A Time-Domain Model-Based Method for the Identification of Multi-frequency Signal Parameters

Ya Guo* and Jinglu Tan
Department of Bioengineering, University of Missouri, Columbia, MO, 65211, USA

* Corresponding Author, Email: guoy@missouri.edu
Phone: (573)-882-5418
Fax: (573)-884-5650
Abstract:

Discrete Fourier Transform (DFT) is frequently used to determine parameters like frequency, amplitude, and phase; however, the finite length of data being used often leads to errors. Modeling a signal process in time-domain and estimating model parameters from observed data will provide a prediction about the data outside the sampled window and thus may lead to a better estimation of the signal spectrum. Many models were developed in literature for this purpose. While these models can provide better spectrum estimation than DFT, they may not be the best option because the parameter estimation is only suboptimal of least squares or the methods have very narrow convergence area. In this paper, the traditional nonlinear separable estimation has been modified and a new time-domain least-squares method was developed for the identification of multi-frequency signal parameters. Algorithm convergence was analyzed and improved through segmentation of data. Simulations and an experiment were provided to validate the effectiveness of the developed method.

Keywords: Least-squares method; Signal processing; Parameter identification; Fourier transform
1. Introduction

In the fields of measurement, control, and communication, spectral analysis is often used to obtain important parameters such as frequency composition, magnitude, and phase. A sampled signal is usually transformed from the time domain to the frequency domain through DFT (Discrete Fourier Transform) [1]. In reality, it is impossible to use infinitely long data for DFT or synchronize the sampling rate with all the components of the signal. Windowing the data assumes that the unobserved data outside the window periodically repeat those inside the window. This leads to the well-known granularity effect (also known as leakage and picket-fence effects) [2, 3], which produces errors with the computed DFT spectrum. Some correction methods in the frequency domain have been developed to improve DFT spectrum [4-11]. An overview of different frequency domain methods can be found in [12]. Approximations are often applied in these methods to simplify the equations or computation, which unavoidably affect the estimation accuracy. Another effort of estimating parameters of multipath channels through the theory of linear time-variant system can be found in Sanz-González et al. [13]. Since the frequency-domain spectra of a signal originate from the time-domain data, the time-domain data should contain all the information of frequency, amplitude, and phase. This suggests that estimating signal parameters from time domain directly may be more precise.

Modeling a signal process in time-domain and estimating model parameters from the observed data will provide a prediction about the data outside the sampled window. This may lead to a better estimation of the signal spectrum. Many models were developed in literature for this purpose. Some of these models are rational-transfer-function-based
models, such as AR (Autoregressive) model, MA (Moving Average) model, ARMA
(Autoregressive Moving Average) model, and their variants like Maximum Entropy
method [14]. This type of model is not based on the summation of Sine (or Cosine)
functions and the estimated model parameters should be fed into some equations to
compute spectrum parameters. The solution for those spectrum parameters may not be
the best least-squares solution for the signal. There are also some problems with these
models including spectral line splitting [15], high sensitivity to noise [16], no phase
information, and suboptimal techniques employed for model parameters estimation [14].
Prony’s model (Prony spectral line estimation) is based on the summation of Sine (or
Cosine) functions [17] and its excellent performance on spectrum estimation has been
demonstrated by Kay and Marple [14]. However, to avoid computational difficulty,
Prony’s method does not optimize the total least-squares errors and is also only a
suboptimal minimization of least squares [14]. The traditional nonlinear separable
estimation method [18-20] is truly least-squares based and tries to optimize the total
least-squares errors, but its convergence is negatively correlated to data length. More data
are critical for improving estimation accuracy under noise conditions. The dilemma
between convergence and estimation accuracy limits the application of the traditional
nonlinear separable estimation method. Detailed comparison on model-based spectrum
estimation can be found in [14, 21]. While these models can provide better spectrum
estimation than DFT, they may not be the best option because of either the random model
structures or the suboptimal minimization of least squares or very narrow convergence
area, in which, an initialization will lead to the global optimum by iteratively adjusting
parameter increments according to errors between experimental data and model predictions.

In this work, we modified the traditional separable estimation method and developed a new time-domain least-squares method for the identification of multi-frequency signal parameters. The model structure remains the feature of the traditional separable estimation method. Segmentation of the data is used to improve algorithm convergence. After the algorithm is presented, results from simulations and from a real experiment are used to show that the developed method can improve the identified parameter accuracy compared with the methods reported in literature and can improve algorithm convergence compared to the traditional separable estimation method.

