

RESEARCH

Open Access

# High-order approximation algorithm using Gram-Schmidt QR transformations coupled with an iterative correction process for tracking frequency drift in OFDM systems

Rainfield Y Yen<sup>1</sup>, Hong-Yu Liu<sup>2\*</sup> and Yung-Chang Yin<sup>2</sup>

## Abstract

To jointly track frequency drift (or fine frequency offset) and channel state variation for orthogonal frequency division multiplexing (OFDM) communications over mobile wireless channels, the maximum-likelihood (ML) estimation techniques are commonly adopted. A major difficulty arises from the highly nonlinear nature of the log-likelihood function which renders local extrema or multiple solutions. The use of mathematical approximation coupled with an adaptive iteration algorithm is a viable approach to ease problem solving. The approximation methods used in existing works are usually of the first-order level. In this work, we devise a high-order approximation algorithm to improve the tracking performance. As shall be seen, the problem amounts to finding the roots of an approximate high-degree polynomial equation derived from the log-likelihood function. To this end, we triangularize the companion matrix constructed from the high-degree polynomial using iterative Gram-Schmidt QR transformations, an eigenvalue problem in matrix computation, coupled with another iterative correction process to produce a frequency offset estimate to good accuracy. It is found that a high-order approximation algorithm formulated can achieve much better tracking performance in terms of estimation accuracy, tracking range, and error rate results than various other tracking algorithms appearing in the literature.

**Keywords:** Frequency tracking; Channel estimation; Maximum-likelihood estimation; QR transformation; Orthogonal frequency division multiplexing

## 1. Introduction

To operate an orthogonal frequency division multiplexing (OFDM) system in a mobile wireless environment, initial acquisition, sometimes referred to as the coarse synchronization, is required for both time and frequency synchronization [1-6]. The usual approach for acquisition uses some correlation techniques (utilizing the property of correlation between repeated symbol blocks [2], correlation between data symbols and the redundant symbol copies within cyclic prefix [3], or correlation between first and second half of the training symbol block [4]) which require no knowledge of the channel

state information. After acquisition has been completed, because of Doppler spread, multipath fading, as well as instability of the local oscillators, small carrier frequency drift, or fine carrier frequency offset (CFO), and time variation of the channel impulse response (CIR) or channel frequency response (CFR) will continue to exist and hence must be tracked periodically or frequently (depending on the channel fading rate, tracking period or frequency may vary in different applications) [1]. Frequency tracking is sometimes referred to as fine frequency synchronization [1] which is the theme of this paper. In the literature, numerous frequency tracking algorithms have been proposed for OFDM communications [1,7-13]. Unlike the coarse frequency synchronization in acquisition, the fine frequency synchronization or frequency tracking usually requires channel

\* Correspondence: hongyu.liu@msa.hinet.net

<sup>2</sup>Department of Electrical Engineering, Fu Jen Catholic University, No. 510, Zhongzheng Rd., Xinzhuang Dist, New Taipei City 24205, Taiwan  
Full list of author information is available at the end of the article

knowledge. Also, tracking channel state variation usually requires frequency to be synchronized [14-20]. That is to say, either perfect channel estimation is assumed in deriving frequency offset algorithms or vice versa, perfect frequency synchronization is assumed in deriving channel estimation algorithms [10]. As a result, many researchers propose joint frequency tracking and channel estimation algorithms for OFDM communications [1,9,10,12-14,21-25]. All the tracking schemes mentioned above adopt the maximum-likelihood (ML) estimation (MLE) technique. The major task in MLE is to maximize a log-likelihood function with respect to the parameters to be estimated. For the joint tracking problem at hand, two parameters, *viz.*, CFO and CIR (or CFR), are to be estimated. To find the maximum or the peak, we set partial differentiations of the log-likelihood function to zero with respect to these two parameters resulting in two simultaneous equations in terms of the two parameters. Solving the simultaneous equations gives the solutions of the two sought-after estimators. However, the log-likelihood function is usually highly nonlinear; its derivatives will also be highly nonlinear. Therefore, there exist multiple peaks (often infinitely many) or multiple solutions. The highest peak is called the global solution and is the sought-after solution. The rest of the peaks are called local maxima or more generally local extrema as some applications may seek a minimum or a valley. A local extremum is not the desired solution. To design an algorithm to accurately locate the desired global solution may be no easy task. To achieve the purpose, many algorithm processes are impractically complicated [9,13]. A straightforward approach is to use the gradient method [10]. However, for the gradient method to converge to the desired global solution, it is critical to properly choose the starting point as well as the adaptation step size. Improper choices of the starting point and/or step size may lead to an undesired local extremum solution or even result in divergence. The starting point and step size may vary when the channel characteristic or the environment changes. For most practical applications, hardly an analysis can be provided for the choices of the starting point and step size. These choices then must be made by trial and error which may often take much time or even render failure. In general, when the hyperboloid surface represented by the log-likelihood function is highly irregular in shape with densely distributed local extrema, the search by gradient method becomes very difficult or even impossible. Thus, though theoretically workable, a gradient algorithm may not be practically implementable. At any rate, simulation results given by existing algorithms cited previously are mostly given for small frequency drifts. We have tested some

of these algorithms, though not all of them, and found that, when the frequency drift (true CFO value) becomes large (say, exceeding 0.25 of the carrier spacing), the algorithms are not able to deliver a convincing performance. The culprit is the local extrema obstacle. To overcome the local extrema difficulty, a viable approach is to use mathematical approximation as done in the expectation-maximization (EM)-based algorithm of [12,24,25] and the joint maximum-likelihood channel and frequency estimation (JML-CFE) algorithm of [12]. The EM-based algorithm was first proposed by [12]. Then, the authors in [24] extended the idea to MIMO applications and [25] extended it to fast fading channels. Another approximation approach is called the recursive least squares (RLS)-based method. This approach requires linearization of the estimated parameters. The basic idea of approximation is to expand the cost function into its Taylor series form and then truncate the terms of high power orders as an approximation. The approximation methods used in [12,24,25] and [14] are all the first-order approximation methods (the derivative of the truncated Taylor series is a polynomial of degree one). However, the first order is too low an order to achieve tracking/estimation results with sufficient accuracy, especially when the frequency drift is large. Therefore, again, simulation results in [12] and [14] are given only for small frequency drifts.

In this work, we are inspired to extend the first-order approximation method to a high-order approximation method in an attempt to improve the tracking/estimation performance. As will be shown, the problem finally amounts to finding the roots of a high-degree real coefficient polynomial equation resulting from the high-order Taylor series truncation (the derivative of the log-likelihood function is expanded into a Taylor series). To this end, we triangularize the companion matrix constructed from the high-degree polynomial using iterative *QR* transformations by way of the Gram-Schmidt orthogonalization process (to be called the Gram-Schmidt *QR* transformations) [26-28]. However, due to the Taylor series truncation, the triangularized matrix result obtained from the Gram-Schmidt *QR* transformations is only approximate. Then, to complete our entire high-order approximation algorithm, a correction process, also iterative, is introduced to work in conjunction with the Gram-Schmidt *QR* transformations to eventually produce a final CFO estimate to good accuracy. We will show that such a high-order approximation algorithm, in terms of tracking range, estimation accuracies, and symbol error rate (SER), will gain much improved performance over other algorithms appearing in the literature for OFDM communications.

We note here that the development of the fine CFO tracking with wide tracking range while keeping high estimation accuracy and low computational complexity is beneficial to the design of the synchronization algorithms, viewing the fact that different coarse synchronization algorithms lead to different magnitudes of the fine residual frequency offsets. Therefore, if the followed fine synchronization algorithm has wide estimation range, the number of times of performing coarse synchronization in a certain time period can be reduced because only the fine synchronization is needed to be performed without doing coarse synchronization in advance each time, which reduces the computational loads. The reason explains why we make efforts on improving the fine synchronization with wide tracking range, high estimation accuracy, and low computational complexity.

## 2. Signal and system model

Define  $\mathbf{X} = \text{diag}\{X_0, X_1, \dots, X_{N-1}\}$  as the diagonal matrix with  $\{X_k, k = 0, 1, \dots, N-1\}$  as the set of transmitted baseband frequency domain data symbols over an OFDM symbol block of length  $N$  (in symbol units),  $\mathbf{r} = [r_0, r_1, \dots, r_{N-1}]^T$  as the time domain baseband received signal vector, and  $\mathbf{w} = [w_0, w_1, \dots, w_{N-1}]^T$  as the time domain baseband noise vector where  $\{w_n\}$  are independent, identically distributed (i.i.d.) complex Gaussian random variables (RVs) with zero mean and variance  $\sigma_w^2$  and can usually be denoted by  $w_n \sim \mathcal{N}(0, \sigma_w^2)$  for convenience, where  $T$  denotes transpose. For a frequency-selective channel of dispersion length  $\nu$ , let  $\mathbf{h} = [h_0, h_1, \dots, h_{\nu-1}]^T$  be the CIR vector with  $\{h_m, m = 0, 1, \dots, \nu-1\}$  being spatially uncorrelated and  $\mathbf{H} = [H_0, H_1, \dots, H_{N-1}]^T$  be the CFR vector [29,30]. Here,

$$\mathbf{H} = \mathbf{F}_\nu \mathbf{h}, \quad (1)$$

where

$$\mathbf{F}_\nu = \begin{bmatrix} 1 & 1 & \dots & \dots & 1 \\ 1 & e^{-j2\pi/N} & \dots & \dots & e^{-j2\pi(\nu-1)/N} \\ 1 & e^{-j4\pi/N} & \ddots & \dots & e^{-j4\pi(\nu-1)/N} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 1 & e^{-j2\pi(N-1)/N} & \dots & \dots & e^{-j2\pi(N-1)(\nu-1)/N} \end{bmatrix} \quad (2)$$

is an  $N \times \nu$  discrete Fourier transform (DFT) matrix. Assuming the frequency offset normalized to subcarrier spacing is  $\delta$ . Then, adopting unitary DFT for signal data, after demodulation and discarding the cyclic prefix at the receiver, the complex baseband sample at the  $n$ th

time slot in the received time domain OFDM block is given by [2,9,10]

$$r_n = e^{j2\pi n \delta / N} y_n + w_n, \quad n = 0, 1, \dots, N-1, \quad (3)$$

where  $y_n$  is the offset-free noiseless data sample given as

$$y_n = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} H_k X_k e^{j2\pi n k / N}. \quad (4)$$

We note here that removal of cyclic prefix symbols requires accurate time synchronization which, as stated previously, has been completed in acquisition. In fact, timing error can be divided into integer part and fractional part. The coarse timing acquisition deals with the correction of the integer part of the timing error, while the fractional part of timing error is absorbed into the estimation of channel impulse response [31].

