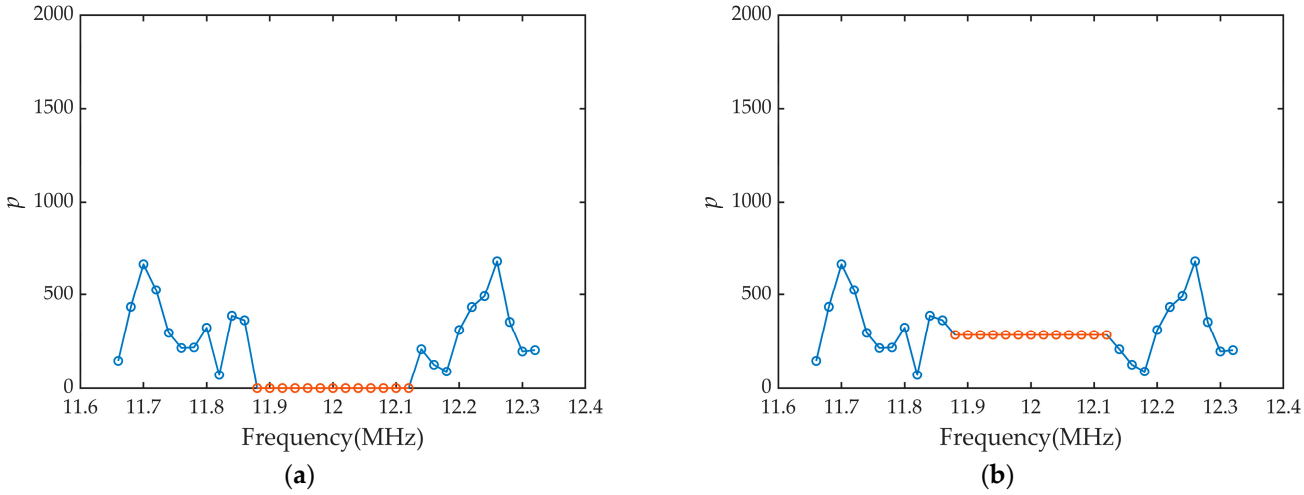

Figure 5. Partial enlargement of the spectrum using different p processing methods. (**a**) Partial enlargement of the spectrum at 12 MHz after using the zero-setting method for the parameter $\alpha = 6$. (**b**) Partial enlargement of the spectrum at 12 MHz after using the averaging method for the parameter $\alpha = 6$.

3.4. Steps of the CSML Algorithm

Estimating the frequency and hopping time of multiple FH signals is the main task of the CSML algorithm proposed in this paper. The CSML algorithm estimates the frequency first, then the hopping time. A flow chart of the CSML algorithm is shown in Figure 6.

The specific steps are described as follows:

Step 1: Split the received signal ($\mathbf{x} \in \mathbb{R}^L$) into segments of size G_0, where K is the total number of signal segments:

$$\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_K] \in \mathbb{R}^{G_0 \times K}. \tag{18}$$

Step 2: Obtain the compressed spectrum of the each segment using Equations (13), (15) and (16);

Step 3: Record the frequency corresponding to the maximum p. Set the maximum p to zero using the zero-setting method or the averaging method. Repeat this step until the kurtosis value (ε) of this compressed spectrum is smaller than the kurtosis threshold μ;

Step 4: Repeat step 3 for the compressed spectrum of the remaining signal segments until the frequencies of all signal segments are obtained. Compare the frequencies between adjacent segments to find the frequency of the multiple FH signals and the segment in which the frequency hopping occurred;

Step 5: Estimate the hopping time using the hopping-time fast estimation algorithm. Repeat this step until all hopping times are obtained.

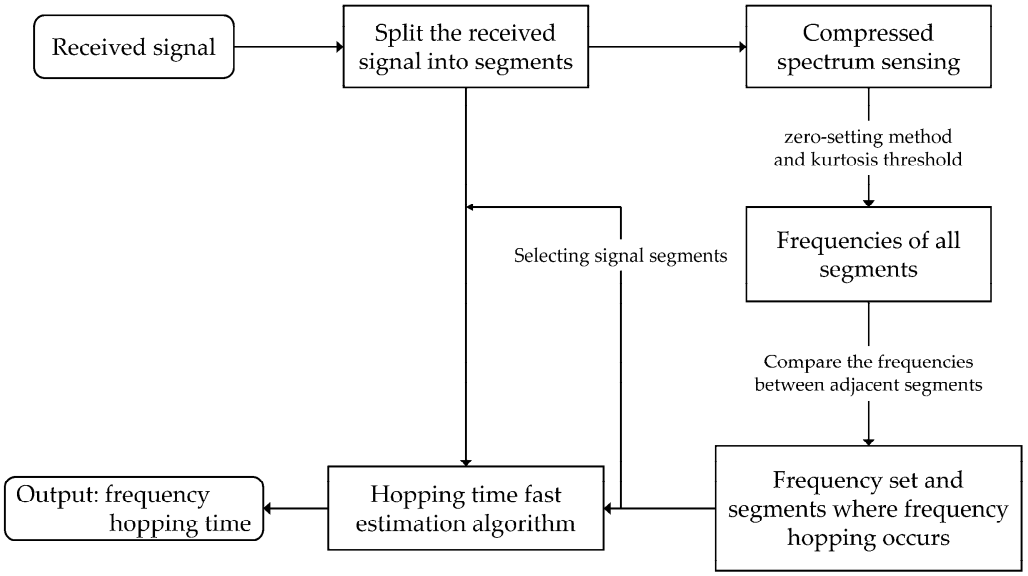

Figure 6. Flow chart of the CSML algorithm.

The purposes of splitting the received signal are as follows:

- Avoid processing large amounts of data at the same time and reduce the requirement for hardware devices;
- By comparing the frequencies of each segment, the segments in which the frequency hopping occurs, as well as FH frequencies, are identified, in addition to preparing for the estimation of hopping time.

It is worth noting that it is not the entire received signal that is fed into the hopping-time fast estimation algorithm but the selected signal segments where frequency hopping occurs after frequency estimation. The length of these segments is G_0. Therefore, the computational complexity of the fast estimation algorithm of the frequency-hopping signal parameters does not increase with the length of the received signal. To avoid errors, the size of the segments (G_0) should be smaller than the minimum FH signal duration.

3.5. Complexity Analysis

Continuing with the relevant symbols from the previous section, the number of FH signals is N, the sampling length of the signal is L, the received signal is divided into segments of length G_0, K is the total number of signal segments; the length of the compressed sampled signal is Y and the compression ratio of compressed sampling is λ. A slight difference is that K' is the total number of signal segments in which frequency hopping occurs.