2. Formulation for a time-domain least-squares algorithm

2.1. Parameterization in the Traditional Separable Estimation Method

Assume that the sampled experimental data of a multi-frequency signal is denoted as \( \bar{x}(k\Delta t) \), \( k=0, 1, \ldots, N-1 \), where \( N \) is the number of data points, \( \Delta t \) is the sampling interval and satisfies Nyquist-Shannon’s sampling theorem. If the multi-frequency signal contains \( M \) components, it can be represented as:

\[
x(k\Delta t) = \sum_{m=1}^{M} x_m(k\Delta t) = \sum_{m=1}^{M} A_m \cos(2\pi f_m k\Delta t + \phi_m), \quad k=0, 1, \ldots, N-1
\]

where \( A_m, f_m, \) and \( \phi_m \) are the amplitude, frequency, and phase of the \( m^{th} \) \( (m=1 \) to \( M \) \) component \( x_m \).

The unknowns \( A_m, f_m, \) and \( \phi_m \) are to be identified from the measured data \( \bar{x}(k\Delta t) \). To do this, a least-squares-based method could be used to minimize the error between the
measured data \( \bar{x} \) and the predictions \( x \) from Eqn (1) by adjusting \( A_m, f_m, \) and \( \phi_m, \)
where \( \bar{x} = [\bar{x}(0) \quad \bar{x}(\Delta t) \quad \cdots \quad \bar{x}((N-1)\Delta t)]^T \) and \( x = [x(0) \quad x(\Delta t) \quad \cdots \quad x((N-1)\Delta t)]^T. \)

For a given set of \( A_m, f_m, \) and \( \phi_m, \) the sum of squared errors between \( \bar{x} \) and \( x \) is:
\[
e = \| \bar{x} - x \|^2 \tag{2}
\]
e thus depends on \( A_m, f_m, \) and \( \phi_m. \) To minimize \( e, \) \( 3M \) parameters should be optimized.
Too many independent model parameters usually result in slow computation speed and difficulty in converging to the global optimal point. It is thus desirable to reformat Eqn (2) to reduce independent model parameters if allowable.

Eqn (1) can be written as
\[
x(k\Delta t) = \sum_{m=1}^{M} A_m [\cos(2\pi f_m k\Delta t)\cos \phi_m - \sin(2\pi f_m k\Delta t)\sin \phi_m] \\
= \sum_{m=1}^{M} [a_m \cos(2\pi f_m k\Delta t) + b_m \sin(2\pi f_m k\Delta t)]
\tag{3}
\]
where \( a_m = A_m \cos \phi_m, \quad b_m = -A_m \sin \phi_m, \) \( (m=1 \text{ to } M). \)

According to [18-20], minimization of error \( e \) in Eqn (2) is the same as finding an optimal solution of \( \varphi \) for
\[
\Psi \varphi = \bar{x} \tag{4}
\]
where
\[
\Psi = \begin{bmatrix}
\cos(2\pi f_1 \Delta t) & \sin(2\pi f_1 \Delta t) & \cdots & \cos(2\pi f_M \Delta t) & \sin(2\pi f_M \Delta t) \\
\cos(2\pi f_1 (N-1)\Delta t) & \sin(2\pi f_1 (N-1)\Delta t) & \cdots & \cos(2\pi f_M (N-1)\Delta t) & \sin(2\pi f_M (N-1)\Delta t)
\end{bmatrix} \tag{5}
\]
and
\[
\varphi = [a_1 \quad b_1 \quad \cdots \quad a_M \quad b_M]^T \tag{6}
\]
Usually, the number of sampled data points \( N \) is much larger than the number of components \( M \). For a given frequency vector \( f = [f_1, f_2, \ldots, f_M] \), the least-squares solution for \( \varphi \) is:

\[
\varphi = (\Psi^T \Psi)^{-1} \Psi^T \bar{x}
\]

The sum of squared errors \( e \) in Eqn (2) can thus be expressed as:

\[
e = \|\bar{x} - \Psi \varphi\|^2 = \|\bar{x} - \Psi (\Psi^T \Psi)^{-1} \Psi^T \bar{x}\|^2
\]

The total squared error expressed by Eqn (8) is a function of only frequency \( f \). A least-squares algorithm could be used to determine the frequencies that minimize \( e \) in Eqn (8). With the frequencies determined, the amplitude and phase of the corresponding harmonic component can be determined as:

\[
A_m = \sqrt{a_m^2 + b_m^2}
\]

\[
\phi_m = \arctan(-b_m / a_m)
\]

The sign of \( b_m \) and \( a_m \) should be considered to guarantee that the calculated \( \phi_m \) does not differ from the real value by \( \pi \).