It is readily seen that (3) can be put in vector form as

$$\mathbf{r} = \frac{1}{\sqrt{N}} \mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{H} + \mathbf{w} = \frac{1}{\sqrt{N}} \mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_\nu \mathbf{h} + \mathbf{w}, \quad (5)$$

where  $H$  denotes Hermitian transpose and

$$\mathbf{D}_\delta = \text{diag}\{1, e^{j2\pi\delta/N}, \dots, e^{j2\pi(N-1)\delta/N}\}. \quad (6)$$

## 3. The high-order approximation algorithm

The novelty of our proposed algorithm lies in the use of high-order approximation for a nonlinear derivative of the log-likelihood function and the use of an innovative iterative correction process to refine the approximate solution obtained via QR transformation adopted from a matrix computation theory. The most important is such an approach indeed yields very satisfying results. We shall now present our scheme in detail.

### 3.1 Formulation of the maximum-likelihood estimation

From (5), a log-likelihood function can be derived as

$$\begin{aligned} \ln \Lambda &= -N \ln(\pi \sigma_w^2) - \frac{1}{\sigma_w^2} \left\| \mathbf{r} - \frac{1}{\sqrt{N}} \mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_\nu \mathbf{h} \right\|^2 \\ &= -N \ln(\pi \sigma_w^2) - \frac{1}{\sigma_w^2} \left( \mathbf{r}^H \mathbf{r} - \frac{2}{\sqrt{N}} \text{Re}\{ \mathbf{r}^H \mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_\nu \mathbf{h} \} \right. \\ &\quad \left. + \frac{1}{N} \mathbf{h}^H \mathbf{F}_\nu^H \mathbf{X}^H \mathbf{F}_N \mathbf{D}_\delta^H \mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_\nu \mathbf{h} \right) \\ &= -N \ln(\pi \sigma_w^2) - \frac{1}{\sigma_w^2} \left( \mathbf{r}^H \mathbf{r} - \frac{2}{\sqrt{N}} \text{Re}\{ \mathbf{r}^H \mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_\nu \mathbf{h} \} \right. \\ &\quad \left. + \mathbf{h}^H \mathbf{F}_\nu^H \mathbf{X}^H \mathbf{X} \mathbf{F}_\nu \mathbf{h} \right), \end{aligned} \quad (7)$$

where  $\text{Re}\{\cdot\}$  means real part. Now, by setting  $\partial \ln \Lambda / \partial \mathbf{h} = \mathbf{0}$ , we can obtain a solution for  $\mathbf{h}$  that will render a maximum  $\ln \Lambda$  for a fixed  $\delta$ . This is just an ML estimate of  $\mathbf{h}$  at a fixed  $\delta$  given by [10]

$$\hat{\mathbf{h}} = \frac{1}{\sqrt{N}} (\mathbf{F}_v^H \mathbf{X}^H \mathbf{X} \mathbf{F}_v)^{-1} \mathbf{F}_v^H \mathbf{X}^H \mathbf{F}_N \mathbf{D}_\delta^H \mathbf{r}. \quad (8)$$

Constant modulus training sequence has been proven optimal for channel estimation [32]. Chu sequence [33], for example, falls onto this category. We shall use a Chu sequence given by  $X_k = e^{j\pi m k^2 / N}$   $m$  being any integer relatively prime to  $N$ . This results in  $\mathbf{X}^H \mathbf{X} = \mathbf{I}_N$ . Then, (8) can be simplified to

$$\hat{\mathbf{h}} = \frac{1}{N\sqrt{N}} \mathbf{F}_v^H \mathbf{X}^H \mathbf{F}_N \mathbf{D}_\delta^H \mathbf{r}. \quad (9)$$

Next, setting  $\frac{\partial \ln \Lambda}{\partial \delta} = 0$  leads to

$$\text{Im}\{\mathbf{r}^H \mathbf{Q} \mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v \mathbf{h}\} = 0, \quad (10)$$

where  $\mathbf{Q} = \text{diag}\{0, 1, \dots, N-1\}$  and  $\text{Im}\{\cdot\}$  means imaginary part. Replacing the  $\mathbf{h}$  in (10) by the  $\hat{\mathbf{h}}$  of (9), we find

$$\begin{aligned} \text{Im}\{\mathbf{r}^H \mathbf{Q} \mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v \mathbf{F}_v^H \mathbf{X}^H \mathbf{F}_N \mathbf{D}_\delta^H \mathbf{r}\} &= \text{Im}\{\mathbf{r}^H \mathbf{D}_\delta \mathbf{G} \mathbf{D}_\delta^H \mathbf{r}\} \\ &= \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} \text{Im}\left\{r_m^* r_n g_{m,n} e^{j2\pi\delta(m-n)/N}\right\} = 0, \end{aligned} \quad (11)$$

where  $g_{m,n}$  is the  $(m+1, n+1)$ th element of an  $N \times N$  matrix  $\mathbf{G}$  given by

$$\begin{aligned} \mathbf{G} &= \mathbf{Q} \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v \mathbf{F}_v^H \mathbf{X}^H \mathbf{F}_N = \left\{g_{m,n}\right\} \\ &= \begin{bmatrix} 0 & & & & 0 \\ \sum_{l=0}^{v-1} \sum_{k=0}^{N-1} X_k e^{-j2\pi(l-1)k/N} & \sum_{k'=0}^{N-1} X_{k'}^* e^{j2\pi l k' / N} & & & \sum_{l=0}^{v-1} \sum_{k=0}^{N-1} X_k e^{-j2\pi(l-1)k/N} & \sum_{k'=0}^{N-1} X_{k'}^* e^{j2\pi(l-1)k' / N} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ (N-1) \sum_{l=0}^{v-1} \sum_{k=0}^{N-1} X_k e^{-j2\pi(l-N+1)k/N} & \sum_{k'=0}^{N-1} X_{k'}^* e^{j2\pi l k' / N} & & & (N-1) \sum_{l=0}^{v-1} \sum_{k=0}^{N-1} X_k e^{-j2\pi(l-N+1)k/N} & \sum_{k'=0}^{N-1} X_{k'}^* e^{j2\pi(l-1)k' / N} \\ \dots & & & & & \\ \dots & \sum_{l=0}^{v-1} \sum_{k=0}^{N-1} X_k e^{-j2\pi(l-1)k/N} & \sum_{k'=0}^{N-1} X_{k'}^* e^{j2\pi(l-N+1)k' / N} & & & \\ \dots & \vdots & \vdots & \vdots & & \\ \dots & (N-1) \sum_{l=0}^{v-1} \sum_{k=0}^{N-1} X_k e^{-j2\pi(l-N+1)k/N} & \sum_{k'=0}^{N-1} X_{k'}^* e^{j2\pi(l-N+1)k' / N} & & & \end{bmatrix}. \end{aligned} \quad (12)$$

Since (11) is now channel independent, we have decoupled  $\delta$  from  $\mathbf{h}$  and (11) can thus be solved for  $\delta$  alone. However, (11) is highly nonlinear in  $\delta$  and contains infinite number of solutions. We only desire the one solution that yields the global maximum of  $\ln \Lambda$ . The task is not possible by analytical means. However, we can resort to the numerical method.

