

RESEARCH

Open Access



Fast basis search for adaptive Fourier decomposition

Ze Wang¹, Feng Wan^{1*} , Chi Man Wong¹ and Tao Qian²

Abstract

The adaptive Fourier decomposition (AFD) uses an adaptive basis instead of a fixed basis in the rational analytic function and thus achieves a fast energy convergence rate. At each decomposition level, an important step is to determine a new basis element from a dictionary to maximize the extracted energy. The existing basis searching method, however, is only the exhaustive searching method that is rather inefficient. This paper proposes four methods to accelerate the AFD algorithm based on four typical optimization techniques including the unscented Kalman filter (UKF) method, the Nelder-Mead (NM) algorithm, the genetic algorithm (GA), and the particle swarm optimization (PSO) algorithm. In the simulation of decomposing four representative signals and real ECG signals, compared with the existing exhaustive search method, the proposed schemes can achieve much higher computation speed with a fast energy convergence, that is, in particular, to make the AFD possible for real-time applications.

Keywords: Adaptive Fourier decomposition, Unscented Kalman filter, Nelder-Mead algorithm, Genetic algorithm, Particle swarm optimization algorithm

1 Introduction

The adaptive Fourier decomposition (AFD), introduced by Qian et. al., is a type of positive frequency expansion algorithm based on a given basis search dictionary [1–3]. It offers fast energy decomposition via the adaptive basis, which is different from the conventional Fourier decomposition that is based on the Fourier basis. All the traditional methods, including the wavelet one, are of the same nature. Accordingly, the AFD has been successfully applied to the system identification, modeling, signal compression and denoising [4–9].

The AFD is based on the rational orthogonal system, $\{B_n\}_{n=1}^\infty$ where

$$B_n(e^{jt}) = \frac{\sqrt{1 - |a_n|^2}}{1 - \bar{a}_n e^{jt}} \prod_{k=1}^{n-1} \frac{e^{jt} - a_k}{1 - \bar{a}_k e^{jt}}, \quad (1)$$

$a_n \in \mathbb{D}$ ($n = 1, 2, \dots$), $\mathbb{D} = \{z \in \mathbb{C} : |z| < 1\}$, and \mathbb{C} is the complex plane [1]. For a given signal $G(e^{jt})$ in H^2 space,

the core AFD, as basis of the other AFD methods and itself is often abbreviated as AFD, expresses $G(e^{jt})$ as

$$G(e^{jt}) = \sum_{n=1}^N \langle G_n(e^{jt}), e_{\{a_n\}} \rangle B_n(e^{jt}) + R_N(e^{jt}) \quad (2)$$

where $R_N(e^{jt})$ denotes the standard remainder at the decomposition level N , and $G_n(e^{jt})$ denotes the reduced remainder at the decomposition level n that is defined as

$$G_n(e^{jt}) = R_{n-1}(e^{jt}) \cdot \prod_{l=1}^{n-1} \frac{1 - \bar{a}_l e^{jt}}{e^{jt} - a_l}, \quad (3)$$

$e_{\{a_n\}}$ is the evaluator at a_n that is

$$e_{\{a_n\}} = \frac{\sqrt{1 - |a_n|^2}}{1 - \bar{a}_n e^{jt}}, \quad (4)$$

and $\langle G_n(e^{jt}), e_{\{a_n\}} \rangle$ denotes the inner product of $G_n(e^{jt})$ and $e_{\{a_n\}}$ in L^2 space. The most important step at each decomposition level is to determine a suitable a_n to achieve a fast energy convergence rate. In the AFD, the maximal selection principle (MSP) is applied to identify such a_n by solving the following optimization problem:

*Correspondence: fwan@umac.mo

¹Department of Electrical and Computer Engineering, Faculty of Science and Technology, University of Macau, Avenida da Universidade, Taipa, Macau, China
Full list of author information is available at the end of the article

$$\begin{aligned} & \text{maximize } A_{G_n}^2(a_n) = \left| \left| G_n(e^{jt}), e_{\{a_n\}} \right| \right|^2, \\ & \text{subject to } |a_n| < 1. \end{aligned} \quad (5)$$

This process to get $G_{n+1}(e^{jt})$ from $G_n(e^{jt})$ through the MSP is called maximal sifting [1]. The convergence, convergence rate, and robustness of the AFD have been theoretically proved in [1, 10, 11]. For convenience, $B_n(e^{jt})$, $G(e^{jt})$, $G_n(e^{jt})$, and $R_n(e^{jt})$ are abbreviated as B_n , G , G_n , and R_n in the following equations.

There are several versions of the AFD, including the core AFD, the unwinding AFD, and the cyclic AFD, proposed in literature to determine the suitable a_n array [2, 12, 13]. The key decomposition strategy of the unwinding AFD coincides with the study of the nonlinear phase unwinding of functions by Coifman et al. [14, 15]. Although they adopt different decomposition processes or basis representations to improve the computation efficiency, these versions of the AFD all require to implement the MSP. Until now, the most common implementation for the MSP in the AFD is the exhaustive search method [1, 3, 4, 7–9, 12, 13]. In the exhaustive search, the parameters a_1, \dots, a_n , as indices of the selected dictionary elements, are selected according to the MSP in the one by one manner.

To make sure that the searched result closely gives rise to the global optimum, the density of the search dictionary should be sufficiently high. Since the objective function in (5) is highly nonlinear and complicated, this exhaustive and inefficient strategy would be usually time consuming, which seriously limits the practicability of the AFD. It turns out to be a crucial problem of the implementation of the AFD. Besides the above mentioned versions of the AFD, Plonka et al. proposed a sparse approximation of the exponential summations for the adaptive Fourier series which can estimate the almost optimal basis, and thus can provide significantly good convergence behavior [16]. However, this paper will not focus on proposing a new algorithm for the decomposition based on the Takenaka-Malmquist system but focus on improving the computation efficiency of the AFD by improving the searching strategy in the MS.