2.2. Influence of Data Length on Algorithm Convergence

Signal-to-noise ratio (SNR), the ratio of signal power to the noise power, affects the accuracy of estimated model parameters. In this work, all the SNRs are in dB, which are computed as:

\[
SNR = 10 \log \frac{P_{signal}}{P_{noise}}
\]

where \( P_{signal} \) is the average power of the signal, \( P_{noise} \) is the average power of the noise.
For a time-domain least squares method, more data can mitigate the effects of the noise. From this point of view, longer data length is desirable; however, longer data length will result in narrow convergence area, which is explained and demonstrated below. In general, harmonic data series of the same length but different frequencies are not fully orthogonal to one another. This results in spectral leakage. The leakage makes the total least squares error $e$ in Eqn (8) change with frequency, which is the base for least squares algorithm adjusting model parameters. Usually, the longer the data are, the better the orthogonality, the less the DFT spectral energy leakage. The better orthogonality or frequency resolution resulting from longer data, however, will make error $e$ monotonically change with $f$ only in a small region near the real frequency as demonstrated in Figure 1. Therefore, the optimization of $f$ through a least-squares algorithm may be easily trapped into local minima, thus deteriorating the convergence. In Figure 1, the signal is $x = 10 \cos(2\pi 100t + \pi / 9)$ Hz sampled at 1024 Hz. The least-squares error $e$ was computed with Eqn (8) for different given frequencies. The data segment lengths used were 1024, 256, 64, and 8. The error for each segment length was normalized by using its maximum. The local minima in Figure 1 determine the range of frequency initializations $f_0$ that will lead to convergence to the global optimum.
Figure 1. Normalized least-squares error as a function of frequency for different data segment lengths. The longer data length makes error $e$ monotonically change with $f$ only in a small region.

Figure 1 shows that least-squares algorithm can converge to the global optimum point (100 Hz) with a larger range of initial $f$ values when a shorter data length is used. Monte Carlo simulations were performed to validate this conclusion. The initial estimates $f_0$ does not affect the accuracy of achieved frequencies estimates as long as the least-squares based algorithm can converge to the global minima.

2.3. Algorithm Development

Obviously, data length has opposite effect on algorithm convergence and robustness of parameter estimates. Fortunately, this problem can be solved through reformatting Eqn (8). Based on the above analysis, Eqn (8) should be modified to improve algorithm convergence while taking advantage of the full data set. In this developed method, the sampled data set is divided into short segments with the same length as done in conventional DFT or FFT algorithms. Some segments may overlap if total data length is
not an integer multiple of the segment length. If the data are split into \( P \) segments of \( Q \) data points \( (Q \geq M) \), the total squared error for segment \( p \) is

\[
e_p = \left\| \bar{x}_p - \Psi_p \left( \Psi_p^T \Psi_p \right)^{-1} \Psi_p^T \bar{x}_p \right\|^2 \quad (p=1 \text{ to } P)
\]

where \( \bar{x}_p \) is the data vector of the \( p^{th} \) segment, and

\[
\Psi_p = \begin{bmatrix}
\cos(2\pi f_{t_1}) & \sin(2\pi f_{t_1}) & \cdots & \cos(2\pi f_{t_P}) & \sin(2\pi f_{t_P}) \\
\cos(2\pi f_{(N-1)\Delta t + t_1}) & \sin(2\pi f_{(N-1)\Delta t + t_1}) & \cdots & \cos(2\pi f_{(N-1)\Delta t + t_P}) & \sin(2\pi f_{(N-1)\Delta t + t_P}) \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
\cos(2\pi f_{(N-1)\Delta t + (N-1)\Delta t + t_1}) & \sin(2\pi f_{(N-1)\Delta t + (N-1)\Delta t + t_1}) & \cdots & \cos(2\pi f_{(N-1)\Delta t + (N-1)\Delta t + t_P}) & \sin(2\pi f_{(N-1)\Delta t + (N-1)\Delta t + t_P})
\end{bmatrix}
\]

\( t_p \) is the starting time for the \( p^{th} \) data segment.