The computational complexity of the compressed spectrum sensing part can be expressed as:

$$\xi_1 = KY = \frac{KG_0}{\lambda} = \frac{L}{\lambda}. \tag{19}$$

The computational complexity of the kurtosis value calculation once is G_0 can be expressed as:

$$\xi_2 = KNG_0 = LN. \tag{20}$$

The computational complexity of the hopping-time fast estimation algorithm can be expressed as:

$$\xi_3 = K'N(\frac{G_0}{c} + c - 1). \tag{21}$$

The computational complexity of the CSML algorithm proposed in this paper is:

$$\xi_{CSML} = \xi_1 + \xi_2 + \xi_3 = \frac{L}{\lambda} + LN + K'N(\frac{G_0}{c} + c - 1). \tag{22}$$

The computational complexity of the ASW algorithm for one iteration can be expressed as [32]:

$$\xi_4 = \frac{NG_0^2}{4\lambda} \tag{23}$$

The computational complexity of the ASW algorithm in the process of estimating the hopping time is as follows:

$$\xi_4' = \frac{K'NG_0^2}{4\lambda} \tag{24}$$

The computational complexity of the CSOMP-ASW algorithm [32] is:

$$\xi_{CSOMP-ASW} = \frac{L}{\lambda} + \frac{LNG_0}{\lambda} + \frac{K'NG_0^2}{4\lambda}. \tag{25}$$

Usually, λ is much smaller than G_0. Therefore, the computational complexity of the CSML algorithm proposed in this paper is smaller than that of CSOMP-ASW.

4. Simulations and Results

In this section, 100 Monte Carlo tests are performed on the proposed CSML algorithm to investigate the optimal values of the parameters μ, α and c. Our simulations were performed on a desktop computer with an Intel(R) Core(TM) i5-9400F CPU @2.90 GHz processor. The operating system was Microsoft Windows 10, with 16 GB of RAM. The comparison with the four algorithms mentioned in Section 1 is also performed. The sampling length L of the signal used in the simulations is 51200, and the sampling frequency (f_s) of the signal is 128 MHz.

The parameters of the multiple FH signals without frequency-switching time are shown in Table 1, and a schematic diagram of the multiple FH signals is shown in Figure 7. The multiple FH signals are used for the simulation tests in Sections 4.1–4.4. FH signal No. 1 ($n = 1$) starts at 12 MHz, and the frequency becomes 24 MHz when $l = 9001$; the starting frequency of FH signal No. 2 ($n = 2$) is 54 MHz, and the frequency changes to 60 MHz when $l = 20901$; signal No. 3 ($n = 3$) is a fixed-frequency signal with a frequency of 18 MHz.

Table 1. The parameters of the multiple FH signals.

n	$f_{n,1}$ (MHz)	$f_{n,2}$ (MHz)	$l'_{n,1}$	$\Delta_{n,1}$	$l_{n,2}$
$n = 1$	12	24	9000	1	9001
$n = 2$	54	60	20,900	1	20,901
$n = 3$	18		51,200		

Other necessary parameters are shown in Table 2. The number of FH signals (N) is two; the sampling length of the signal (L) is 51,200; the sampling frequency (f_s) is 128 MHz; the received signal is divided into segments of length $G_0 = 3200$, and the compression ratio of compressed sampling (λ) is 8.

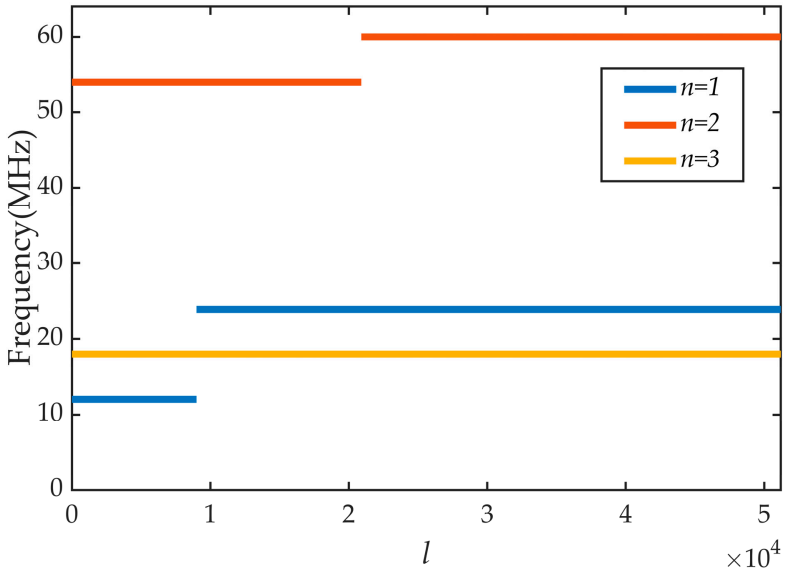

Figure 7. Schematic diagram of the multiple FH signals.

Table 2. Other necessary parameters.

N	L	f_s (MHz)	G_0	λ
2	51,200	128	3200	8

The SNR is given by:

$$\mathrm{SNR} = 10\log_{10}\left(\frac{\|\mathbf{s}\|_2^2}{N\sigma^2}\right), \tag{26}$$

where $\mathbf{s}$ is the multiple FH signals, and σ^2 is the variance of Gaussian white noise.

The common performance indicators of FH signal parameter estimation algorithms include mean square error (MSE), CPU time, accuracy, etc. In this paper, the frequency estimation accuracy and CPU time are used to evaluate the frequency estimation performance of the proposed algorithm; the mean square error (MSE) of the hopping time and CPU time are used to evaluate the hopping-time estimation performance of the proposed algorithm.

The frequency estimation accuracy is defined as:

$$R_f = \frac{1}{NNe}\sum_{i=1}^{Ne} Nc_i, \tag{27}$$

where Ne is the total number of trials, and Nc_i is the number of frequencies that were correctly estimated in the ith trial.

The mean square error (MSE) of the hopping-time estimation is defined as:

$$e_t = \frac{1}{2NN_E}\left(\sum_{i=1}^{N_E}\sum_{n=1}^{N}\left(\frac{\hat{l}_{n,i} - l_{n,2}}{l_{n,2}}\right)^2 + \sum_{i=1}^{N_E}\sum_{n=1}^{N}\left(\frac{\hat{l}'_{n,i} - l'_{n,1}}{l'_{n,1}}\right)^2\right), \tag{28}$$

where $\hat{l}_{n,i}$ and $\hat{l}'_{n,i}$ are the estimated values of $l_{n,2}$ and $l'_{n,1}$ in the ith trial, respectively.

4.1. The Effect of Parameters μ and α on Frequency Estimation

The parameter α was set to 5, and the value of parameter μ was selected within the range of $[1.5, 2.4]$ with an interval of 0.1. A total of 100 Monte Carlo tests were conducted under different SNR conditions. The results of the simulations are shown in Figure 8.

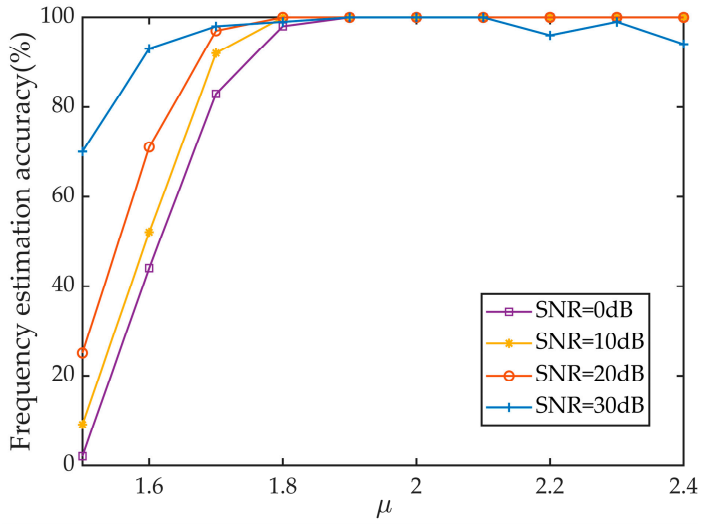

Figure 8. The effect of μ and SNR on the frequency estimation accuracy.