Normally, the objective function in (5) is highly nonlinear and contains an uncertain term G_n that varies for different input signals at different decomposition levels. Therefore, calculating the gradient information and objective function values of (5) is complicated and time consuming. In our previous work shown in [6], a preliminary study of applying the NM method to improve the computation efficiency of the AFD shows that the NM method can reduce the computation time of the AFD to the half of that based on the conventional exhaustive search method. However, in [6], the performance of the NM method is only verified by one kind of special

signals, i.e., the ECG signals, with non-optimal parameter selection, and is only compared with that of the exhaustive search. Besides our previous work, Kirkbas et al. propose the Jaya-based AFD method for reducing the computation time [17]. Thanks to the advanced searching strategy and remarkable convergence speed of the Jaya method that is one kind of the novel population-based heuristic optimization methods, the Jaya-based AFD method can provide faster computation speed comparing to the conventional method with the accurate signal representation. Similarly with our previous work, the performance of the Jaya-based AFD method is only verified by one cosine signal and one kind of specific signals, i.e. speech signals, and only compared with the conventional method. In this paper, four typical optimization algorithms that require neither the gradient information nor too many function evaluations are reviewed and adopted to determine each successive a_n in the AFD, including the unscented Kalman filter (UKF) method which is based on the deterministic sampling, the NM algorithm which is a simplex method, and the genetic algorithm (GA) as well as the particle swarm optimization (PSO) algorithm which belong to the stochastic search. In order to apply these methods, the optimization problem in (5) is reformulated from a maximization problem with one complex-valued variable to a minimization problem with two real-valued variables. The performances of these proposed methods in the sifting of the AFD are compared with the conventional exhaustive search method in the decomposition of four representative signals, including the heavisine signal, the doppler signal, the block signal, and the bump signal. These signals are chosen because they caricature spatially variable functions arising in imaging, spectroscopy, and other scientific signal processing [18]. In addition, to verify the performance of the proposed methods for real signals, simulations are also carried out for real ECG signals from the MIT-BIH Arrhythmia Database [19, 20]. Simulation results show that, compared with the existing exhaustive search method, all these proposed four optimization methods can provide higher computation speed with a fast energy convergence rate. In addition, the UKF method performs best among all the tested algorithms.

The rest of this paper is organized as follows. In Section 2.1, the reformulated optimization problem and the method for determining the initial points are proposed. In addition, a brief review of the above mentioned optimization methods, i.e., the NM method, the UKF method, the GA, and the PSO algorithm, is provided. Section 3 and Section 4 show effects of optimization parameters and comparison results of these acceleration methods in simulations as well as the detail computation results of the UKF method. Finally, the conclusion is given in Section 5.

2 Proposed implementation method and simulation settings

2.1 Efficient implementation of basis search for AFD

2.1.1 Maximal sifting problem reformulation

The optimization problem of the MSP in (5) is a maximum problem with complex-valued variables. However, the selected optimization methods, including the NM algorithm, the UKF method, the GA, and the PSO algorithm, are all designed for the minimization problem with real-valued variables. Therefore, the original optimization problem shown in (5) needs to be further adjusted.

For the NM algorithm, the GA, and the PSO algorithm, since the objective function $A_{G_n}^2(a_n)$ only contains non-negative function values, finding the global maximum of $A_{G_n}^2$ is equivalent to finding the global minimum of $-A_{G_n}^2$. Moreover, $A_{G_n}^2(a_n)$ is mainly determined by the magnitude ρ_n and the phase α_n of a_n , and thus the corresponding equivalent minimization problem with real-valued variables can be expressed as

$$\begin{aligned} & \text{minimize } Y_1(\rho_n, \alpha_n) = - \left\| \left\langle G_n, e^{\{\rho_n e^{j\alpha_n}\}} \right\rangle \right\|^2, \\ & \text{subject to } 0 \leq \rho_n < 1 \text{ and } 0 \leq \alpha_n < 2\pi. \end{aligned} \quad (6)$$

For the UKF method, since it requires that the objective function values are non-negative, the reformulated optimization problem shown in (6) is not further suitable. According to the orthogonal property of $\{B_n\}_{n=1}^\infty$, (5) is equivalent to

$$\begin{aligned} & \text{minimize} \\ & Y_2(\rho_n, \alpha_n) = \|G\|^2 - \sum_{m=1}^n \left\| \left\langle G_m, e^{\{\rho_m e^{j\alpha_m}\}} \right\rangle \right\|^2, \\ & \text{subject to } 0 \leq \rho_n < 1 \text{ and } 0 \leq \alpha_n < 2\pi, \end{aligned} \quad (7)$$

which can be applied for the UKF method.

2.1.2 Determination of initial points

For the selected optimization methods, initial points are important for the optimization performance. In the NM algorithm and the UKF method, to determine suitable initial points, a coarse search step is applied. First, a set of $(\rho_{n,k}, \alpha_{n,k})$ in the search range $0 \leq \rho_{n,k} < 1$ are selected randomly with the uniform distribution where $k = 1, 2, 3, \dots, N_{\text{rand}}$ and N_{rand} denote the total number of points in the dictionary of determining suitable initial points. Then, the objective function values at these points $Y(\rho_{n,k}, \alpha_{n,k})$ are evaluated. Finally, points at which the objective function contains small values are selected as the initial points. Since the initial points are only required to approximate to the point at which the objective function

achieves the global optimum, the number of points to be evaluated can be much smaller compared to that in the conventional exhaustive search method. In the GA and the PSO algorithm, such kind of the coarse initial point searching has already been included in their stochastic searching process. Therefore, the dictionary of determining suitable initial points in the GA and the PSO algorithm is equivalent to that of individuals, i.e., a set of $(\rho_{n,k}, \alpha_{n,k})$ where $k = 1, 2, 3, \dots, N_{\text{ind}}$, and N_{ind} denotes the total number of individuals.