### 3.2 The approximation approach

We now expand the exponential term  $e^{j2\pi(m-n)\delta/N}$  in (11) into an infinite series (Taylor series expansion) and then truncate this infinite series beyond terms with power order higher than  $K$  to obtain

$$\begin{aligned} e^{j2\pi(m-n)\delta/N} &= 1 + j \frac{2\pi(m-n)}{N} \delta - \frac{1}{2!} \left[ \frac{2\pi(m-n)}{N} \right]^2 \delta^2 - \dots \\ &\quad + \frac{1}{K!} \left[ j \frac{2\pi(m-n)}{N} \right]^K \delta^K. \end{aligned} \quad (13)$$

Substituting this approximate expression of (13) for the exponential term into (11), we will get a  $K$  degree polynomial of  $\delta$  with real coefficients. Therefore, solving (11) becomes equivalent to finding the roots of a real polynomial of degree  $K$ . This is an eigenvalue problem in matrix computations [26]. Express the  $K$  degree polynomial as

$$p(\delta) = a_0 + a_1 \delta + \dots + a_{K-1} \delta^{K-1} + \delta^K = 0. \quad (14)$$

Notice that we have normalized the polynomial such that the coefficient  $a_K$  is unity. This can be easily done just by dividing the original polynomial equation by the original nonzero  $a_K$ . To make it distinguishable, denote the original nonzero  $a_K$  by a different symbol  $\tilde{a}_K$ . It then

can be readily verified that the coefficients  $\{a_k\}$  in (14) are given by

$$\begin{aligned}
 a_k &= \frac{1}{\tilde{a}_K} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} \text{Im} \left\{ \frac{r_m^* r_n g_{m,n}}{k!} \left[ \frac{j2\pi(m-n)}{N} \right]^k \right\} \\
 &= \frac{1}{\tilde{a}_K \cdot k!} \left( \frac{2\pi}{N} \right)^k \text{Im} \left\{ j^k \sum_{n=0}^k (-1)^n C_n^k \mathbf{r}^H \mathbf{Q}^{k-n} \mathbf{G} \mathbf{Q}^n \mathbf{r} \right\}, \\
 & \quad k = 0, 1, \dots, K
 \end{aligned} \tag{15}$$

where  $C_n^m = \frac{m!}{(m-n)!n!}$ ,  $m \geq n$ .

A word is in order here. We define our approximation order as  $K$  when the polynomial degree in (14) is  $K$ . However, note that (14) is the approximation of (11) which is a derivative of the log-likelihood function of (7). Thus, here the Taylor series truncation is performed after differentiation of the log-likelihood function, while in [12] and [14], the Taylor series truncation is performed directly on the log-likelihood function. According to our definition, [12] and [14] are actually using first-order approximations. Note that when  $K=1$ , we also have a first-order approximation algorithm.

### 3.3 QR transformations

Now, from (13), we construct a  $K \times K$  square matrix called the companion matrix as [27]

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & \cdots & \cdots & 0 & -a_0 \\ 1 & 0 & \ddots & \ddots & 0 & -a_1 \\ 0 & 1 & \ddots & \ddots & \vdots & \vdots \\ \vdots & 0 & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 & \vdots \\ 0 & \cdots & \cdots & 0 & 1 & -a_{K-1} \end{bmatrix}. \tag{16}$$

It has been known that a square matrix can be triangularized by iterative  $QR$  transformations [26-28], where  $Q$  is an orthogonal matrix and  $R$  is an upper triangular matrix. When all the roots of the polynomial of (14) are real, these roots will constitute the diagonal elements of the eventual triangularized  $A$  matrix that are also the eigenvalues of  $A$  [26-28]. In case the polynomial equation of (14) has complex roots in conjugate pairs, the iterative  $QR$  transformations will lead to a Hessenberg matrix [26] which is quasi-triangular. That is, the subdiagonal immediately below the main diagonal will contain non-zero elements. However, on the main diagonal, each real root will appear as an element while each complex root will not appear but is replaced by a certain unpredictable number, either real or complex, as an element [26]. There is no way of telling which elements are real roots and which are the numbers replacing complex roots. However, it can be certain that the one real root is there to be the desired global solution for the CFO estimate

which must be real. We shall detail later how to find this global solution by our algorithm design.

The process of iterative  $QR$  transformations called the Gram-Schmidt  $QR$  transformations involves two operation phases alternatively performed. The first operation phase is called the Gram-Schmidt  $QR$  decomposition. The  $QR$  decomposition is carried out by the Gram-Schmidt orthogonalization process [26,34]. The second operation phase is an iterative transformation process (or a triangularization process). However, when the polynomial degree  $K$  gets too high, the triangularization process will begin to produce less accurate results [34]. Fortunately, for the OFDM tracking problem at hand, we do not have to use a very high order of  $K$  and only need a crude result out of the triangularization process since a complementing correction process coupled with the triangularization process will carry the burden and take care the rest of the matter to eventually bring a final CFO estimate result to great accuracy. As a result, we do not need to execute a great many iterations of  $QR$  transformations. It can be demonstrated that just a couple of iterations of  $QR$  transformations will suffice. For fine frequency synchronization, the frequency drift is less than half the carrier spacing ( $|\delta| < 0.5$ ) [9,10,12,14]. In this case, computer experiments show that our order 2 algorithm can produce results with good accuracy. When a wider tracking range is desired ( $|\delta| > 0.5$ ), we will need at least an order 4 algorithm.

The iterative  $QR$  transformations are carried out as follows:

Let the square matrix  $A$  of (13) be denoted as

$$\mathbf{A} = [\mathbf{a}_1 \quad \mathbf{a}_2 \quad \cdots \quad \mathbf{a}_K], \tag{17}$$

where  $\{\mathbf{a}_k, k = 1, 2, \dots, K\}$  are  $K \times 1$  column vectors of  $\mathbf{A}$ . We then carry out the Gram-Schmidt orthogonalization process as follows:

Define the projection of a vector  $\mathbf{a}$  on a unit vector  $\mathbf{e}$  as

$$\text{proj}_{\mathbf{e}} \mathbf{a} = \frac{\mathbf{a}^T \mathbf{e}}{\|\mathbf{e}\|^2} \mathbf{e}, \tag{18}$$

where  $T$  denotes the transpose and  $\|\mathbf{e}\| = \sqrt{\mathbf{e}^T \mathbf{e}}$  is the Euclidean norm of  $\mathbf{e}$ . Then, let

$$\begin{aligned}
 \mathbf{u}_1 &= \mathbf{a}_1, \\
 \mathbf{u}_2 &= \mathbf{a}_2 - \text{proj}_{\mathbf{e}_1} \mathbf{a}_2, \\
 \mathbf{u}_3 &= \mathbf{a}_3 - \text{proj}_{\mathbf{e}_1} \mathbf{a}_3 - \text{proj}_{\mathbf{e}_2} \mathbf{a}_3, \\
 & \quad \vdots \\
 \mathbf{u}_K &= \mathbf{a}_K - \sum_{k=1}^{K-1} \text{proj}_{\mathbf{e}_k} \mathbf{a}_K,
 \end{aligned} \tag{19}$$

$$\text{with } \mathbf{e}_k = \frac{\mathbf{u}_k}{\|\mathbf{u}_k\|}.$$

The orthogonal matrix  $Q$  is now formed as

$$\mathbf{Q} = [\mathbf{e}_1 \quad \mathbf{e}_2 \quad \cdots \quad \mathbf{e}_K]. \quad (20)$$

The upper triangular matrix  $\mathbf{R}$  is given by

$$\mathbf{R} = \mathbf{Q}^T \mathbf{A}. \quad (21)$$

This completes the Gram-Schmidt  $QR$  decomposition process. Next, we start the following iterations:

$$\mathbf{A}_1 = \mathbf{R}\mathbf{Q}.$$

Construct  $\mathbf{Q}_1$  from  $\mathbf{A}_1$  using the Gram-Schmidt orthogonal process as done above.

$$\text{Construct } \mathbf{R}_1 = \mathbf{Q}_1^T \mathbf{A}_1.$$

$$\begin{aligned} \mathbf{A}_2 &= \mathbf{R}_1 \mathbf{Q}_1 \\ &\vdots \\ \mathbf{A}_L &= \mathbf{R}_{L-1} \mathbf{Q}_{L-1}. \end{aligned} \quad (22)$$

When  $L$  is sufficiently large, we will find  $\mathbf{A}_L$  to be upper triangular with diagonal elements equal to the roots of the polynomial of (14). Now, the Gram-Schmidt  $QR$  transformations are completed. Note that, as mentioned earlier, in actual operations when coupled with a complementing correction process, we need not to execute many iterations of the  $QR$  transformations ( $L$  needs not be large). In our computer simulations, we have used  $L = 2$  and up to  $K = 6$ .

Of the  $K$  roots obtained, only one root will be desired. Assume the  $K$  roots are  $\delta_1, \delta_2, \dots, \delta_K$ . Substitute each root into (9), then into (7), with the Chu sequence chosen for  $\mathbf{X}$  earlier. The one root that maximizes (7) is the sought-after solution  $\hat{\delta}_0$ , i.e.,

$$\hat{\delta}_0 = \arg \max_{\delta_k} [\ln \Lambda(\delta_k)], \quad k = 1, 2, \dots, K. \quad (23)$$

For the special case of the first-order approximation when  $K = 1$ , (14) directly leads to  $\hat{\delta}_0 = -a_0/a_1$ , and hence, no  $QR$  transformation needs to be executed. Furthermore, in case (14) possesses complex roots,  $\mathbf{A}_L$  will be in the Hessenberg form (quasi-triangular) as stated earlier. However, it will not matter. Exactly as in the all real roots case, we simply substitute each diagonal element into the log-likelihood function of (7). The one that yields the maximum log-likelihood function is the global solution just as given by (23).

### 3.4 The iterative correction process

We now must note that the global solution  $\hat{\delta}_0$  obtained as described above was via an approximation method truncating a Taylor series expansion beyond terms with power order higher than  $K$ , i.e., the approximate expression of (13) was used. Therefore, the solution  $\hat{\delta}_0$  is still an approximate solution (with  $L = 2$  as is to be used, this solution will be even cruder). We can further refine this solution by an iterative correction process as follows.