Figure 8 illustrates that when μ is less than 1.9, the frequency estimation accuracy is proportional to μ. When μ is between 1.9 and 2.1 and the SNR is greater than 0 dB, the frequency estimation accuracy can reach 100%; when μ is greater than 2.1 and the SNR is 0 dB, the frequency estimation accuracy starts to decrease. The results show that either μ being set either too high or too low reduced the frequency estimation accuracy. Therefore, the kurtosis threshold (μ) was set to 2.0.

The parameter μ was set to 2.0, and the SNR was set to 0 dB. The parameter α was selected sequentially in the range of $[0, 4]$ at intervals of 1. The p was processed by the zero-setting method and the averaging method, and 100 Monte Carlo tests were conducted. The results are shown in Figure 9.

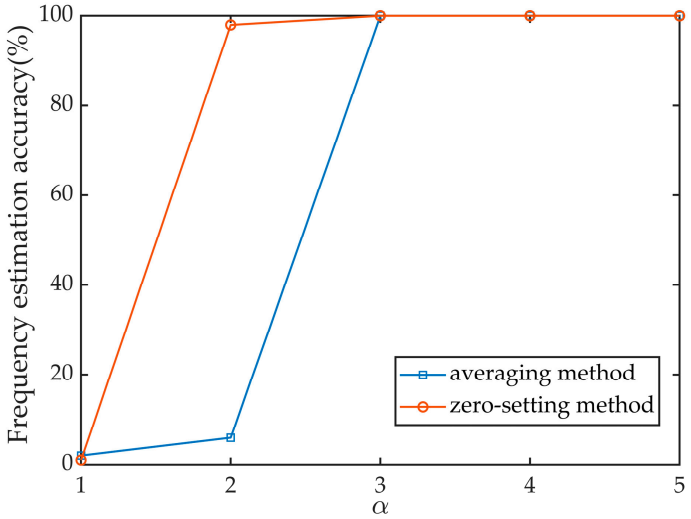

Figure 9. The effect of α on the frequency estimation accuracy.

In Figure 9, the frequency estimation accuracy with $\alpha = 0$ is close to 0, which is consistent with the phenomenon mentioned in Section 3.3. When $\alpha = 3$, the frequency estimation accuracy is 100% for both the averaging method and the zero-setting method. Therefore, in the subsequent simulations, the parameter α was set to 3, and the zero-setting method was chosen to process p.

4.2. The Effect of Parameter c on Hopping-Time Estimation

The parameters of the multiple FH signals are the same as in Section 4.1. The SNR is set to 30 dB, and the parameter c is selected at intervals in the range of $[40, 3200]$. A total of 100 Monte Carlo tests are conducted. The results are shown in Figure 10.

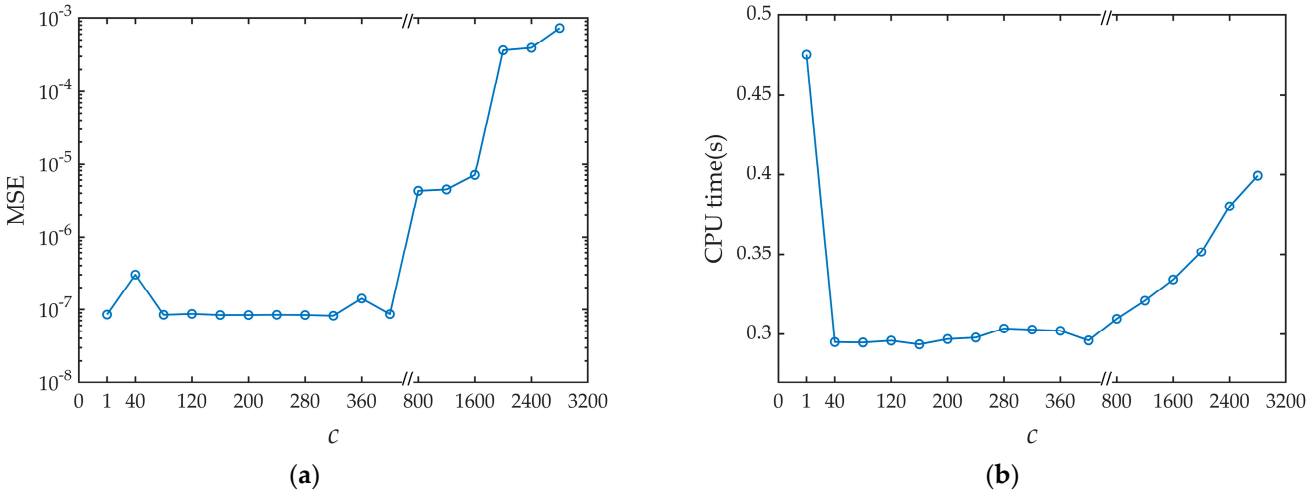

Figure 10. The effect of c on the estimation of hopping time: (**a**) The effect of c on the MSE. (**b**) The effect of c on the CPU time consumed by the hopping-time estimation process.

In Figure 10, the results for $c = 1$ correspond to the performance when the hopping-time fast estimation algorithm is not used, with an MSE equal to 8.5×10^{-8} and a CPU time equal to 0.475 s. When c is greater than 400, the MSE and CPU time of the hopping-time fast estimation algorithm increase with c, and the performance of the fast estimation algorithm gradually degrades. When c is too large, too few sampling points are selected in step 1 of the hopping-time fast estimation algorithm, the hopping time is not between the two sampling points selected in step 2 and the wrong hopping time is obtained. This is the main reason for the increase in MSE when c is too large. When c increases, the computation of step 3 in the hopping-time fast estimation algorithm increases, which leads to an increase in CPU time. When the parameter c is less than 400, the MSE is less than 8.7×10^{-8}, which is the same performance as when the hopping-time fast estimation algorithm is not used. As shown in Figure 10b, the use of the hopping-time fast estimation algorithm leads to a significant decrease in the CPU time consumed during the hopping-time estimation, and the CPU time is the shortest at 0.293 s when parameter c equals 160. Therefore, in the subsequent simulations, parameter c is set to 160.

The sampling frequency selected in this paper is 128 MHz, and the signal fragment length is 3200, which is small enough for most signals to not miss the hopping time. The parameter c obtained at this segment length has a wide applicability.

4.3. Frequency Estimation

A total of 100 Monte Carlo tests were conducted separately under different SNR conditions to compare the ability of the CSML algorithm with that of four other algorithms to estimate frequencies. The results are shown in Figures 11 and 12. The minimum SNR for each algorithm to achieve 100% frequency estimation accuracy and the average CPU time consumed by each algorithm are shown in Table 3.