In addition, for some specific types of signals, such as electrocardiography (ECG) signals, the distributions of a_n have already been recognized, which can be considered as the pre-knowledge for searching suitable a_n [6, 7]. Accordingly, the number of points in the coarse search process for initial points can be further reduced [7]. More specifically, suppose the distribution range of a_n is known, which is that the phase search range of a_n is limited into $[\alpha_{\min}, \alpha_{\max}]$, and the magnitude search range of a_n is limited into $[\rho_{\min}, \rho_{\max}]$, the k th point in the dictionary for searching initial points can be computed as

$$\begin{cases} \rho_{n,k} = \sqrt{(\rho_{\max}^2 - \rho_{\min}^2) u_k + \rho_{\min}^2} \\ \alpha_{n,k} = (\alpha_{\max} - \alpha_{\min}) v_k + \alpha_{\min} \end{cases} \quad (8)$$

where u_k and v_k are two random numbers in $[0, 1)$. The distributions of u_k and v_k follow the distributions of a_n . Suppose only the searching range of a_n is known, u_k and v_k can be assumed as the uniform distribution to achieve the maximum entropy and thus cover most points in the searching range [21]. Since the following simulations are carried out mainly for comparing performances of optimization methods, this strategy is not applied. However, for real applications, the distribution of a_n can be recognized first to further reduce the computation time of searching the initial points. Moreover, since the evaluations of the objective function at different points are not interrupted with each other, the parallel computing can be adopted to enhance the computing speed of the searching the initial points. However, this paper is mainly to verify the effects of parameters and the performances of following optimization methods for the computation efficiency of the AFD. Therefore, in the following simulations, the parallel computing is not adopted.

In the next section, the NM algorithm, the UKF method, the GA, and the PSO algorithm will be reviewed, which will be adopted to solve (6). The pseudocode of the AFD based on the four optimization algorithms is shown in Algorithm 1. Although this implementation is based on the core AFD, these optimization algorithms can also be applied for the unwinding AFD and the cyclic AFD.

Algorithm 1 AFD based on the NM algorithm, the UKF method, the GA, or the PSO algorithm.

Input:

- $y(t)$: testing signal;
 N_{decom} : maximum decomposition level;
 Parameters required by optimization algorithms.
- 1: Initialization $a_1 = 0$, $N = 1$, $e_{\{a_1\}} = 1$, $B_1 = 1$;
 - 2: Transfer $y(t)$ to its analytic representation in H^2 space as G_1
 - 3: **if** NM algorithm or UKF method is applied **then**
 - 4: Use (8) to generate the dictionary \mathbf{D} for searching initial points;
 - 5: **else if** GA or PSO algorithm **then**
 - 6: Use (8) to generate the dictionary of individuals;
 - 7: **end if**
 - 8: **while** $N \leq N_{\text{decom}}$ **do**
 - 9: $G_{N+1} \leftarrow (G_N - \langle G_N, e_{\{a_N\}} \rangle e_{\{a_N\}}) \frac{1 - \bar{a}_N e^{it}}{e^{it} - a_N}$;
 - 10: **if** NM algorithm or UKF method is applied **then**
 - 11: Search initial points at which the objective function values in (6) are small from \mathbf{D} ;
 - 12: **end if**
 - 13: **if** NM algorithm, GA, or PSO algorithm **then**
 - 14: Search the result $(\rho_{\text{opt}}, \alpha_{\text{opt}})$ of (6);
 - 15: **else if** UKF method **then**
 - 16: Search the result $(\rho_{\text{opt}}, \alpha_{\text{opt}})$ of (7);
 - 17: **end if**
 - 18: $a_{N+1} \leftarrow \rho_{\text{opt}} e^{j\alpha_{\text{opt}}}$;
 - 19: $B_{N+1} \leftarrow \frac{\sqrt{1 - |a_{N+1}|^2}}{1 - \bar{a}_{N+1} e^{it}} \frac{e^{it} - a_N}{\sqrt{1 - |a_N|^2}} B_N$;
 - 20: $F_{N+1} \leftarrow \langle G_{N+1}, e_{\{a_{N+1}\}} \rangle B_{N+1}$;
 - 21: $N \leftarrow N + 1$;
 - 22: **end while**

Output:

F_n : decomposition component at n decomposition level where $n = 1, 2, 3, \dots, N_{\text{decom}}$.

minimum. The searching strategy is shown in Algorithm 2 [22, 24].

Algorithm 2 Searching strategy of the NM algorithm.

Input:

- $Y_1(\rho, \alpha)$: objective function shown in (6);
 $\{(\rho_k, \alpha_k)\}_{k=1, \dots, N_{\text{rand}}}$: set of initial points;
 N_{iter} : maximum iteration number.
- 1: **repeat**
 - 2: **for** each point **do**
 - 3: Denote this point and its closest two points as \mathbf{B} , \mathbf{G} and \mathbf{W} where $Y_1(\mathbf{B}) \leq Y_1(\mathbf{G}) \leq Y_1(\mathbf{W})$;
 - 4: Set
 $\mathbf{R} = 2\mathbf{M} - \mathbf{W}$, $\mathbf{E} = 2\mathbf{R} - \mathbf{M}$,
 $\mathbf{C}_1 = (\mathbf{W} + \mathbf{M})/2$, $\mathbf{C}_2 = (\mathbf{M} + \mathbf{R})/2$,
 $\mathbf{S} = (\mathbf{B} + \mathbf{W})/2$
 as shown in Fig. 1;
 - 5: **if** $Y_1(\mathbf{R}) < Y_1(\mathbf{W})$ **then**
 - 6: $\mathbf{W} \leftarrow \arg \min \{Y_1(\mathbf{X}) : \mathbf{X} \in \{\mathbf{R}, \mathbf{E}\}\}$;
 - 7: **else if** $\min \{Y_1(\mathbf{C}_1), Y_1(\mathbf{C}_2)\} < Y_1(\mathbf{W})$ **then**
 - 8: $\mathbf{W} \leftarrow \arg \min \{Y_1(\mathbf{X}) : \mathbf{X} \in \{\mathbf{C}_1, \mathbf{C}_2\}\}$;
 - 9: **else**
 - 10: $\mathbf{W} \leftarrow \mathbf{S}$ and $\mathbf{G} \leftarrow \mathbf{M}$;
 - 10: **end if**
 - 11: **end for**
 - 12: **until** the convergence is finished or the iteration number has reached N_{iter} .