With the initial estimate  $\hat{\delta}_0$  (this is why we have used the subscript 0 to begin with), we can correct the received signal vector as

$$\mathbf{r}_1 = \mathbf{D}_{\hat{\delta}_0}^H \mathbf{r}_0, \quad (24)$$

where  $\mathbf{r}_0 = \mathbf{r}$  is just the original received signal vector. After this CFO correction,  $\mathbf{r}_1$  is expected to be cleaner than  $\mathbf{r}_0$  ( $= \mathbf{r}$ ). We thus use this corrected signal or replace  $\mathbf{r}$  with  $\mathbf{r}_1$  in (9) to come up with a better log-likelihood function from which a new polynomial of degree  $K$  is then generated, i.e., a new (9) is generated. Carrying out the iterative Gram-Schmidt  $QR$  transformations again as above, we get a new and better CFO estimate  $\hat{\delta}_1$ . Continuing this way, eventually at a certain  $M$ th iteration, we should have  $\hat{\delta}_M \rightarrow \delta$ . Then, our final or overall CFO estimate can be obtained as

$$\hat{\delta} = \sum_{m=0}^M \hat{\delta}_m \rightarrow \delta. \quad (25)$$

To avoid confusion, we shall, from here on, call  $\hat{\delta}$  the final or overall CFO estimate, define the  $i$ th interim

CFO estimate as  $\hat{\delta}_{(i)} = \sum_{m=0}^i \hat{\delta}_m$ ,  $i = 0, 1, \dots, M$ , (notice

that  $\hat{\delta}_{(M)} = \hat{\delta}$ ), and refer to  $\delta$  as the true CFO (the very beginning CFO to be estimated).

Now, replacing the final CFO estimate of (25) for the  $\delta$  in (9), we can get the CIR estimator  $\hat{\mathbf{h}}$  immediately.

We will summarize the iterative adaptive  $QR$  transformation algorithm consisting of two phases alternately. To be clear, the matrices  $\mathbf{Q}_l$ ,  $\mathbf{R}_l$ , and  $\mathbf{A}_l$  mentioned above at the first phase are all added with an extra subscript  $m$  as  $\mathbf{Q}_{l,m}$ ,  $\mathbf{R}_{l,m}$  and  $\mathbf{A}_{l,m}$  respectively, corresponding to the  $m$ th run of the second phase. The proposed iterative adaptive algorithm is listed as below:

Initial condition:  $\mathbf{r}_0 = \mathbf{r}$  and  $\mathbf{A}_{0,0} = \mathbf{A}$ .

For  $m = 0, 1, \dots, M$

For  $l = 0, 1, \dots, L - 1$

Construct orthogonal matrix  $\mathbf{Q}_{l,m}$  from companion matrix  $\mathbf{A}_{l,m}$  using Gram-Schmidt orthogonal process.

Construct  $\mathbf{R}_{l,m} = \mathbf{Q}_{l,m}^T \mathbf{A}_{l,m}$ .

Construct  $\mathbf{A}_{l+1,m} = \mathbf{R}_{l,m} \mathbf{Q}_{l,m}$ .

End

Choose the  $K$  diagonal elements of  $\mathbf{A}_{L,m}$  to be  $\delta_k$ ,  $k = 1, 2, \dots, K$ .

Calculate  $\hat{\delta}_m = \arg \max_{\delta_k} [\ln \Lambda(\delta_k)]$ .

Construct  $\mathbf{r}_{m+1} = \mathbf{D}_{\hat{\delta}_m}^H \mathbf{r}_m$ .

Construct companion matrix  $\mathbf{A}_{0,m+1}$  from the  $K$  degree polynomial that approximates  $\text{Im} \left\{ \mathbf{r}_{m+1}^H \mathbf{D}_{\hat{\delta}_m} \mathbf{r}_{m+1} \right\} = 0$ .

End

The final or overall CFO estimate is  $\hat{\delta} = \sum_{m=0}^M \hat{\delta}_m \rightarrow \delta$ , and the CIR estimate is  $\hat{\mathbf{h}} = \frac{1}{N\sqrt{N}} \mathbf{F}_v^H \mathbf{X}^H \mathbf{F}_N \mathbf{D} \hat{\delta}^H \mathbf{r}$ .

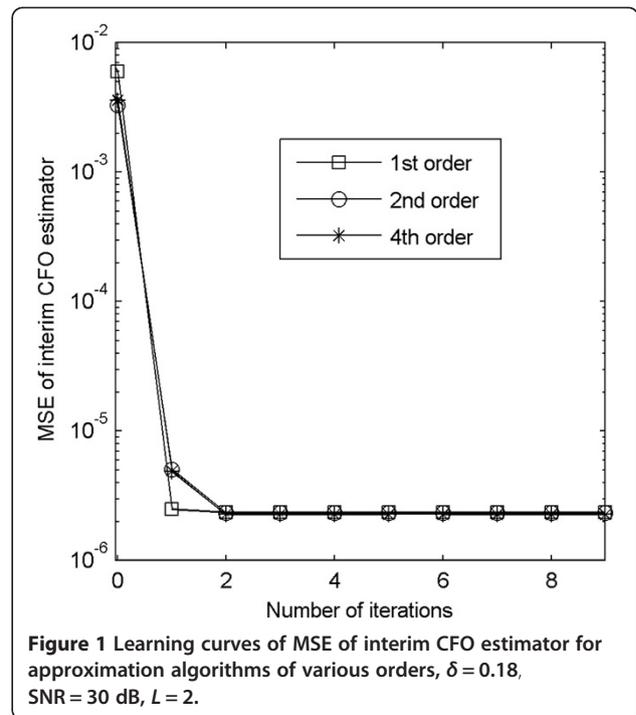
Apparently, when a linear approximation (first-order approximation) is used for a high-degree polynomial, the precision becomes less as compared to a quadratic approximation (second-order approximation). The computational load for a high-order approximation algorithm would be heavier than that of a first-order approximation algorithm since more terms are involved in computations. However, what is important is that we are rewarded with tremendous improvements in CFO tracking range, estimation accuracy, and SER performance. These improvements will be demonstrated with performance comparisons by computer simulations in Section 4.

#### 4. Simulation results and discussions

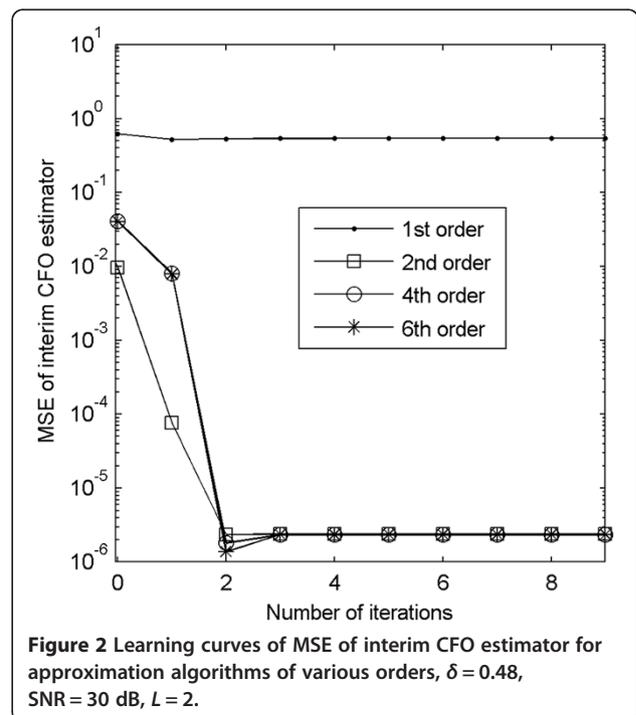
Consider an OFDM system with DFT size  $N = 64$  operated over a frequency-selective channel of dispersion length  $\nu = 9$  having an exponential power profile  $\alpha e^{-\pi m/10}$ ,  $m = 0, 1, \dots, \nu - 1$ , with unit power, where  $\alpha$  is used to normalized the channel power [30]. In all simulations for the triangularization process with  $|\delta| < 0.5$ , only two successive Gram-Schmidt QR transformations will be executed ( $L = 2$ ) for orders up to  $K = 6$ . For notations, denote  $\text{Var}[\hat{\delta}] = E[(\delta - \hat{\delta})^2] = E\left[\left(\delta - \sum_{m=0}^M \hat{\delta}_m\right)^2\right]$

and  $\text{Var}[\hat{\delta}_{(i)}] = E[(\delta - \hat{\delta}_{(i)})^2] = E\left[\left(\delta - \sum_{m=0}^i \hat{\delta}_m\right)^2\right]$  as the variance or MSE of the final CFO estimator and the  $i$ th interim CFO estimator, respectively.