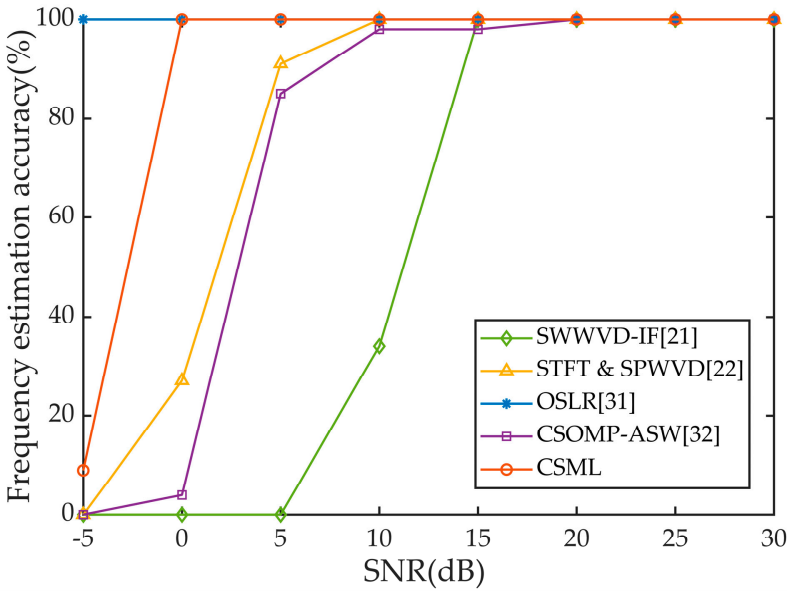

Figure 11. Frequency estimation accuracy of five algorithms under different SNR conditions.

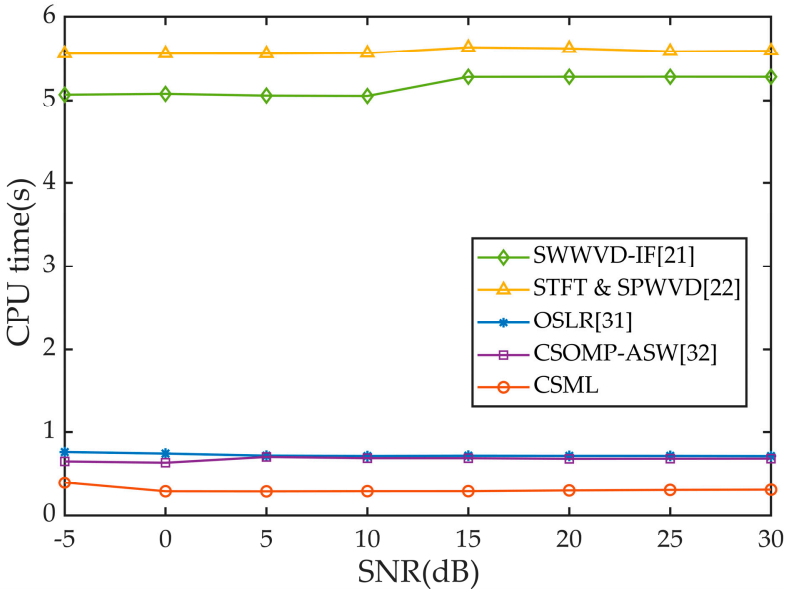

Figure 12. CPU time consumed by the frequency estimation process of the five algorithms under different SNR conditions.

Table 3. The minimum SNR for each algorithm to achieve 100% frequency estimation accuracy and the average CPU time consumed by each algorithm.

Algorithm	OSLR	CSML	CSOMP-ASW	SWWVD-IF	STFT& SPWVD
SNR (dB)	**−5**	0	20	15	10
Average CPU time (s)	0.7277	**0.2984**	0.6796	5.1760	5.5869

The best results are bolded.

In Figure 11, the horizontal axis is the SNR, and the estimation of multiple FH signal parameters is severely affected by noise at 0 dB. The SWWVD-IF algorithm is unable to estimate the frequency of multiple FH signals at an SNR of 0 dB, and the frequency estimation accuracy of this algorithm is only 34% with an SNR of 10 dB. The frequency

estimation accuracy of the CSOMP-ASW algorithm is 4% at an SNR of 0 dB, and the frequency estimation accuracy of this algorithm reaches 85% at an SNR of 10 dB. The STFT&SPWVD algorithm has a frequency estimation accuracy of 27% at an SNR of 0 dB and 91% at an SNR of 10 dB. The OSLR algorithm performs best, but it requires a priori information about the signal. The frequency estimation accuracy of the CSML algorithm proposed in this paper is 100% at an SNR of 0 dB. The CSML algorithm outperforms the CSOMP-ASW, SWWVD-IF and STFT&SPWVD algorithms under low-SNR conditions.

As shown in Figure 12, the CPU time consumed by the five algorithms in estimating the frequency varies less with the SNR. As shown in Table 3, the average CPU times of the OSLR, CSOMP-ASW, SWWVD-IF and STFT&SPWVD algorithms are 0.7277 s, 0.6796 s, 5.176 s and 5.5869 s, respectively. The average CPU time of the CSML algorithm is 0.2984 s. The CSML algorithm consumes the shortest CPU time, which is less than one-half that of the OSLR and CSOMP-ASW algorithms. Therefore, the CSML algorithm proposed in this paper can perform fast and accurate estimation of the frequencies of multiple FH signals without a priori information.

4.4. Hopping-Time Estimation

A total of 100 Monte Carlo tests were conducted separately under different SNR conditions to compare the ability of the CSML algorithm with that of four other algorithms to estimate the hopping time. The results are shown in Figures 13 and 14.

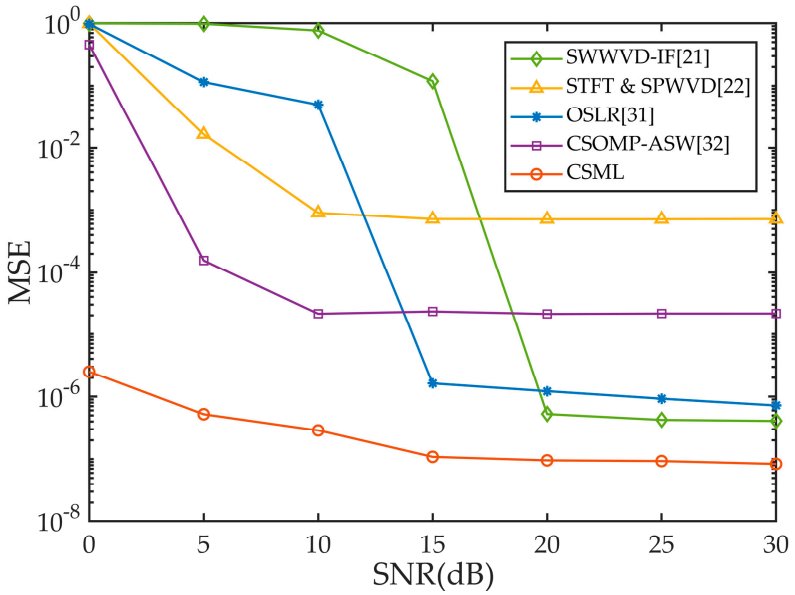

Figure 13. The MSE of five algorithms for the estimation of hopping time.