Output:

\mathbf{B} : approximated minimum point.

The NM algorithm requires the differences of the objective function values rather than directly calculating the gradients of the objective function. Owing to such a better search strategy, the NM algorithm needs much fewer function evaluations in most cases compared with the exhaustive search method.

2.2 Adopted optimization algorithms

2.2.1 Nelder-Mead algorithm

The NM algorithm is known as one of the best simplex methods for finding the local minimum of a function [22]. For two variables, this method performs a pattern search based on three vertices of a triangle [23]. At each stage, among three initial vertices, the worst vertex at which the objective function achieves the largest value is replaced by a new vertex which is generated by reflection, expansion, contraction, or shrink and leads to smallest objective function value compared to the previous vertices [22]. This process is iterated until converging to a local

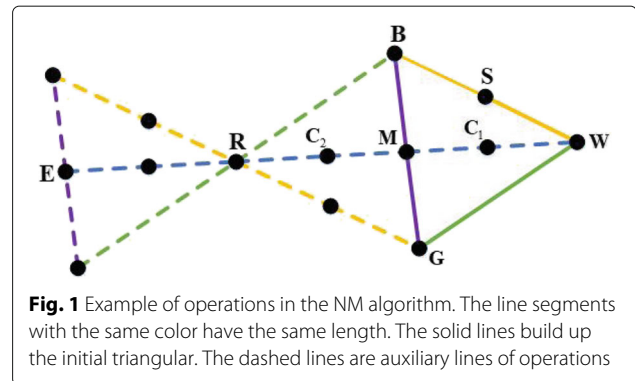


Fig. 1 Example of operations in the NM algorithm. The line segments with the same color have the same length. The solid lines build up the initial triangular. The dashed lines are auxiliary lines of operations

2.2.2 Unscented Kalman filter method

The UKF is a type of extended Kalman filters which has good performance for highly non-linear state transition and observation models [25]. Based on the deterministic

Algorithm 3 Searching strategy of the UKF method.

Input:

- $Y_2(\rho, \alpha)$: objective function shown in (7);
- $\{(\rho_k, \alpha_k)\}_{k=1, \dots, N_{\text{rand}}}$: set of initial points;
- N_{iter} : maximum iteration number;
- β : spread of sigma points;
- κ : secondary scaling parameter.

- 1: $L \leftarrow 2$, which is the dimension of the independent variable of $Y_2(\rho, \alpha)$;
- 2: $\lambda \leftarrow \beta^2 (L + \kappa) - L$;
- 3: $\mathbf{X} \leftarrow \begin{bmatrix} \alpha_1, \dots, \alpha_{N_{\text{rand}}} \\ \rho_1, \dots, \rho_{N_{\text{rand}}} \end{bmatrix}$;
- 4: $\bar{\mathbf{X}} \leftarrow \frac{1}{N_{\text{rand}}} \mathbf{X} \mathbf{1}_{N_{\text{rand}}}^T$
where $\mathbf{1}_{N_{\text{rand}}}$ is the row vector of N_{rand} ones;
- 5: $\mathbf{P}_{\mathbf{X}} \leftarrow \frac{1}{N_{\text{rand}}} \mathbf{X} \mathbf{H} \mathbf{X}^T$
where $\mathbf{H} = \mathbf{I} - \frac{1}{N_{\text{rand}}} \mathbf{1}_{N_{\text{rand}}}^T \mathbf{1}_{N_{\text{rand}}}$ and \mathbf{I} is the identity matrix of size N_{rand} ;

6: repeat

7: Create sigma points

$\mathcal{X} = \left[\bar{\mathbf{X}}, \bar{\mathbf{X}} \mathbf{1}_2 + \sqrt{(L + \lambda) \mathbf{P}'_{\mathbf{X}}}, \bar{\mathbf{X}} \mathbf{1}_2 - \sqrt{(L + \lambda) \mathbf{P}'_{\mathbf{X}}} \right]$,
as shown in Fig. 2, where $\mathbf{1}_2$ is the row vector of 2 ones,
and $\mathbf{P}'_{\mathbf{X}}$ contains the absolute values of elements in $\mathbf{P}_{\mathbf{X}}$;

8: Set weights of sigma points

$$\mathbf{W} = \left[\frac{\lambda}{L + \lambda}, \frac{1}{2(L + \lambda)}, \dots, \frac{1}{2(L + \lambda)} \right];$$

9: $\bar{\mathcal{X}} \leftarrow \mathcal{X} \mathbf{W}^T$;

10: $\mathbf{P}_{\mathcal{X}} \leftarrow \mathcal{X} \mathbf{H} \mathbf{W} \mathbf{W}_{\text{diag}} \mathbf{H}^T \mathcal{X}^T$

where $\mathbf{H}_{\mathbf{W}} = \mathbf{I} - \mathbf{W}^T \mathbf{1}_{N_{\text{rand}}}$ and \mathbf{W}_{diag} denotes the square matrix with the elements of \mathbf{W} on the main diagonal;

11: $\mathcal{Y} \leftarrow Y_2(\mathcal{X})$;

12: $\bar{\mathcal{Y}} \leftarrow \mathcal{Y} \mathbf{W}^T$;

13: $\mathbf{P}_{\mathcal{Y}} \leftarrow \mathcal{Y} \mathbf{H}_{\mathbf{W}} \mathbf{W}_{\text{diag}} \mathbf{H}_{\mathbf{W}}^T \mathcal{Y}^T$;

14: $\mathbf{P}_{\mathcal{X}\mathcal{Y}} \leftarrow \mathcal{X} \mathbf{H}_{\mathbf{W}} \mathbf{W}_{\text{diag}} \mathbf{H}_{\mathbf{W}}^T \mathcal{Y}^T$;

15: $\kappa \leftarrow \mathbf{P}_{\mathcal{X}\mathcal{Y}} \mathbf{P}_{\mathcal{Y}}^{-1}$;

16: $\bar{\mathbf{X}} \leftarrow \bar{\mathbf{X}} + \kappa (\mathbf{0} - \bar{\mathcal{Y}})$;

17: $\mathbf{P}_{\mathbf{X}} \leftarrow \mathbf{P}_{\mathcal{X}} - \kappa \mathbf{P}_{\mathcal{Y}} \kappa^T$;

18: until the convergence is finished or the iteration number has reached N_{iter} .