In Figure 1, assuming a small true CFO of  $\delta = 0.18$  with a 30-dB signal-to-noise ratio (SNR), we present the learning curves of the MSE of the interim CFO estimator (i.e.,  $\text{Var}[\hat{\delta}_{(i)}]$  vs.  $i$ ) for three approximation orders  $K = 1, 2$ , and 4. Note that no Gram-Schmidt QR transformation is executed for the first-order algorithm. From the figure, we see that the first-order algorithm takes one iteration step or one correction cycle ( $i = 1$ ) to reach the steady-state value, while the second and fourth algorithms need two correction cycles ( $i = 2$ ). This may be because the heavy computation load required by higher degree polynomials will encumber the correction process. We have mentioned earlier that when the polynomial degree  $K$  is too high, the triangularization process will begin to give less accurate results [34]. Then, Figure 2 shows the results of repeated experiments using a larger  $\delta = 0.48$ . The result of the sixth-order algorithm is now added. This time,



the first-order algorithm fails to deliver an acceptable result. Again, for the similar reason of computational encumbrance by high-order process, the second-order algorithm has a smoother path to reach the steady state needing two correction iterations ( $i = 2$ ), while

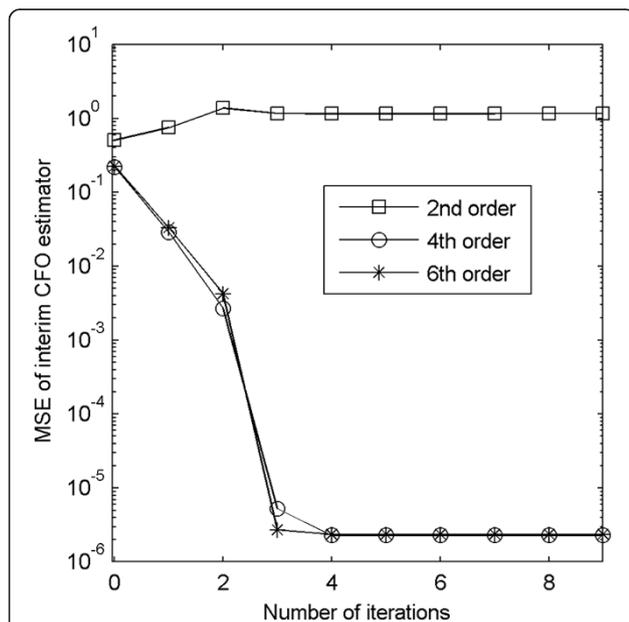


**Figure 2** Learning curves of MSE of interim CFO estimator for approximation algorithms of various orders,  $\delta = 0.48$ , SNR = 30 dB,  $L = 2$ .

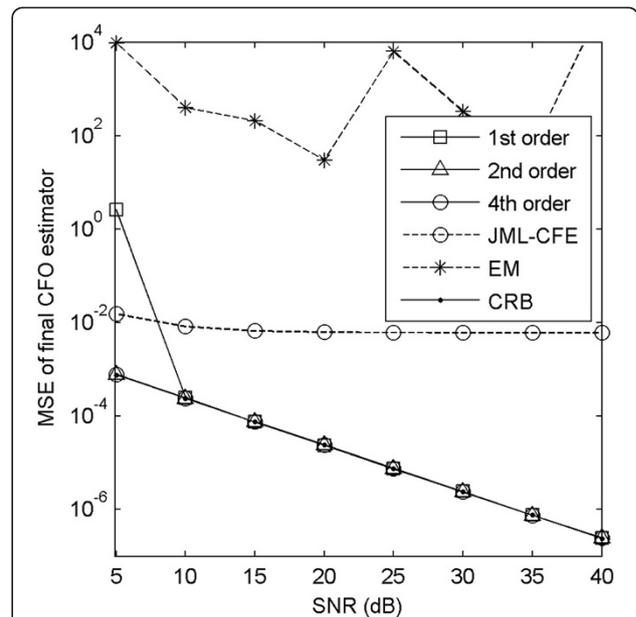
both the fourth- and sixth-order algorithms require one more iterations ( $i = 3$ ) and struggle slightly on way to convergence (the learning curves experience small fluctuations). However, if we increase  $\delta$  further beyond half the carrier spacing ( $|\delta| > 0.5$ ), we shall begin to see the advantage of using higher orders. In Figure 3, the curves using  $\delta = 0.6$  are presented. Now, lower order algorithms ( $K \leq 2$ ) start to diverge. However, the high-order algorithms (fourth and sixth) can still deliver good performance with the higher order (sixth order) doing better.

From Figures 1, 2 and 3, the minimum required correction cycles ensuring convergence for carrier frequency offsets of 0.18, 0.48, and 0.6 are 2, 3, and 4, respectively, for the proposed method. Accordingly, we choose the correction cycle parameter  $M = 4$  for all the subsequent simulations for our proposed method.

Next, over the same channel system with the small frequency drift of  $\delta = 0.18$ , Figures 4 and 5 respectively give the performance curves of the final CFO estimator MSE (i.e.,  $\text{Var}[\hat{\delta}]$  or  $\text{Var}[\hat{\delta}_{(m)}]$ ) vs. SNR and CIR estimator MSE ( $\text{Var}[\hat{\mathbf{h}}]$ ) vs. SNR for various algorithms including those of the EM-based algorithm and JML-CFE algorithm given by [12,14]. For all the estimator MSE performance curves, results are averaged over 10,000 runs. Also incorporated in the figure is the Cramer-Rao bound

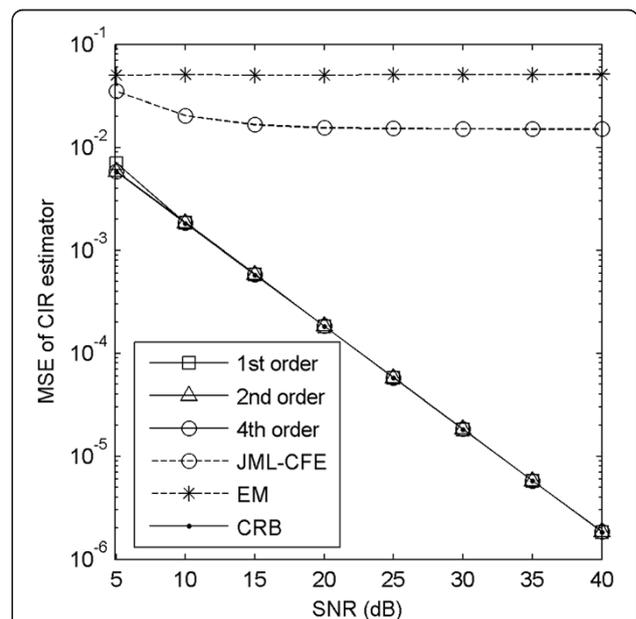


**Figure 3** Learning curves of MSE of interim CFO estimator for approximation algorithms of various orders,  $\delta = 0.6$ , SNR = 30 dB,  $L = 2$ .



**Figure 4** MSE of final CFO estimator vs. SNR curves for various algorithms with  $\delta = 0.18$ .

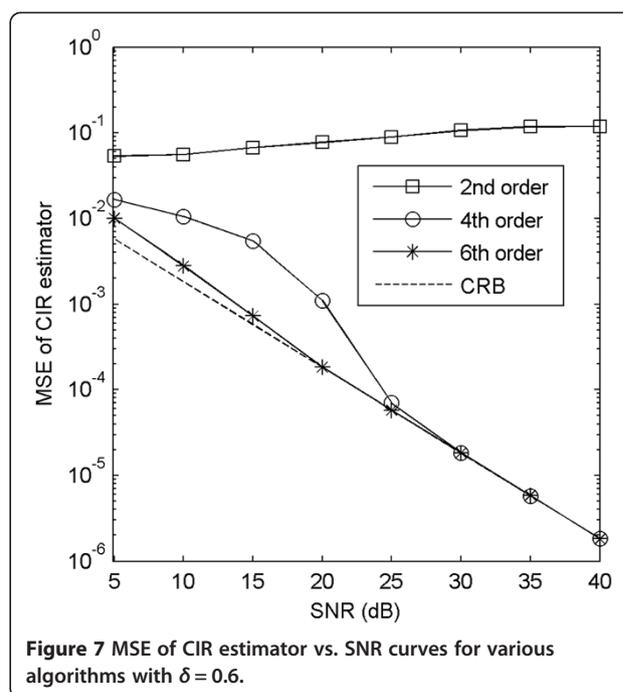
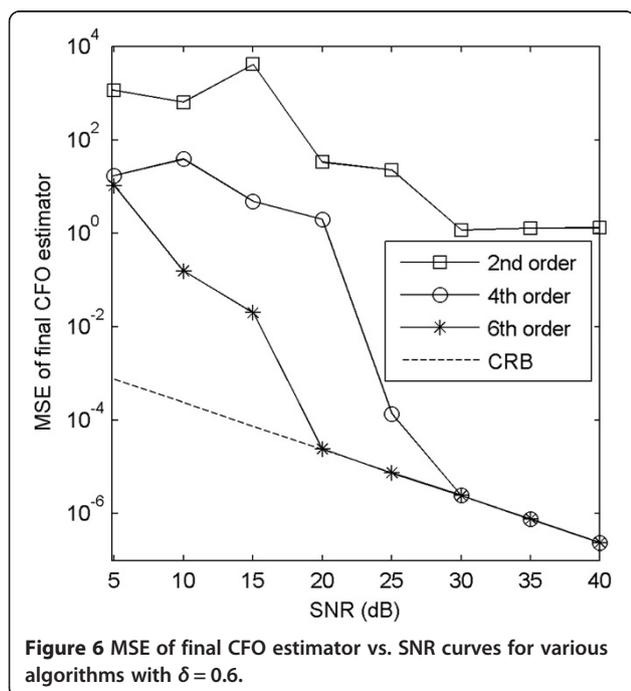
(CRB) vs. SNR curve for performance comparison. The derivations for CRB are presented in the Appendix for clarity. For this small frequency drift, we see that our algorithms of order  $K = 1, 2$ , and 4 all yield comparable performance with estimator MSE curves adhering closely