The total squared error for all the segments is:

\[
e = \sum_{p=1}^{P} \left\| \bar{x}_p - \Psi_p \left( \Psi_p^T \Psi_p \right)^{-1} \Psi_p^T \bar{x}_p \right\|^2
\]

The Levenberg-Marquardt method [22-24] can be used to find an optimal solution according to Eqn (14) by updating \( \mathbf{f} \) at each iteration with

\[
\Delta \mathbf{f} = (\lambda \mathbf{I} + \sum_{p=1}^{P} \mathbf{J}_p^T \mathbf{J}_p)^{-1} \sum_{p=1}^{P} \mathbf{J}_p^T (\bar{x}_p - \Psi_p \left( \Psi_p^T \Psi_p \right)^{-1} \Psi_p^T \bar{x}_p)
\]

where \( \lambda \) is a damping factor for improving algorithm convergence, \( \mathbf{I} \) is the identity matrix, \( \mathbf{J}_p \) is the Jacobian matrix of partial derivatives of \( \mathbf{x}^r \) with respect to \( \mathbf{f} \) evaluated at all \( Q \) data points; \( i.e., \)

\[
\mathbf{J}_p = \begin{bmatrix}
\frac{\partial x_1^p}{\partial f_1} & \cdots & \frac{\partial x_1^p}{\partial f_M} \\
\vdots & \ddots & \vdots \\
\frac{\partial x_Q^p}{\partial f_1} & \cdots & \frac{\partial x_Q^p}{\partial f_M}
\end{bmatrix}
\]
Because of noise and phase differences, the $\phi$ value \( (\Psi_p^T \Psi_p)^{-1} \Psi_p^T \tilde{X}_p \) computed from different data segments may not be exactly the same. Another round of least-squares optimization could be performed with all the data as one segment and the determined frequencies as initial values.

![Figure 2](image)

**Figure 2. Illustration of $e$ as a function of $M$**

If the number of components ($M$) is unknown, the Levenberg-Marquardt method can be used to determine $f$ for a range of $M$ values. Each of the $M$ values corresponds to a squared error ($e$). If $e$ does not appreciably decrease beyond $M^*$, $M^*$ can be used as the optimum $M$. Figure 2 is a plot of $e$ vs. $M$ for a signal with four components. It is clear that an $M$ value higher than 4 does not reduce $e$ significantly and thus 4 can be selected as the optimum $M$ as expected. Some other methods are also available for the estimation of $M$, especially under strong noise condition and they can be found in [25].

The least-squares based algorithm can thus be summarized into the following major steps:

1. **Divide the data into segments of $Q$ data points ($Q \geq M$).**
(2) Initialize $M$ and the $M$ frequency values. Fourier transform can be used to analyze the full data set. The frequencies corresponding to peak values on the obtained Fourier spectrum can be used to initialize the frequency values.

(3) Run the Levenberg-Marquardt algorithm based on Eqn (14) till the error $e$ does not change appreciably and/or $\Delta f$ becomes very small.

(4) Run the Levenberg-Marquardt algorithm again by using all data as one segment based on Eqn (8) till the error $e$ does not change appreciably and/or $\Delta f$ becomes very small.

(5) Compute $\phi$ by using Eqn (7) and amplitude and phase by Eqns (9) and (10).

Because of differences in noise levels, total available data points, richness of frequency components, and how close the initialized frequency values to the real values, it is difficult to conclude what is the best segment for all measured signals. In real measurements, a low pass filter is often applied to eliminate aliasing effect. In most applications, the number of total measured data points is much larger than the number of frequency components, which allows splitting the data in a wide range. A Fourier transform can be performed on the whole data set to estimate the rough frequency values for initialization. As long as the initialized frequencies are within the convergence range, the exact number of data points in each sub-segment is not critical. As explained, the longer the data segment, the shorter the convergence area. If the program cannot converge, the data can be divided into more segments. A data splitting sub-routine can be easily developed to split the data into more pieces automatically without manual tries. For a signal, if there are too many frequency components, the algorithm may not work when very few data are measured.
It is worth to mention that the total number of parameters will be reduced to one for signals that can be expressed as \( x(t) = \sum_{m=1}^{M} A_m \cos(2\pi f_m t + \phi_m) \) with known \( r_m \) values.