Output:

$\bar{\mathbf{X}}$: approximated minimum point.

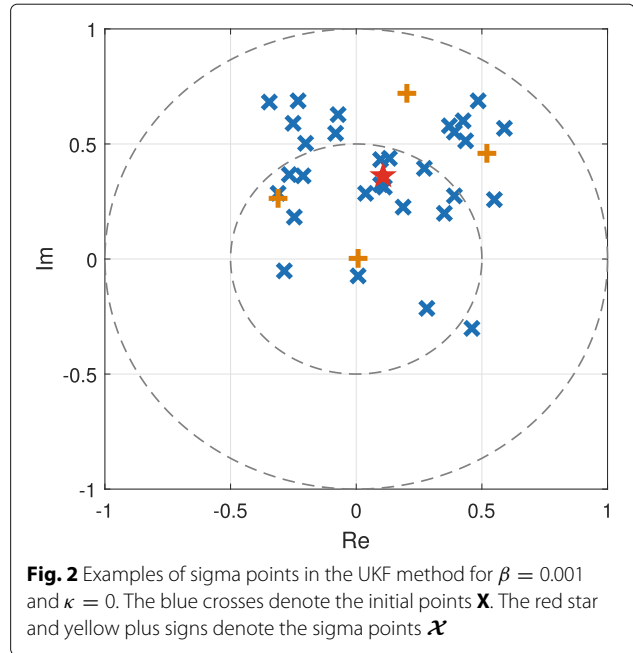


Fig. 2 Examples of sigma points in the UKF method for $\beta = 0.001$ and $\kappa = 0$. The blue crosses denote the initial points \mathbf{X} . The red star and yellow plus signs denote the sigma points \mathcal{X}

sampling technique called unscented transform, the UKF minimizes the absolute error between the estimated observation and the true measurement. In the optimization problem, by setting the true measurement as 0 and the estimated observation as the objective function, the UKF can be considered as a numerical optimization method to minimize the absolute value of the objective function. The searching strategy is shown in Algorithm 3 [26, 27]. In the following simulations, the parameters of the UKF method are set following the suggestions in [25], i.e., $\beta = 0.001$ and $\kappa = 0$.

Based on the unscented transform technique, the UKF does not require the gradient information and normally does not need too many function evaluations.

2.2.3 Evolutionary algorithms

Evolutionary algorithms belong to stochastic search methods [28]. They mimic the metaphor of the natural biological evolution and the social behavior of species [29]. In this paper, the GA and the PSO algorithm are studied.

The GA is inspired by the improved fitness of biological systems through the evolution, used in several research areas to find exact/approximate solutions to the optimization problems [30]. In the GA, a population of candidate solutions, called chromosomes, containing low objective function values are selected from a random population [31, 32]. These selected chromosomes change their elements, called genes, through crossover or mutation processes to produce offspring chromosomes [33]. Then, these offspring chromosomes are evaluated by the

Table 1 Suggested selection ranges of parameters in NM, UKF, GA, and PSO algorithms

Algorithm	N_{rand}	N_{iter}
NM	[600,1000]	[10,200]
UKF	[200,1000]	[1,8]
	N_{gen}	N_{ind}
GA	[5,200]	[10,50]
PSO	[10,40]	[10,40]

objective function and selected to evolve the population if they could provide better solutions than weak population members do [34]. In the crossover process, selected chromosomes containing better solutions exchange parts of their information to produce offspring chromosomes [35]. As opposed to the crossover process, the mutation process changes a piece of genes in one offspring chromosome randomly, which generates new genetic material to avoid the genetic algorithm converging to local minimum [32].

The PSO algorithm is a population-based search algorithm, inspired by the social behavior of a flock of

migrating birds trying to reach an unknown destination [32, 36]. The optimization procedure initializes with a random generation of points in the search space, usually called particles [37]. As opposed to the GA, the PSO algorithm does not create new generations. The particles in the population only evolve their movement speed and position to achieve the desired position based on their own experience and also the experience of others [38]. In every search step, the position of the best particle who achieves the minimum objective function value is determined as the best fitness of all particles. Based on this position and its own previous best position, each particle updates its velocity to catch up with the best particle [39].

As evolutionary algorithms in general are based on stochastic search, they do not require the gradient information as well as sifting initial points in the computation. In addition, comparing to the exhaustive search method, less number of function evaluations is needed, normally.

2.3 Evaluation indices

In the following simulation studies, three indices are considered to evaluate the performances of different optimization algorithms:

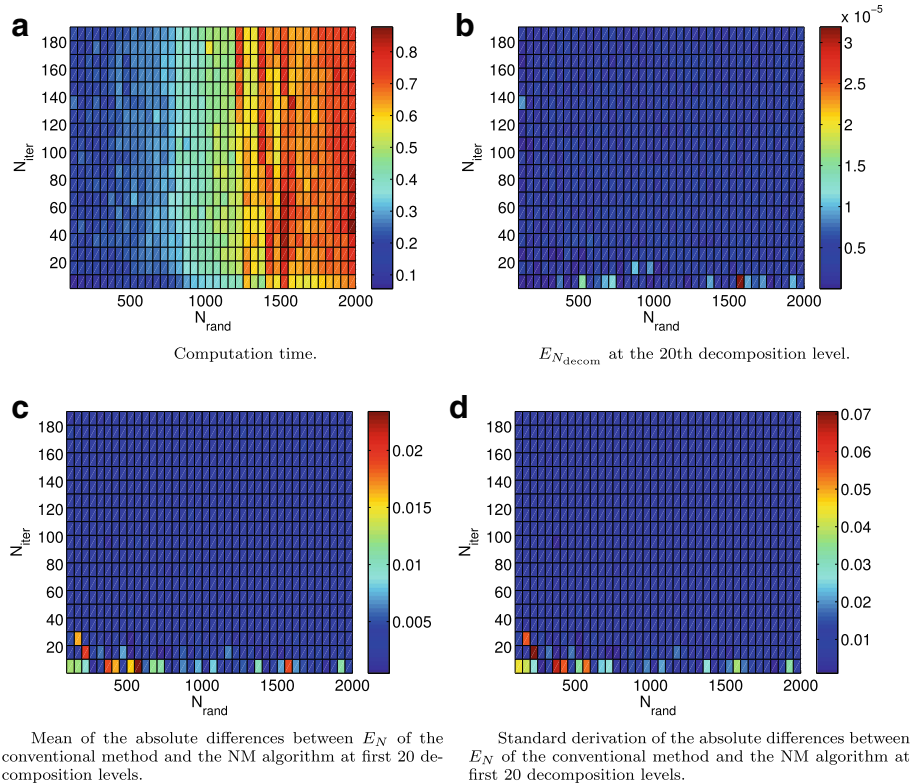


Fig. 3 Effects of N_{rand} and N_{iter} in the NM algorithm for the complex-valued signal $G(z)$. Four figures show the computation time and $E_{N_{decom}}$ at the 20th decomposition level as well as the mean and the standard deviation of the absolute differences between E_N of the conventional method and the NM algorithm at first 20 decomposition levels for different values of N_{rand} and N_{iter}

1. The reconstruction energy error at the maximum decomposition level N_{decom} , denoted as $E_{N_{\text{decom}}}$, is defined as

$$E_{N_{\text{decom}}} = \frac{\left| \|s_{\text{ori}}\|^2 - \|s_{N_{\text{decom}}}\|^2 \right|}{\|s_{\text{ori}}\|^2} \times 100\% \quad (9)$$

where s_{ori} and $s_{N_{\text{decom}}}$ denote the original signal and the reconstructed signal at the maximum decomposition level, respectively. $E_{N_{\text{decom}}}$ is assessed to verify whether the AFD based on the optimization algorithm converges or not;

2. The absolute difference between the reconstruction error in sense of energy at the N th decomposition level, E_N , of the conventional exhaustive search method and the other optimization method at each decomposition level where E_N is defined in (10), which is used to verify whether the energy convergence rate remains satisfactory by considering the search results of the conventional exhaustive search method as the standard results.

$$E_N = \frac{\left| \|s_{\text{ori}}\|^2 - \|s_N\|^2 \right|}{\|s_{\text{ori}}\|^2} \times 100\% \quad (10)$$

where s_N denote the reconstructed signal at the N th decomposition level.

3. The computation time that is applied to evaluate the computation efficiency of the AFD. The units of time for the following simulation results are the second.

All following simulations are conducted in MATLAB R2014a at a PC equipped with Intel(R) Core(TM) i7-4770 CPU @ 3.40 GHz and 12 GB RAM. Moreover, in the following simulations, all numerical integrations in algorithm 1 are implemented based on the 6 order Newton-Cotes formula. The lengths of the processed signals in the following simulations are all set as 2500 sample points.

3 Simulation results

3.1 Effects of optimization parameters

To reveal the effects of optimization algorithm parameters, a complex-valued signal given by (11), which is also studied in [1], is taken as an example. In this part, only effects of parameters for the complex-valued signal $G(z)$ is

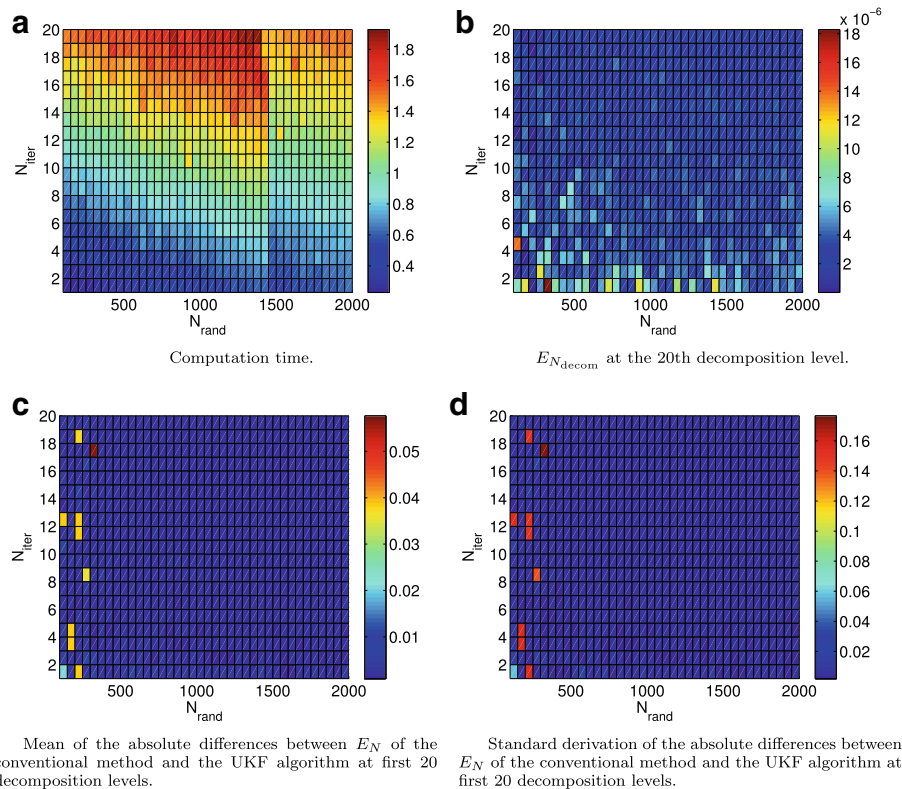


Fig. 4 Effects of N_{rand} and N_{iter} in the UKF method for the complex-valued signal $G(z)$. Four figures show the computation time and $E_{N_{\text{decom}}}$ at the 20th decomposition level as well as the mean and the standard deviation of the absolute differences between E_N of the conventional method and the UKF method at first 20 decomposition levels for different values of N_{rand} and N_{iter}

shown in detail. For real-valued signals in Section 3.2 and real ECG signals in Section 3.3, effects of parameters are similar to the case for $G(z)$.