**Figure 5** MSE of CIR estimator vs. SNR curves for various algorithms with  $\delta = 0.18$ .

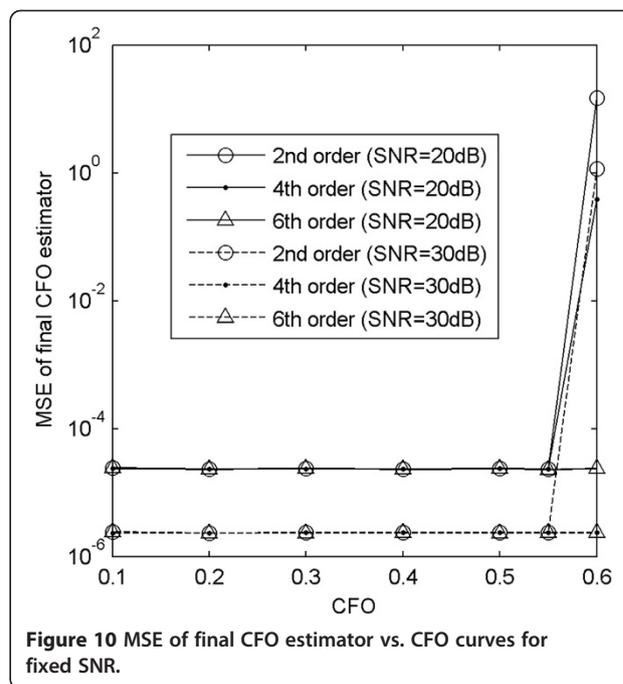
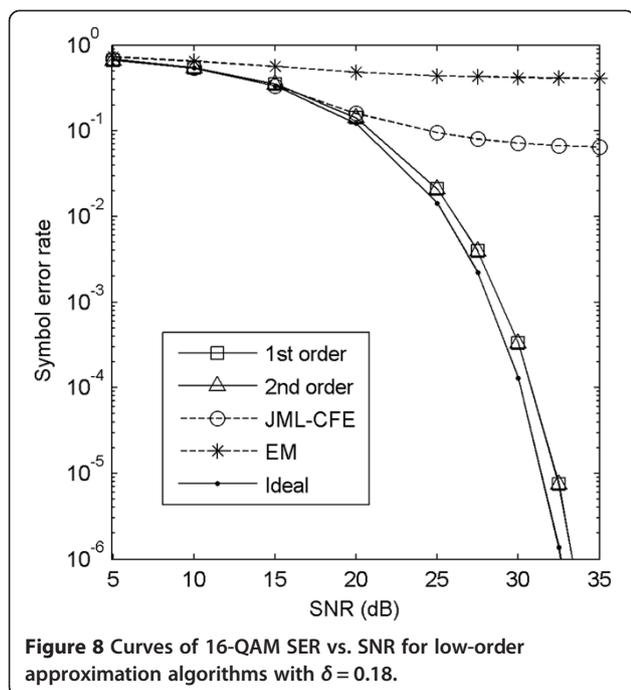
to the CRB curve, and they all outperform the EM-based and the JML-CFE algorithms. In fact, it can be apparently seen that the EM-based algorithm performance is unacceptable. Simulation results show that the EM-based algorithm can work well only with  $\delta < 0.05$ . The reason that the EM-based algorithm works poorly may be due to the fact that the expectation step uses MMSE estimate rather than ML estimate for the CIR in addition to the poor use of low first-order approximation. If we increase  $\delta$  to 0.48, we will find that all the first-order algorithms including ours will all fail to work, while the three high-order algorithms ( $K=2, 4, 6$ ) still perform comparably with estimator MSE curves very closely adhering to the CRB (results not shown for space saving). For  $\delta$  values beyond half the carrier spacing, Figures 6 and 7 respectively plot the curves of final CFO estimator MSE and the CIR estimator MSE against SNR along with the CRB curves for  $\delta=0.6$ . Orders 2, 4, and 6 are given. We can now see that the sixth-order algorithm clearly prevails as both the CFO and CIR estimator MSEs will begin to adhere closely to the CRBs from SNR = 20 dB, while for the fourth-order algorithm, the SNR must be greater than 30 dB for estimator MSEs to closely adhere to the CRBs. For orders  $K \leq 2$ , the algorithm will fail to perform at  $\delta=0.6$  just as in Figure 3.

Although the estimator MSE is an important and very useful performance metric for parameter estimations, the ultimate performance measure in communication systems belongs to error rate. We therefore should



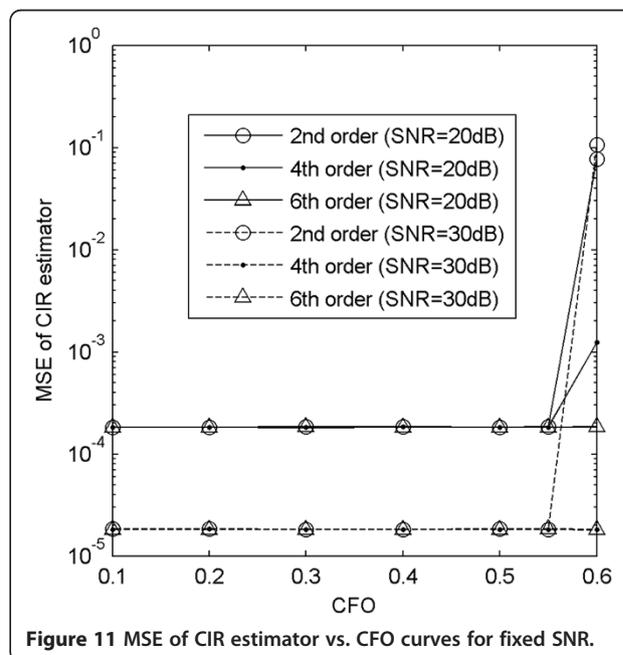
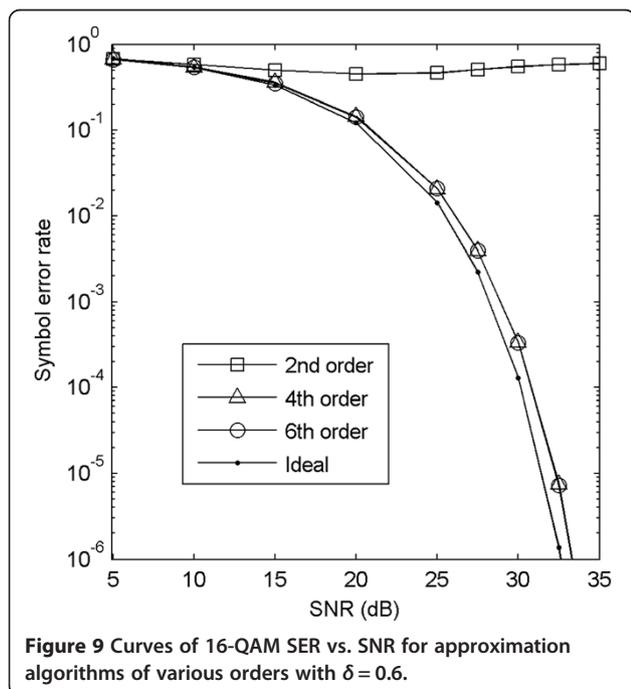
compare the error rate performances of various algorithms. It is well known that OFDM error rate performance relies on the accuracies of frequency and channel estimation [30]. A good tracking scheme should achieve better CFO and CIR estimations hence lower error rate. Employing 16-ary square QAM transmission over the same channel system, with small frequency drift  $\delta=0.18$ , Figure 8 compares SER vs. SNR curves for all the first-order schemes as well as the second-order algorithm. Clearly, both the schemes of [12] and [14] fail to perform satisfactorily, while our first- and second-order algorithms perform comparably well with SER values just slightly higher than an ideal case (perfect CFO synchronization and channel estimation). Then, we assume a larger CFO  $\delta=0.6$ . Now as shown in Figure 9, both our first- and second-order algorithms fail to work. The fourth- and sixth-order algorithms perform comparably well with SER curve slightly higher than the ideal result.