The developed method can also be applied to complex signals and the corresponding algorithm can be derived by following the above procedures.

3. Simulations and experiment

3.1. A signal with Two Components

The signal

\[
x(n) = \sin\left(\frac{2\pi 49.85n}{f_s} + 1.2\right) + 0.01\sin\left(\frac{2\pi 61.814n}{f_s} + 0.95\right)
\]

was used in [1]. The sampling frequency \( f_s \) was 1500 Hz and a total of 512 data points were used, as in [1]. For the proposed time-domain least-squares method, the data were split into 8 segments of 64 data points each; and 30 Hz and 70 Hz were used as initial values. The results from the proposed method are compared with those of [1] in Table 1.

The proposed method gave precisely the true values and outperformed the MICA (multi-spectrum-line interpolation correction algorithm) used in [1], which includes some approximation when the method was developed. This is expected because the approximation in MICA prevents the accuracy of MICA reaching machine intrinsic precision for noise-free data. Note that the signal is expressed with sine functions as was done in [1].
Table 1. Parameters estimated for a signal with two frequencies

<table>
<thead>
<tr>
<th>True values</th>
<th>$f$</th>
<th>49.85</th>
<th>61.814</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>1</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td>$\Phi$</td>
<td>1.2</td>
<td>0.95</td>
<td></td>
</tr>
</tbody>
</table>

Results in Wu and Zhao [1]

<table>
<thead>
<tr>
<th>$f$</th>
<th>49.8503</th>
<th>61.7911</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>1</td>
<td>0.01</td>
</tr>
<tr>
<td>$\Phi$</td>
<td>1.1997</td>
<td>0.9701</td>
</tr>
</tbody>
</table>

Results from proposed method

<table>
<thead>
<tr>
<th>$f$</th>
<th>49.850000000</th>
<th>61.814000000</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>1.0000000000</td>
<td>0.0100000000</td>
</tr>
<tr>
<td>$\Phi$</td>
<td>1.2000000000</td>
<td>0.9500000000</td>
</tr>
</tbody>
</table>

(Please note: The frequency errors with the method in Wu and Zhao [1] are 0.006% and 0.04%. The errors with the proposed method are 0.000000000%, which indicates the accuracy has reached machine intrinsic precision.)

3.2. A Multi-frequency Signal with Noise

White noise is very common in applications and incurs difficulty for signal parameter estimation because all the estimated frequency points are contaminated by the energy from the white noise. When uniform distributed white noise with its maximum amplitude as 0.01 was added to the signal in Eqn (18), the developed time-domain least-squares method could still give good estimation for the multi-frequency signal parameters. The 512 data points were split into two segments. The frequencies were initialized with 45 Hz and 65 Hz. The results are shown in Table 2.

Table 2. Parameters estimation for a multiple-frequency signal with random noise

<table>
<thead>
<tr>
<th>True values</th>
<th>$f$</th>
<th>49.85</th>
<th>61.814</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>1</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td>$\Phi$</td>
<td>1.2</td>
<td>0.95</td>
<td></td>
</tr>
</tbody>
</table>

Results from proposed method

<table>
<thead>
<tr>
<th>$f$</th>
<th>49.84971719</th>
<th>61.80624652</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>1.000253196</td>
<td>0.009850594</td>
</tr>
<tr>
<td>$\Phi$</td>
<td>1.200247758</td>
<td>0.948589241</td>
</tr>
</tbody>
</table>

To further test the performance of the developed algorithm, Monte Carlo simulations were performed under different SNRs and different lengths of data. The data were split
into two segments. All the simulations indicated that the method was reliable. Figure 3 displays the relative estimated frequency error at different data lengths and SNRs for the 49.85 Hz frequency component given in Eqn (18). It clearly shows that the relative error does not significantly increase over a wide range of SNRs and data length. Figure 4 shows the relative frequency error under different SNRs simulated by Monte Carlo when only 256 data points are available. Each point in Figure 4 represents one simulation with randomly assigned initial values in the range ±5 Hz centered around the true values. If the traditional separable estimation method is used, the algorithm cannot converge with such a wide range of parameter initialization.