Figure 13 and Table 4 show the MSEs of the hopping-time estimation of the five algorithms at different SNRs. As the SNR increases, the MSEs of the hopping time estimated by the five algorithms decreases, finally converging to different values. When the SNR is greater than 10 dB, the MSEs of the CSOMP-ASW and STFT&SPWVD algorithms are basically unchanged at about 2.13×10^{-5} and 7.15×10^{-4}, respectively. When the SNR is greater than 15 dB, the MSE of the OSLR algorithm is less than 1.62×10^{-6}, and that of the CSML algorithm is less than 1.086×10^{-7}. When the SNR is greater than 20 dB, the MSE of the SWWVD-IF algorithm is less than 5.259×10^{-7}. Under the same SNR condition, the MSE of the CSML algorithm is smaller than that of the other four algorithms.

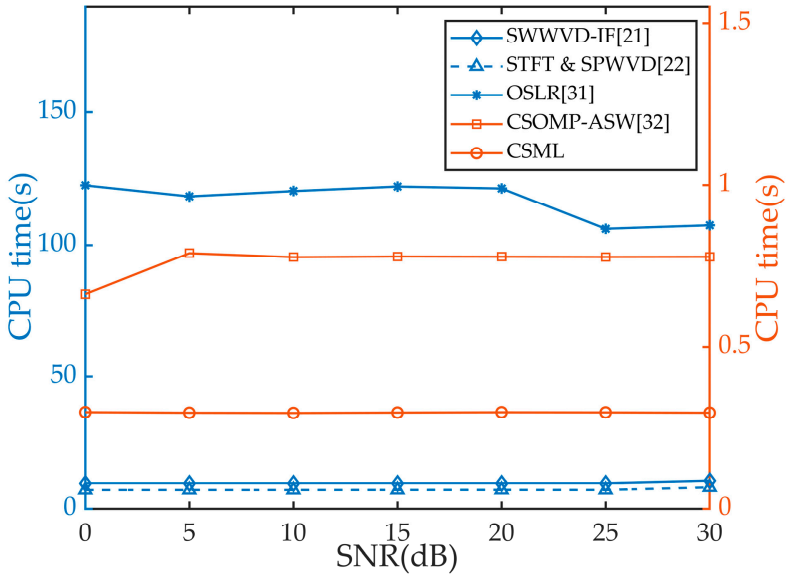

Figure 14. CPU time consumed by the five algorithms in the estimation process of hopping time.

Table 4. The MSE of the hopping-time estimation of the five algorithms under different SNR conditions.

Algorithm	SNR = 0 dB	SNR = 5 dB	SNR = 10 dB	SNR = 15 dB	SNR = 20 dB	SNR = 25 dB	SNR = 30 dB
OSLR	0.973	0.115	0.050	1.62×10^{-6}	1.218×10^{-6}	9.246×10^{-7}	7.212×10^{-7}
CSML	**2.502×10^{-6}**	**5.223×10^{-7}**	**2.879×10^{-7}**	**1.086×10^{-7}**	**9.545×10^{-8}**	**9.298×10^{-8}**	**8.372×10^{-8}**
CSOMP-ASW	0.459	1.557×10^{-4}	2.130×10^{-5}	2.318×10^{-5}	2.115×10^{-5}	2.147×10^{-5}	2.148×10^{-5}
SWWVD-IF	0.996	0.980	0.765	0.119	5.259×10^{-7}	4.245×10^{-7}	4.082×10^{-7}
STFT&SPWVD	0.986	0.0166	8.988×10^{-4}	7.151×10^{-4}	7.115×10^{-4}	7.114×10^{-4}	7.144×10^{-4}

The best results are bolded.

Figure 14 and Table 5 reflect the CPU time consumed by the five algorithms. The estimation of hopping time by all five algorithms is based on the estimation of frequency. Therefore, the CPU time in this section includes the CPU time consumed during frequency estimation. The CPU time consumed by the OSLR algorithm fluctuates widely, with a minimum of 106.1 s and a maximum of 122.4 s. The CPU time consumed by the remaining four algorithms fluctuates very little. The average CPU times of the OSLR, CSOMP-ASW, SWWVD-IF and STFT&SPWVD algorithms are 116.87 s, 0.7646 s, 9.5659 s and 7.1659 s, respectively. The average CPU time of the CSML algorithm is 0.3088 s. Compared with the other four algorithms, the CSML algorithm can estimate the hopping time of multiple FH signals faster and more accurately.

Table 5. The average CPU time consumed by the five algorithms in the process of estimating the hopping time.

Algorithm	OSLR	CSML	CSOMP-ASW	SWWVD-IF	STFT& SPWVD
Average CPU time (s)	116.87	**0.3088**	0.7646	9.5659	7.1659

The best results are bolded.

4.5. Parameter Estimation Performance of the CSML Algorithm for Multiple FH Signals with Frequency-Switching Time

The parameters of the multiple FH signals used for the simulations in this section are shown in Table 6, and a schematic diagram of this signal is shown in Figure 15. The starting

frequency of FH signal No. 1 ($n = 1$) is 12 MHz; it enters the frequency-switching time at $l = 35,001$, with a frequency-switching time of 1004, and the frequency becomes 24 MHz at $l = 36,005$. The starting frequency of FH signal No. 2 ($n = 2$) is 54 MHz; it enters the frequency-switching time at $l = 26,001$, with a frequency-switching time of 4000, and the frequency changes to 60 MHz at $l = 30,001$. The third signal is a fixed-frequency signal with a frequency of 18 MHz. Other necessary parameters are shown in Table 2.

Table 6. Parameters of multiple FH signals with frequency-switching time.

n	$f_{n,1}$ (MHz)	$f_{n,2}$ (MHz)	$l'_{n,1}$	$\Delta_{n,1}$	$l_{n,2}$
$n = 1$	12	24	35,001	1004	36,005
$n = 2$	54	60	26,001	4000	30,001
$n = 3$	18		51,200		

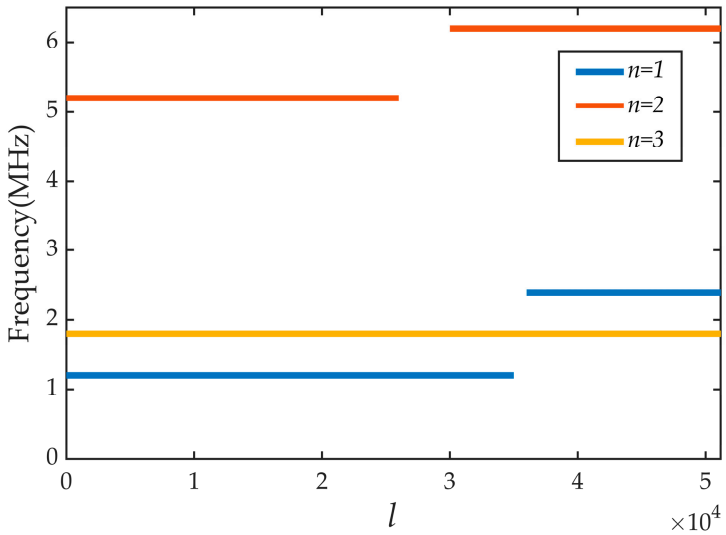

Figure 15. Schematic diagram of multiple FH signals with frequency-switching time.