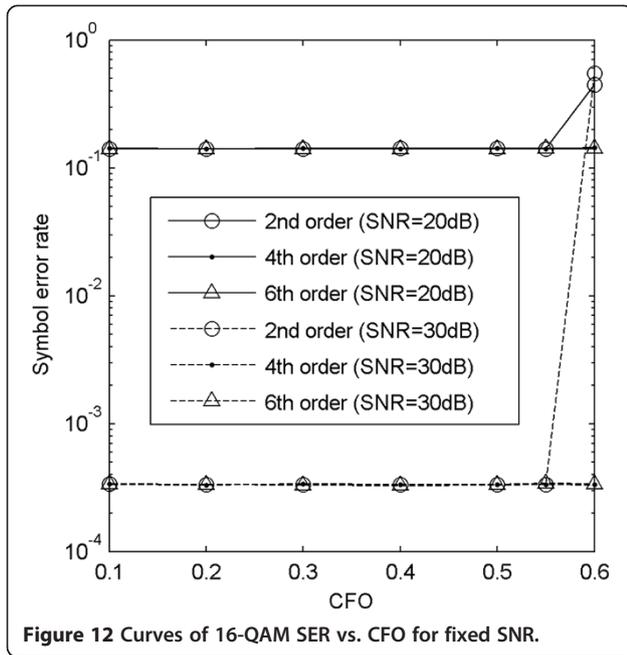
Figures 10 and 11 show the MSE of final CFO estimate vs. CFO  $\delta$  and the MSE of CIR estimator vs. CFO  $\delta$ , respectively, at fixed SNR. Both the figures show that the sixth-order algorithm works well for CFO up to 0.6, while the second-order algorithm performs comparably well with the sixth order for CFO up to 0.55 but fails when CFO approaches 0.6. For the fourth order, when the SNR is greater than 30 dB, it performs comparably well with the sixth order. The results show agreement



with what we get from Figures 4, 5, 6 and 7, in which only two specific CFO values, 0.18 and 0.6, are considered. Figure 12 shows the curves of SER vs. CFO  $\delta$ . It shows agreement with what we get from Figures 8 and 9 that the fourth and the sixth have the comparable

performance in terms of SER up to CFO of 0.6. Notice that seeing from Figures 4, 5, 6, 7, 10 and 11, the fourth order has poor MSEs for both the CFO and CIR estimators working on CFO approaching 0.6 at low SNR, i.e., SNR = 20 dB. However, seeing from





Figures 8, 9 and 12, the SER curves of the fourth order are closely attached to those of the sixth order for CFO up to 0.6 for all values of SNR. This means that, at low SNR which corresponds to high SER, the SER performance makes no difference for the fourth- and sixth-order algorithms even though they have big differences in MSE values at low SNR. Since the ultimate performance depends on SER, instead of MSE, we conclude that the fourth order will suffice for covering the CFO exceeding half the subcarrier spacing.

By observing various simulation results, we can conclude that, for a tracking range up to  $|\delta| < 0.6$ , the fourth-order algorithm should suffice. To cover a wider tracking range, one can use orders higher than 4. However, using orders higher than 4 is not recommended since it will only increase the computation load without much gain

except for a wider tracking range which is rarely necessary for fine frequency tracking. In fact, for fine frequency tracking, the frequency drift ought to be less than half the carrier spacing or  $|\delta| < 0.5$ . The  $|\delta| < 0.5$  range is already a much wider range than those covered by all the existing tracking algorithms cited previously. Our second-order algorithm can easily cover this range with adequate performance. Thus, our proposed algorithm design indeed provides a significant contribution to the OFDM tracking study.

The significant performance improvement provided by our high-order approximation algorithm is not without price. It is conceivable that the higher the order, the higher is the computation complexity and hence more complicated and costly it will be for practical implementation. Depending on the application, a trade-off between complexity and performance may be made. Table 1 evaluates and compares the computation complexities of various algorithms discussed in this paper. It is seen that the complexity of an algorithm is related to channel dispersion length  $\nu$ , OFDM symbol block length  $N$ , order of approximation  $K$ , as well as iteration number  $M$  in a rather complicated way. Generally, higher approximation order will give higher complexity. Usually,  $N$  is much larger than  $\nu$ ,  $K$ , or  $M$  and hence has the dominant impact on the computation complexity. Thus, from Table 1, we can get a rough estimate that, among the first-order approximation algorithms, the EM-based algorithm should have the lowest complexity, but its performance is poorest. Our first-order algorithm may have higher complexity than the EM-based and the JML-CFE algorithms, but not much higher. Therefore, in light of all the gaining in tracking range, estimation accuracy, and error rate performance, it should be well worthwhile to adopt our algorithm.

## 5. Conclusions

To track frequency drift and channel state variation in OFDM systems over mobile wireless channels, we propose

**Table 1 Computation complexity comparison between various tracking algorithms**

	Real multiplication	Real addition	Real division	Square roots
EM	$(M+1)(8\nu N+13N-1)$	$2(M+1)(2\nu N+N-\nu-2)$	$M+1$	0
JML-CFE	$2N(13N+2\nu+8)$	$12N^2+2\nu N-3N-2\nu-6$	1	0
The first-order algorithm	$(M+1)(8N^2+13N-1)+4\nu N$	$(M+1)(4N^2-3)+2\nu(N-1)$	$M+1$	0
The $K$ th-order algorithm ( $K \geq 2$ )	$(M+1)\{(20K+8)N^2+(5K+4)N+K[L(3K^2-1)-1]\}+4\nu N$	$(M+1)\{2(2K+1)N^2-2(K+1)N+L(K-1)(3K^2+K/2-1)\}+M+2\nu(N-1)$	$(M+1)[LK(3K-1)/2+K]$	$(M+1)LK$

a high-order approximation algorithm using the MLE scheme which outperforms existing OFDM tracking algorithms in terms of tracking range, estimation accuracy, as well as SER performance. Computer simulation results show that a fourth-order algorithm will suffice to cover a wide tracking range exceeding half the carrier spacing while producing results with good accuracy. A second-order algorithm will suffice if the tracking range is less than half the carrier spacing.

## Appendix

### Cramer-Rao bounds

Define the following real vectors by stacking the real and imaginary parts of corresponding complex vectors as

$$\bar{\mathbf{r}} = \left[ \text{Re}\{\mathbf{r}\}^T, \text{Im}\{\mathbf{r}\}^T \right]^T, \quad (26)$$

$$\begin{aligned} \bar{\mathbf{m}} &= \frac{1}{\sqrt{N}} \left[ \text{Re}\{\mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v \mathbf{h}\}^T, \text{Im}\{\mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v \mathbf{h}\}^T \right]^T \\ &= [\bar{\mathbf{m}}_1^T, \bar{\mathbf{m}}_2^T]^T, \end{aligned} \quad (27)$$

$$\bar{\boldsymbol{\theta}} = \left[ \text{Re}\{\mathbf{h}\}^T, \text{Im}\{\mathbf{h}\}^T, \delta \right]^T. \quad (28)$$

Then, the log-likelihood function of (7) can be rewritten as

$$\ln \Lambda = -N \ln(\pi \sigma_w^2) - \frac{1}{\sigma_w^2} \|\bar{\mathbf{r}} - \bar{\mathbf{m}}\|^2. \quad (29)$$

The fisher information matrix for the real vector  $\bar{\boldsymbol{\theta}}$  is given by [35]

$$\mathbf{J}(\bar{\boldsymbol{\theta}}) = \frac{2}{\sigma_w^2} \text{Re} \left\{ \begin{bmatrix} \mathbf{F}_v^H \mathbf{X}^H \mathbf{X} \mathbf{F}_v & j \mathbf{F}_v^H \mathbf{X}^H \mathbf{X} \mathbf{F}_v & \frac{j2\pi}{N^2} \mathbf{F}_v^H \mathbf{X}^H \mathbf{F}_N \mathbf{Q} \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v \mathbf{h} \\ -j \mathbf{F}_v^H \mathbf{X}^H \mathbf{X} \mathbf{F}_v & \mathbf{F}_v^H \mathbf{X}^H \mathbf{X} \mathbf{F}_v & \frac{2\pi}{N^2} \mathbf{F}_v^H \mathbf{X}^H \mathbf{F}_N \mathbf{Q} \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v \mathbf{h} \\ \frac{j2\pi}{N^2} \mathbf{h}^H \mathbf{F}_v^H \mathbf{X}^H \mathbf{F}_N \mathbf{Q} \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v & -\frac{2\pi}{N^2} \mathbf{h}^H \mathbf{F}_v^H \mathbf{X}^H \mathbf{F}_N \mathbf{Q} \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v & \frac{4\pi^2}{N^3} \mathbf{h}^H \mathbf{F}_v^H \mathbf{X}^H \mathbf{F}_N \mathbf{Q}^2 \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v \mathbf{h} \end{bmatrix} \right\}. \quad (36)$$

$$\mathbf{J}(\bar{\boldsymbol{\theta}}) = E \left[ \frac{\partial \ln \Lambda}{\partial \bar{\boldsymbol{\theta}}} \left( \frac{\partial \ln \Lambda}{\partial \bar{\boldsymbol{\theta}}} \right)^T \right], \quad (30)$$

from which the  $(l, l')$ th element is given by

$$\begin{aligned} \mathbf{J}_{l,l'} &= \frac{2}{\sigma_w^2} \frac{\partial \bar{\mathbf{m}}^T}{\partial \theta_l} \left( \frac{\partial \bar{\mathbf{m}}}{\partial \theta_{l'}} \right) = \frac{2}{\sigma_w^2} \text{Re} \left\{ \frac{\partial(\bar{\mathbf{m}}_1 + j\bar{\mathbf{m}}_2)^H}{\partial \theta_l} \left( \frac{\partial(\bar{\mathbf{m}}_1 + j\bar{\mathbf{m}}_2)}{\partial \theta_{l'}} \right) \right\} \\ &= \frac{2}{N\sigma_w^2} \text{Re} \left\{ \frac{\partial(\mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v \mathbf{h})^H}{\partial \theta_l} \left( \frac{\partial(\mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v \mathbf{h})}{\partial \theta_{l'}} \right) \right\}, \end{aligned} \quad (31)$$

Where  $\bar{\theta}_l$  denotes the  $l$ th element of  $\bar{\boldsymbol{\theta}}$ . Alternatively, using (A6), we can write

$$\mathbf{J}(\bar{\boldsymbol{\theta}}) = \frac{2}{N\sigma_w^2} \text{Re} \left\{ \frac{\partial(\mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v \mathbf{h})^H}{\partial \bar{\boldsymbol{\theta}}} \frac{\partial \mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v \mathbf{h}}{\partial \bar{\boldsymbol{\theta}}^T} \right\}. \quad (32)$$

Now, using the following facts

$$\frac{\partial \mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v \mathbf{h}}{\partial \text{Re}\{\mathbf{h}\}^T} = \mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v, \quad (33)$$

$$\frac{\partial \mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v \mathbf{h}}{\partial \text{Im}\{\mathbf{h}\}^T} = j \mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v, \quad (34)$$

$$\frac{\partial \mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v \mathbf{h}}{\partial \delta} = j \frac{2\pi}{N} \mathbf{Q} \mathbf{D}_\delta \mathbf{F}_N^H \mathbf{X} \mathbf{F}_v \mathbf{h}, \quad (35)$$

the Fisher information of (32) can be expressed as

The Cramer-Rao bounds (CRBs) can then be obtained from the diagonal elements of  $J^{-1}$ . Note that the CRBs obtained here are jointly for the ML estimates of CIR  $\mathbf{h}$  and CFO  $\delta$  (i.e.,  $\hat{\mathbf{h}}$  and  $\hat{\delta}$ ).