Figure 3. Simulated relative frequency error for the 49.85Hz component under different available data lengths and SNRs (The unit of SNR is dB.)
3.3. A Multi-frequency Signal with Four Close Frequencies

The following signal appears in [26].

\[
x(n) = 5.3 \cos(2\pi 100.3n / f_s + 20\pi / 180) + 8.4 \cos(2\pi 102.3n / f_s + 50\pi / 180)
+ 3.7 \cos(2\pi 104.3n / f_s + 70\pi / 180) + 1.8 \cos(2\pi 106.3n / f_s + 100\pi / 180)
\]  

It is a noise-free signal. A total 1024 data points were sampled at 1024 Hz. The sampling frequency and the number of data points are the same as those in [26]. For the proposed time-domain least-squares method, the 1024 data points were split into 16 segments with 64 data points in each segment. The frequencies were initialized with 90 Hz, 101 Hz, 105 Hz and, 109 Hz, respectively. The results from the proposed method are compared with the results of Qin et al. [26] in Table 3. The proposed method produced the true values, therefore it is clearly more accurate than the results reported in [26]. The latter also included some approximations in the algorithm, which therefore would never reach machine intrinsic precision for noise-free data. If the traditional separable estimation method is used, the algorithm cannot converge with such a wide range of parameter
initialization. If all the initializations are within a special range, e.g. above 109 Hz or below 90 Hz, the results from the developed algorithm may be trapped into local minima and the results may have two or more identical frequencies or the residual has one or more clear frequency components. In this case, the number of effective frequency components is actually reduced. If this happens, the initializations can be changed by program or manually to reach global optimum.

**Table 3.** Parameters estimation for a signal with four close frequencies

<table>
<thead>
<tr>
<th>Theoretical values</th>
<th>( f )</th>
<th>100.3</th>
<th>102.3</th>
<th>104.3</th>
<th>106.3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( A )</td>
<td>5.3</td>
<td>8.4</td>
<td>3.7</td>
<td>1.8</td>
</tr>
<tr>
<td>( \Phi )</td>
<td>20</td>
<td>50</td>
<td>70</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Results in Qin et al. [26]</th>
<th>( f )</th>
<th>100.304382</th>
<th>102.301483</th>
<th>104.299728</th>
<th>106.296937</th>
</tr>
</thead>
<tbody>
<tr>
<td>( A )</td>
<td>5.197095</td>
<td>8.138096</td>
<td>3.721981</td>
<td>1.801701</td>
<td></td>
</tr>
<tr>
<td>( \Phi )</td>
<td>18.092668</td>
<td>49.242608</td>
<td>70.018152</td>
<td>100.389114</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Results from proposed method</th>
<th>( f )</th>
<th>100.300000000</th>
<th>102.300000000</th>
<th>104.300000000</th>
<th>106.300000000</th>
</tr>
</thead>
<tbody>
<tr>
<td>( A )</td>
<td>5.300000000</td>
<td>8.400000000</td>
<td>3.700000000</td>
<td>1.800000000</td>
<td></td>
</tr>
<tr>
<td>( \Phi )</td>
<td>20.000000003</td>
<td>50.000000000</td>
<td>69.999999999</td>
<td>100.000000000</td>
<td></td>
</tr>
</tbody>
</table>

(The frequency errors with the method Qin et al. [26] are around 0.004%. The errors with the proposed method are 0.00000000%, which indicates the accuracy has reached machine intrinsic precision.)

3.4. A Measured Signal

A sinusoid signal was generated with a function generator (Model DS340, Stanford Research Systems, SunnyVale, CA, USA) with a frequency of 5.123 Hz and amplitude 8 V. Labview and a 16-bit NI USB6216 data acquisition card (National Instrument, Austin, TX, USA) were used to sample the signal. A total of 512 data points were acquired. The signal was separated into two segments with 256 data points each. The algorithm estimated the signal frequency as 5.12305 Hz with a relative error of 0.0001%. Without step 4, the relative error would be 0.0005%.
4. Conclusion

In this work, a modified time-domain least-squares method has been proposed, which improves the convergence of the traditional separable estimation method while keeping the accuracy of the traditional separable estimation method. For noise-free signals, the errors reach the intrinsic machine precision. If a signal contains too many frequency components and there are very limited measured data points, the method may be not reliable and lead to error results.

References