The OSLR, SWWVD-IF and STFT&SPWVD algorithms cannot estimate the parameters of the FH signals with frequency-switching time. In this section, only the CSML algorithm and the CSOMP-ASW algorithm are simulated. The results were shown in Figures 16 and 17.

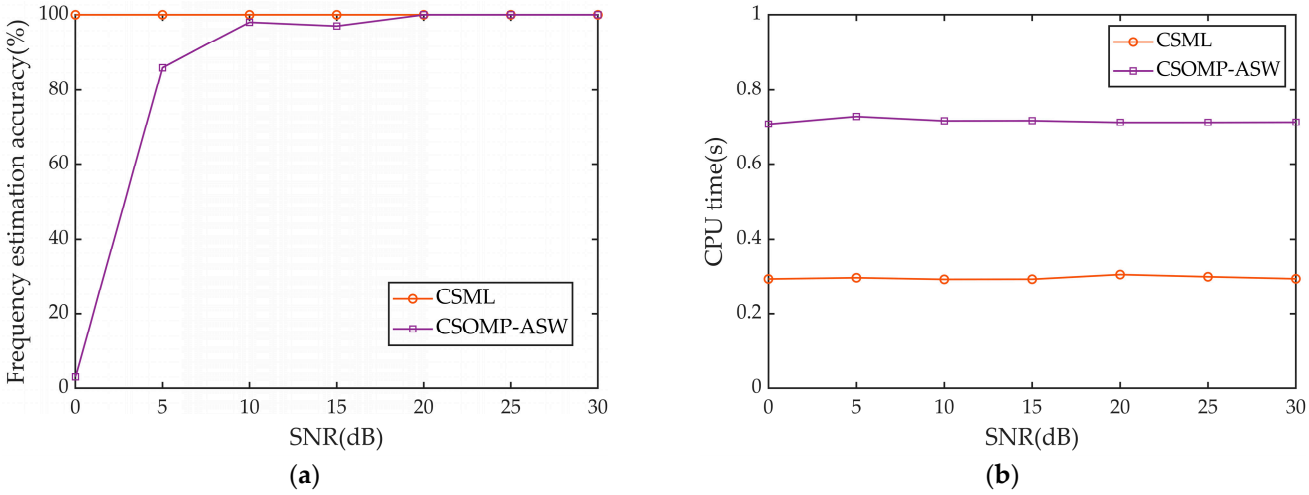

Figure 16. Performance of the two algorithms when estimating the frequency. (**a**) The frequency estimation accuracy of the two algorithms. (**b**) CPU time consumed by the two algorithms.

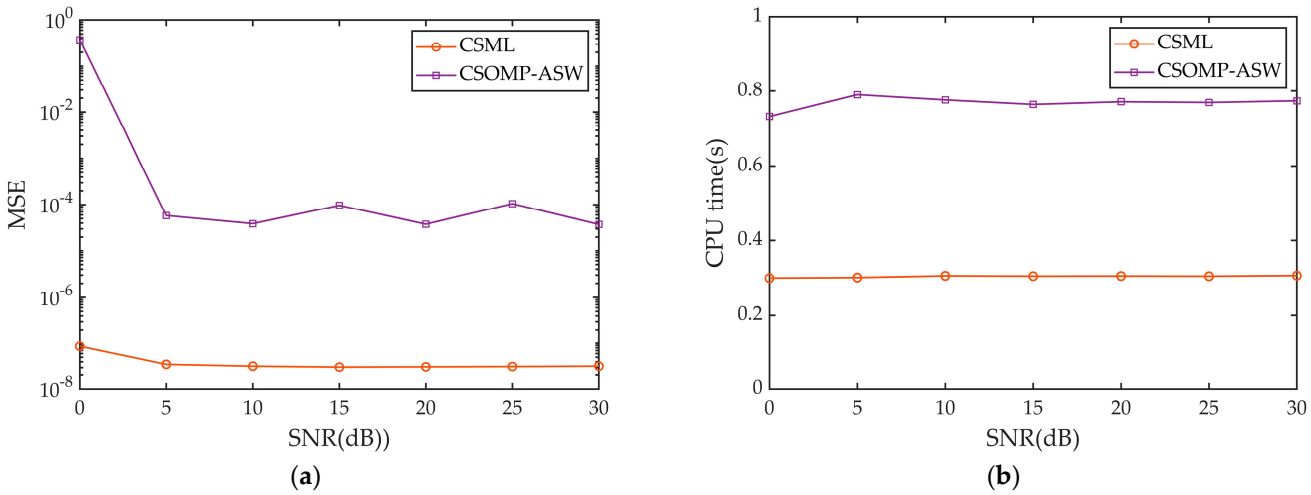

Figure 17. Performance of two algorithms when estimating hopping time. (**a**) The MSEs of the two algorithms for hopping-time estimation. (**b**) CPU time consumed by both algorithms in the process of estimating the hopping time.

Figure 16a shows the correct rates of the CSML algorithm and the CSOMP-ASW algorithm for frequency estimation of multiple FH signals with frequency-switching time. When the SNR is greater than 0 dB, the frequency estimation accuracy of the CSML algorithm is 100%; when the SNR is greater than 10 dB, the frequency estimation accuracy of the CSOMP-ASW algorithm is greater than 98%; when the SNR is greater than 20 dB, the frequency estimation accuracy of the CSOMP-ASW algorithm is equal to 100%. As shown in Figure 16b, the average CPU time consumed by the CSOMP-ASW algorithm is 0.7159 s, while that of the CSML algorithm is 0.2956 s.

Figure 17a shows the MSE of the estimated hopping time for the CSML algorithm and the CSOMP-ASW algorithm when estimating hopping time for multiple FH signals with frequency-switching time. When the SNR is greater than 5 dB, the MSE of the CSML algorithm is less than 3.514×10^{-8}, and the MSE of the CSOMP-ASW algorithm is less than 5.811×10^{-5}. As shown in Figure 17b, the average CPU time consumed by the CSOMP-ASW algorithm is 0.768 s, while that of the CSML algorithm is 0.3032 s. Therefore, the CSML algorithm can perform fast and accurate estimation of the parameters of multiple FH signals with frequency-switching time.

5. Conclusions

In this paper, a CSML algorithm based on compressed spectrum sensing and maximum likelihood theory is proposed for parameter estimation of multiple FH signals with frequency-switching time when the number of signals is unknown. For the problem of low accuracy when compressed spectrum sensing is used for frequency estimation alone, the kurtosis threshold and zero-setting methods are applied to improve the accuracy of frequency estimation. As for the problem of high computational demand and error in hopping-time estimation, a hopping-time fast estimation algorithm based on maximum likelihood theory is proposed. Simulation results show that the CSML algorithm proposed in this paper achieves higher accuracy and significantly reduced computation time compared with existing algorithms. The estimation of the hopping time in the compressed domain further reduces the computational effort. In future work, we hope to explore the implementation of hopping-time estimation algorithms in the compressed domain. In this this paper, we only considered a case in which the spectra of multiple FH signals do not overlap. When the spectra overlap, the estimation of the frequency and hopping time of the signals is more difficult. Therefore, in future work, we will use more advanced mathematical tools and algorithms to solve the problem of overlapping spectra, such as image

processing and machine-learning-based methods, to improve the accuracy and efficiency of frequency and hopping-time estimation.