#### Abbreviations

CFO: carrier frequency offset; CFR: channel frequency response; CIR: channel impulse response; DFT: discrete Fourier transform; EM: expectation-maximization; i.i.d: independent, identically distributed; JML-CFE: joint maximum-likelihood channel and frequency estimation; ML: maximum-likelihood; MLE: ML estimation; OFDM: orthogonal frequency division multiplexing; RLS: recursive least squares; RV: random variable; SER: symbol error rate.

#### Competing interests

The authors declare that they have no competing interests.

#### Author details

<sup>1</sup>Department of Electrical Engineering, Tamkang University, No. 151, Yingzhuang Rd., Danshui Dist, New Taipei City 25137, Taiwan. <sup>2</sup>Department of Electrical Engineering, Fu Jen Catholic University, No. 510, Zhongzheng Rd., Xinzhuang Dist, New Taipei City 24205, Taiwan.

Received: 18 July 2013 Accepted: 6 January 2014

Published: 27 January 2014

#### References

1. T Keller, L Piazzi, P Mandarini, L Hanzo, Orthogonal frequency division multiplex synchronization techniques for frequency-selective fading channels. *IEEE J. Selected Areas Commun.* **19**(6), 999–1008 (2001)
2. PH Moose, A technique for orthogonal frequency division multiplexing frequency offset correction. *IEEE Trans. Commun.* **42**(10), 2908–2914 (2001)
3. JJ van de Beek, M Sandell, PO Borjesson, ML estimation of time and frequency offset in OFDM systems. *IEEE Trans. Signal Proces.* **45**(7), 1800–1805 (1997)
4. TM Schmidl, DC Cox, Robust frequency and timing synchronization for OFDM. *IEEE Trans. Commun.* **45**(12), 1613–1621 (1997)
5. YH Kim, JH Lee, Joint maximum likelihood estimation of carrier and sampling frequency offsets for OFDM systems. *IEEE Trans. Broadcast.* **57**(2), 277–283 (2011)
6. WL Chin, ML estimation of timing and frequency offsets using distinctive correlation characteristics of OFDM signals over dispersive fading channels. *IEEE Trans. Vehicular Technol.* **60**(2), 444–456 (2011)
7. YS Choi, PJ Voltz, FA Cassara, ML estimation of carrier frequency offset for multicarrier signals in Rayleigh fading channels. *IEEE Trans. Vehicular Technol.* **50**(2), 644–655 (2001)
8. BB Salberg, A Swami, Doppler and frequency-offset synchronization in wideband OFDM. *IEEE Trans. Wireless Commun.* **4**(6), 2870–2881 (2005)
9. M Morelli, U Mengali, Carrier-frequency estimation for transmissions over selective channels. *IEEE Trans. Commun.* **48**(9), 1580–1589 (2000)
10. X Ma, H Kobayashi, SC Schwartz, *Joint frequency offset and channel estimation for OFDM. Proceedings of the IEEE GLOBECOM 2003, vol. 1* (IEEE, Piscataway, San Francisco, 2003), pp. 15–19
11. L Zheng, W Cheng, J Hu, D Yuan, *The ML frequency tracking algorithm for OFDM systems based on pilot symbols and decision data. Proceedings of the IEEE International Symposium on Microwave, Antenna, Propagation, and EMC Technologies for Wireless Communications, vol. 2* (IEEE, Piscataway, Beijing, 2005), pp. 1611–1614
12. JH Lee, JC Han, SC Kim, Joint carrier frequency synchronization and channel estimation for OFDM systems via the EM algorithm. *IEEE Trans. Vehicular Technol.* **55**(1), 167–172 (2006)
13. MO Pun, M Morelli, CCJ Kuo, Maximum-likelihood synchronization and channel estimation for OFDMA uplink transmissions. *IEEE Trans. Commun.* **54**(4), 726–736 (2006)
14. FZ Merli, GM Vitetta, Iterative ML-based estimation of carrier frequency offset, channel impulse response and data in OFDM transmissions. *IEEE Trans. Commun.* **56**(3), 497–506 (2008)
15. JJ van de Beek, O Edfors, M Sandell, SK Wilson, PO Borjesson, *On channel estimation in OFDM systems. Proceedings of the IEEE 45th Vehicular Technology Conference* (IEEE, Piscataway, Chicago, 1995), pp. 815–819

16. O Edfors, M Sandell, JJ van de Beek, SK Wilson, PO Borjesson, OFDM channel estimation by singular value decomposition. *IEEE Trans. Commun.* **46**(7), 931–939 (1998)
17. Y Li, Pilot-symbol-aided channel estimation for OFDM in wireless systems. *IEEE Trans. Vehicular Technol.* **49**(4), 1207–1215 (2000)
18. Y Li, Simplified channel estimation for OFDM systems with multiple transmit antennas. *IEEE Trans. Wireless Commun.* **1**(1), 67–75 (2002)
19. Y Li, JH Winters, NR Sollenberger, MIMO-OFDM for wireless communications: signal detection with enhanced channel estimation. *IEEE Trans. Commun.* **50**(9), 1471–1477 (2002)
20. S Coleri, M Ergen, A Puri, A Bahai, Channel estimation techniques based on pilot arrangement in OFDM systems. *IEEE Trans. Broadcast.* **48**(3), 223–229 (2002)
21. H Nguyen-Le, T Le-Ngoc, CC Ko, RLS-based joint estimation and tracking of channel response, sampling, and carrier frequency offsets for OFDM. *IEEE Trans. Broadcast.* **55**(1), 84–94 (2009)
22. M Jamalabdollahi, S Salari, B Zahedi, *Efficient joint frequency synchronization and channel estimation in OFDM systems. Proceedings of the IEEE 19th Conference on Signal Processing and Communications Applications* (IEEE, Piscataway, Antalya, 2011), pp. 206–209
23. Y Liu, Z Tan, H Wang, KS Kwak, Joint estimation of channel impulse response and carrier frequency offset for OFDM Systems. *IEEE Trans. Vehicular Technol.* **60**(9), 4645–4650 (2011)
24. S Salari, RS Kandovan, *Joint frequency and channel estimation for MIMO-OFDM systems via the EM algorithm. Proceedings of the IEEE Wireless and Optical Communications Networks* (IEEE, Piscataway, Colombo, 2010), pp. 1–5
25. EP Simon, L Ros, H Hijazi, M Ghogho, Joint carrier frequency offset and channel estimation for OFDM systems via the EM algorithm in the presence of very high mobility. *IEEE Trans. Signal Proces.* **60**(2), 754–765 (2012)
26. G Sewell, *Computational Methods of Linear Algebra*, 2nd edn. (Wiley, New York, 2005)
27. GH Golub, CF Van Loan, *Matrix Computations*, 3rd edn. (The Johns Hopkins University Press, Maryland, 1996)
28. JFG Francis, The QR transformation, a unitary analogue to the LR transformation-part 1. *Comp. J.* **4**(3), 265–271 (1961)
29. JG Proakis, M Salehi, *Digital Communications*, 5th edn. (McGraw-Hill, New York, 2008)
30. RY Yen, HY Liu, WK Tsai, QAM symbol error rate in OFDM systems over frequency-selective fast Ricean fading channels. *IEEE Trans. Vehicular Technol.* **57**(2), 1322–1325 (2008)
31. M Morelli, CJ Kuo, MO Pun, Synchronization techniques for orthogonal frequency division multiple access (OFDMA): a tutorial review. *Proc. IEEE* **95**(7), 1394–1427 (2007)
32. P Stoica, O Besson, Training sequence design for frequency offset and frequency-selective channel estimation. *IEEE Trans. Commun.* **51**(11), 1910–1917 (2003)
33. DC Chu, Polyphase codes with good periodic correlation properties. *IEEE Trans. Info. Theory.* **18**(4), 531–532 (1972)
34. S Haykin, *Adaptive Filter Theory*, 4th edn. (Prentice Hall, New Jersey, 2002)
35. SM Kay, *Fundamentals of Statistical Signal Processing, Volume I: Estimation Theory* (Prentice Hall, New Jersey, 1993)

doi:10.1186/1687-1499-2014-16

**Cite this article as:** Yen et al.: High-order approximation algorithm using Gram-Schmidt QR transformations coupled with an iterative correction process for tracking frequency drift in OFDM systems. *EURASIP Journal on Wireless Communications and Networking* 2014 **2014**:16.