$$G(z) = \frac{0.0247z^4 + 0.0355z^3}{(1 - 0.9048z)(1 - 0.3679z)} \quad (11)$$

The total number of points that need to be evaluated in the MS process affects the optimization accuracy and the workload very much. The more points evaluated, the better computation result but the longer computation time. Therefore, there exists a trade-off between the computation accuracy and the fast speed. Such control parameters for the NM algorithm and the UKF method are the number of points N_{rand} in the dictionary for searching initial points and the maximum iteration number N_{iter} , respectively. The control parameters for the GA and the PSO algorithm are the number of individuals N_{ind} and the maximum number of generations N_{gen} , respectively. For the UKF method, to get the best searching speed, L and λ are set as 2 and 0.001, respectively.

Effects of these parameters are determined based on simulation results. The maximum decomposition level N_{decom} is set to 20 for the complex-valued signal $G(z)$

since the first 20 decomposition components are enough to approximate $G(z)$ according to Ref. [1]. According to the following simulation results, the suggested ranges of the parameters for the optimization algorithms are shown in Table 1, which can lead to relative low computation time and high optimization accuracy at the same time.

For the NM algorithm, N_{rand} and N_{iter} are selected from [100, 2000] and [1, 200] for evaluations, respectively. Simulation results of the $G(z)$ signal are illustrated in Fig. 3. It can be seen that all values of $E_{N_{\text{decom}}}$ are small no matter which values of parameters are selected as shown in Fig. 3b, which means that the NM algorithm can keep the convergence of the AFD. In addition, the simulation result shown in Fig. 3a indicates that the effect of N_{iter} in the given range of the computation speed is not very large. However, the absolute differences of E_N between the conventional method and the NM method are large and unstable when N_{iter} is smaller than 10 and N_{rand} is smaller than 600. A major reason is that, when the evaluated points are not enough, the NM algorithm would not lead to the global optimum, which may deteriorate the convergence rate of the remainder energy. Although increasing N_{rand} and N_{iter} could increase the computation accuracy, the computation time would also be increased.

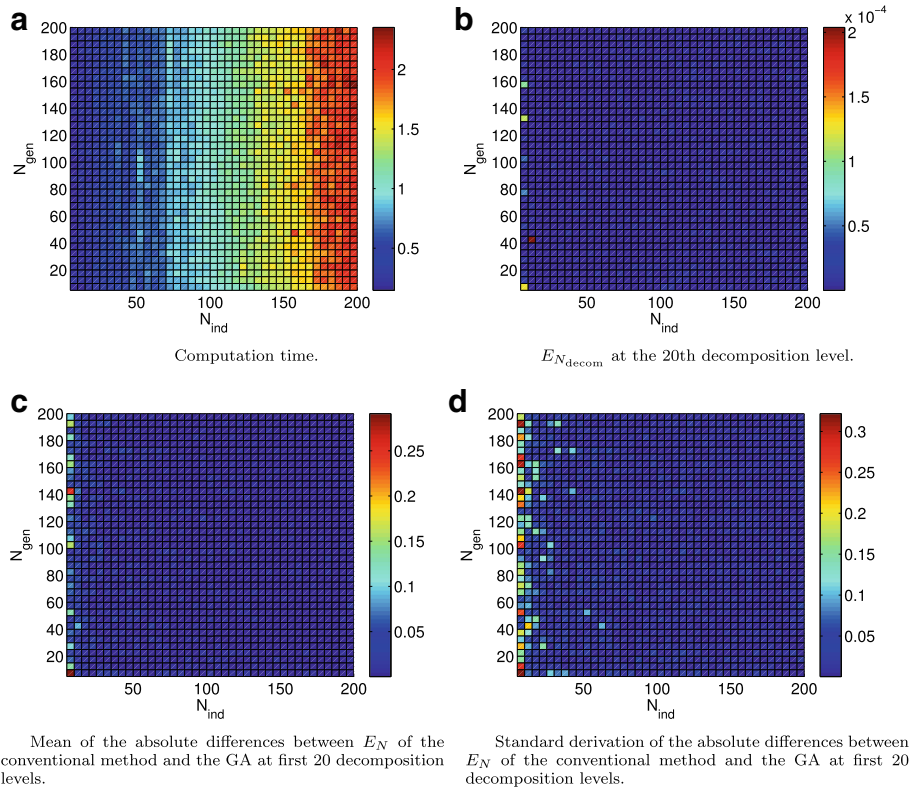


Fig. 5 Effects of N_{gen} and N_{ind} in GA for the complex-valued signal $G(z)$. Four figures show the computation time and $E_{N_{\text{decom}}}$ at the 20th decomposition level as well as the mean and the standard deviation of the absolute differences between E_N of the conventional method and the GA at first 20 decomposition levels for different values of N_{gen} and N_{ind}

In summary, for the NM algorithm, the suggested ranges of N_{rand} and N_{iter} are [600, 1000] and [10, 200], respectively.

For the UKF method, N_{rand} and N_{iter} are selected from [100, 2000] and [1, 20] respectively for evaluations. Simulation results are illustrated in Fig. 4. It can be seen that values of $E_{N_{decom}}$ are small. However, for some N_{iter} values when N_{rand} is smaller than 200, the absolute differences of E_N between the conventional method and the UKF method cannot keep small and stable, or consequently, the convergence rate of the remainder energy can not keep high. In addition, the computation time will be increased very much as N_{rand} and N_{iter} increase as shown in Fig. 4a. Moreover, based on simulation results, the effect of N_{iter} is larger than that of N_{rand} . In summary, the suggested ranges of N_{rand} and N_{iter} for the UKF method are [200, 1000] and [1, 8], respectively.