Author Contributions: Conceptualization, Y.L. (Yixing Li) and F.W.; methodology, Y.L. (Yixing Li); investigation, G.F. and Y.L. (Yixing Li); writing—original draft preparation, Y.L. (Yixing Li); writing—review and editing, Y.Z. and Y.L. (Yang Liu) All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Shanxi Graduate Education Innovation Project under Grant No. 2022Y575 and the Shanxi Scholarship Council of China (2020-108).

Data Availability Statement: Data is contained within the article or supplementary material. The data presented in this study are available in Tables 1–6.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Because (10) and (11) have similar forms, we only consider (10) and minimize it.

To minimize (10), we set $\partial\varphi_{n,1}/\partial a_{n,1} = 0$. Therefore, the estimated value of $a_{n,1}$ can be expressed as:

$$\hat{a}_{n,1} = \frac{1}{l'_{n,1}}\mathbf{s}_{n,1}^{H}\mathbf{x}_{n,1}. \tag{A1}$$

Substituting (A1) into (10), $\varphi_{n,1}(a_{n,1}, f_{n,1}, l'_{n,1})$ becomes:

$$\begin{aligned}\varphi_{n,1}(f_{n,1}, l'_{,1}) &= \varphi_{n,1}(\hat{a}_{n,1}, f_{n,1}, l'_{n,1}) \\ &= \left\|\mathbf{x}_{n,1} - \tfrac{1}{l'_{n,1}}\mathbf{s}_{n,1}^{H}\mathbf{x}_{n,1}\mathbf{s}_{n,1}\right\|^2 \\ &= \left\|\mathbf{x}_{n,1}\right\|^2 - \tfrac{\left\|\mathbf{s}_{n,1}^{H}\mathbf{x}_{n,1}\right\|^2}{l'_{n,1}}.\end{aligned} \tag{A2}$$

Because $\left\|\mathbf{x}_{n,1}\right\|^2$ and $\left\|\mathbf{s}_{n,1}^{H}\mathbf{x}_{n,1}\right\|^2/l'_{n,1}$ of (A2) are positive, minimizing (A2) is equivalent to maximizing (A3).

$$\overline{\varphi}_{n,1}(f_{n,1}, l'_{n,1}) = \frac{\left\|\mathbf{s}_{n,1}^{H}\mathbf{x}_{n,1}\right\|^2}{l'_{n,1}}. \tag{A3}$$

Suppose that $\hat{f}_{n,1}$ is known; substituting it into (A3), the objective function becomes:

$$\overline{\varphi}_{n,1}(l'_{n,1}) = \frac{\left\|\hat{\mathbf{s}}_{n,1}^{H}\mathbf{x}_{n,1}\right\|^2}{l'_{n,1}}, \tag{A4}$$

where:

$$\hat{\mathbf{s}}_{n,1} = \left[e^{j2\pi\hat{f}_{n,1}/f_s}, \ldots, e^{j2\pi\hat{f}_{n,1}l'_{n,1}/f_s}\right]^{\mathrm{T}}. \tag{A5}$$

Therefore, minimizing (10) is equivalent to maximizing (A4).

When $\hat{f}_{n,2}$ is known, minimizing (11) is equivalent to maximizing (A6).

$$\overline{\varphi}_{n,2}(l_{n,2}) = \frac{\left\|\hat{\mathbf{s}}_{n,2}^{H}\mathbf{x}_{n,2}\right\|^2}{L - l_{n,2} + 1}, \tag{A6}$$

where:

$$\hat{\mathbf{s}}_{n,2} = \left[e^{j2\pi\hat{f}_{n,2}l_{n,2}/f_s}, \ldots, e^{j2\pi\hat{f}_{n,2}L/f_s}\right]^{\mathrm{T}}. \tag{A7}$$

Because this paper only considers a condition in which the spectrum of each signal does not overlap, (A4) and (A6) can be rewritten as follows:

$$\begin{aligned} \overline{\varphi}_{n,1}(l'_{n,1}) &= \frac{\|\hat{\mathbf{s}}^H{}_{n,1}\mathbf{X}_1\|^2}{l'_{n,1}}, \\ \overline{\varphi}_{n,2}(l_{n,2}) &= \frac{\|\hat{\mathbf{s}}^H{}_{n,2}\mathbf{X}_2\|^2}{L-l_{n,2}+1}, \end{aligned} \tag{A8}$$

where:

$$\begin{aligned} \mathbf{X}_1 &= [x(1), \ldots, x(l'_{n,1})], \\ \mathbf{X}_2 &= [x(l_{n,2}), \ldots, x(L)]. \end{aligned} \tag{A9}$$

References

1. Feng, Q.; Zhang, J.; Chen, L.; Liu, F. Waveform Reconstruction of DSSS Signal Based on VAE-GAN. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, e3667592. [CrossRef]
2. Nie, R.; Li, B. Detection and Simulation of Quasi Random Frequency Hopping Signal Based on Interference Analysis Algorithm. *Neural Comput. Applic.* **2023**, *35*, 8847–8858. [CrossRef]
3. Lei, Z.; Yang, P.; Zheng, L.; Xiong, H.; Ding, H. Frequency Hopping Signals Tracking and Sorting Based on Dynamic Programming Modulated Wideband Converters. *Appl. Sci.* **2019**, *9*, 2906. [CrossRef]
4. Kang, J.; Shin, Y.; Lee, H.; Park, J.; Lee, H. Radio Frequency Fingerprinting for Frequency Hopping Emitter Identification. *Appl. Sci.* **2021**, *11*, 10812. [CrossRef]
5. Zhu, J.; Wang, A.; Wu, W.; Zhao, Z.; Xu, Y.; Lei, R.; Yue, K. Deep-Learning-Based Recovery of Frequency-Hopping Sequences for Anti-Jamming Applications. *Electronics* **2023**, *12*, 496. [CrossRef]
6. Lei, Z.; Yang, P.; Zheng, L. Detection and Frequency Estimation of Frequency Hopping Spread Spectrum Signals Based on Channelized Modulated Wideband Converters. *Electronics* **2018**, *7*, 170. [CrossRef]
7. Wei, S.; Zhang, M.; Wang, G.; Sun, X.; Zhang, L.; Chen, D. Robust Multi-Frame Joint Frequency Hopping Radar Waveform Parameters Estimation Under Low Signal-Noise-Ratio. *IEEE Access* **2019**, *7*, 177198–177210. [CrossRef]
8. Wan, J.; Zhang, D.; Xu, W.; Guo, Q. Parameter Estimation of Multi Frequency Hopping Signals Based on Space-Time-Frequency Distribution. *Symmetry* **2019**, *11*, 648. [CrossRef]
9. Zhu, H.; Lv, H.; Dai, Z.; Tan, M.; Song, W. A Novel Parameter Estimation Method of FHSS Signal with Low SNR. *IEEJ Trans. Electr. Electron. Eng.* **2023**. [CrossRef]
10. Wang, L.; Liu, Z.; Feng, Y.; Liu, X.; Xu, X.; Chen, X. A Parameter Estimation Method for Time-Frequency-Overlapped Frequency Hopping Signals Based on Sparse Linear Regression and Quadratic Envelope Optimization. *Int. J. Commun. Syst.* **2020**, *33*, e4463. [CrossRef]
11. Su, Y.; Wang, L.; Chen, Y.; Yang, X. ℓp-STFT: A Robust Parameter Estimator of a Frequency Hopping Signal for Impulsive Noise. *Electronics* **2019**, *8*, 1017. [CrossRef]
12. Li, G.; Wang, W.; Ding, G.; Wu, Q.; Liu, Z. Frequency-Hopping Frequency Reconnaissance and Prediction for Non-Cooperative Communication Network. *China Commun.* **2021**, *18*, 51–64. [CrossRef]
13. Li, L.; Cai, H.; Han, H.; Jiang, Q.; Ji, H. Adaptive Short-Time Fourier Transform and Synchrosqueezing Transform for Non-Stationary Signal Separation. *Signal Process.* **2020**, *166*, 107231. [CrossRef]
14. Wang, Y.; He, S.; Wang, C.; Li, Z.; Li, J.; Dai, H.; Xie, J. Detection and Parameter Estimation of Frequency Hopping Signal Based on the Deep Neural Network. *Int. J. Electron.* **2022**, *109*, 520–536. [CrossRef]
15. Fu, W.; Li, X.; Liu, N.; Hei, Y.; Wei, J. Parameter Blind Estimation of Frequency-Hopping Signal Based on Time–Frequency Diagram Modification. *Wirel. Pers. Commun.* **2017**, *97*, 3979–3992. [CrossRef]
16. Ren, H.; Ren, A.; Li, Z. A New Strategy for the Suppression of Cross-Terms in Pseudo Wigner–Ville Distribution. *SIViP* **2016**, *10*, 139–144. [CrossRef]
17. Lee, K.-G.; Oh, S.-J. Detection of Frequency-Hopping Signals With Deep Learning. *IEEE Commun. Lett.* **2020**, *24*, 1042–1046. [CrossRef]
18. Wang, H.; Zhang, B.; Wang, H.; Wu, B.; Guo, D. Parameter Estimation of Multiple Frequency-Hopping Signals Based on Space-Time-Frequency Analysis by Atomic Norm Soft Thresholding with Missing Observations. *China Commun.* **2022**, *19*, 135–151. [CrossRef]
19. Gaikwad, C.J.; Sircar, P. Bispectrum-Based Technique to Remove Cross-Terms in Quadratic Systems and Wigner–Ville Distribution. *SIViP* **2018**, *12*, 703–710. [CrossRef]
20. Liu, S.; Zhang, Y.D.; Shan, T.; Tao, R. Structure-Aware Bayesian Compressive Sensing for Frequency-Hopping Spectrum Estimation With Missing Observations. *IEEE Trans. Signal Process.* **2018**, *66*, 2153–2166. [CrossRef]
21. Kanaa, A.; Sha'ameri, A.Z. A Robust Parameter Estimation of FHSS Signals Using Time–Frequency Analysis in a Non-Cooperative Environment. *Phys. Commun.* **2018**, *26*, 9–20. [CrossRef]

22. Zhang, D.; Shang, Y.; Liang, X.; Lin, J. Efficient Blind Estimation of Parameters for Multiple Frequency Hopping Signals via Single Channel. In Proceedings of the 2022 IEEE 5th International Conference on Automation, Electronics and Electrical Engineering (AUTEEE), Shenyang, China, 18–20 November 2022; pp. 1063–1069.
23. Liu, F.; Jiang, Y. Knowledge-Enhanced Compressed Measurements for Detection of Frequency-Hopping Spread Spectrum Signals Based on Task-Specific Information and Deep Neural Networks. *Entropy* **2022**, *25*, 11. [CrossRef] [PubMed]
24. Wei, Z.; Zhang, J.; Xu, Z.; Liu, Y.; Okarma, K. Measurement Matrix Optimization via Mutual Coherence Minimization for Compressively Sensed Signals Reconstruction. *Math. Probl. Eng.* **2020**, *2020*, 7979606. [CrossRef]
25. Liu, Z.; Li, L.; Lv, D.; Pan, N. Novel Source Recovery Method of Underdetermined Time-Frequency Overlapped Signals Based on Submatrix Transformation and Multi-Source Point Compensation. *IEEE Access* **2019**, *7*, 29610–29622. [CrossRef]
26. Angelosante, D.; Giannakis, G.B.; Sidiropoulos, N.D. Sparse Parametric Models for Robust Nonstationary Signal Analysis: Leveraging the Power of Sparse Regression. *IEEE Signal Process. Mag.* **2013**, *30*, 64–73. [CrossRef]
27. Zhang, X.; Hu, X.; Dong, X. A Joint Algorithm of Parameters Estimation for Frequency-Hopping Signal Based on Sparse Recovery. In Proceedings of the 2017 9th International Conference on Wireless Communications and Signal Processing (WCSP), Nanjing, China, 11–13 October 2017; pp. 1–5.
28. Liu, H.; Yao, T.; Li, R.; Ye, Y. Folded Concave Penalized Sparse Linear Regression: Sparsity, Statistical Performance, and Algorithmic Theory for Local Solutions. *Math. Program.* **2017**, *166*, 207–240. [CrossRef] [PubMed]
29. Zhang, Z.; Rao, B.D. Sparse Signal Recovery With Temporally Correlated Source Vectors Using Sparse Bayesian Learning. *IEEE J. Sel. Top. Signal Process.* **2011**, *5*, 912–926. [CrossRef]
30. Zhu, J.; Han, L.; Meng, X. An AMP-Based Low Complexity Generalized Sparse Bayesian Learning Algorithm. *IEEE Access* **2019**, *7*, 7965–7976. [CrossRef]
31. Wang, Y.; Zhang, C.; Jing, F. Frequency-Hopping Signal Parameters Estimation Based on Orthogonal Matching Pursuit and Sparse Linear Regression. *IEEE Access* **2018**, *6*, 54310–54319. [CrossRef]
32. Fu, W.; Jiang, T. A Parameter Estimation Algorithm for Multiple Frequency-Hopping Signals Based on Compressed Sensing. *Phys. Commun.* **2019**, *37*, 100892. [CrossRef]
33. Ko, C.C.; Zhi, W.; Chin, F. ML-Based Frequency Estimation and Synchronization of Frequency Hopping Signals. *IEEE Trans. Signal Process.* **2005**, *53*, 403–410. [CrossRef]
34. Li, Y.; Wang, F. Parameter Estimation of Frequency Hopping Signals Based on Maximum Likelihood and Orthogonal Matching Pursuit. In *International Conference on Autonomous Unmanned Systems, Proceedings of the 2022 International Conference on Autonomous Unmanned Systems (ICAUS 2022), Xi'an, China, 23–25 September 2022*; Fu, W., Gu, M., Niu, Y., Eds.; Springer Nature: Singapore, 2023; pp. 3397–3406.
35. Hu, C.; Kim, J.Y.; Na, S.Y.; Kim, H.-G.; Choi, S.-H. Compressive Frequency Hopping Signal Detection Using Spectral Kurtosis and Residual Signals. *Wirel. Pers. Commun.* **2017**, *94*, 53–67. [CrossRef]