For the GA, N_{gen} , and N_{ind} are all selected in [5, 200] for evaluations. Simulation results are illustrated in Fig. 5. Although all values of $E_{N_{decom}}$ are small, differences between E_N of the conventional method and the GA become large and unstable when N_{ind} is smaller than 20. Therefore, to achieve the fast energy convergence rate, N_{ind} is required to be larger than 20. Moreover,

N_{ind} will affect the computation speed significantly. As N_{ind} increases, the computation time increases drastically. In summary, the ranges of N_{gen} and N_{ind} for the GA method are suggested as [5, 200] and [10, 50], respectively.

For the PSO algorithm, N_{ind} and N_{gen} are all selected in [10, 100] for evaluations. Simulation results are shown in Fig. 6. It can be seen that all values of $E_{N_{decom}}$ are small. Although the absolute differences of E_N between the conventional method and the PSO algorithm are relatively large and unstable when N_{ind} and N_{gen} are small, they are still acceptable in comparison with the simulation results of the NM algorithm, the UKF method and the GA. Moreover, since all individual states need to be updated one by one, computation time will be increased considerably as N_{ind} and N_{gen} increase. In summary, the suggested ranges of N_{gen} and N_{ind} for the PSO algorithm are all [10, 40].

3.2 Optimization performance comparison for typical signals

In this part, four typical real-valued signals defined in Ref. [18] are considered to compare decomposition performances of the AFD based on the NM algorithm, the

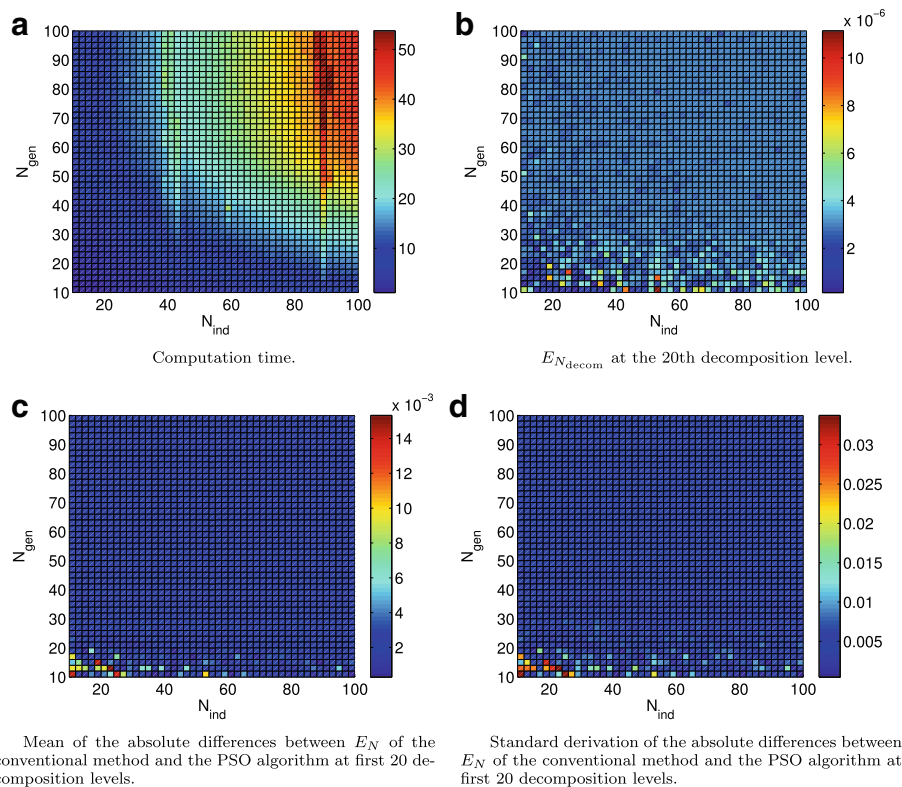


Fig. 6 Effects of N_{gen} and N_{ind} in PSO algorithm for the complex-valued signal $G(z)$. Four figures show the computation time and $E_{N_{decom}}$ at the 20th decomposition level as well as the mean and the standard deviation of the absolute differences between E_N of the conventional method and the PSO algorithm at first 20 decomposition levels for different values of N_{gen} and N_{ind}

UKF method, the GA and the PSO algorithm. According to the suggested selection ranges shown in Table 1, the selected parameters for the optimization algorithms are shown in Table 2. The maximum decomposition levels N_{decom} of the AFD for these four signals are set as 100 to make sure that the AFD can extract most energy from the original signal.

Table 3 lists the computation times in all situations where STD denotes the standard deviation of the computation time for these different signals. The AFD based on the UKF uses the least computational time for all four types of signals. In addition, the computation time of the AFD based on the PSO method is close to or higher than that of the AFD based on the conventional exhaustive search.

The corresponding reconstructed energy error is shown in Table 4. It can be seen that, compared to the relative energy error in [1, 40] for evaluating whether the AFD is converged, all reconstructed signals can approximate the original signals respectively with very small reconstruction error. Therefore, these optimization algorithms do not affect the convergence of the AFD. Comparatively, the reconstructions for the block signal are worse than these for other signals since the block signal, as a combination of several different square waves, contains many high-frequency components and thus requires more decomposition levels to obtain a more accurate reconstruction.

To verify the convergence rate, the remainder energy errors at the first 100 decomposition levels are illustrated in Fig. 7. The UKF and the NM methods have energy decay rates almost the same as the conventional method but larger than the GA and the PSO methods, especially for the bump signal. The reason is that the GA and the PSO methods fail to reach the global minimum in some decomposition levels due to small number of initial points and iteration loops. However, increasing the number of generations and individuals would increase the computation time as shown in Figs. 5a and 6a. Therefore, for these four typical real-valued signals, the GA and the PSO algorithm cannot make the AFD achieve a fast computation speed with a high energy convergence rate at the same time.

Table 2 Selected parameters of NM, UKF, GA, and PSO algorithms for simulations

Algorithm	N_{rand}	N_{iter}
NM	700	10
UKF	700	8
	N_{gen}	N_{ind}
GA	100	20
PSO	10	10

Table 3 Computation time for five optimization algorithms in four typical real-valued signals

	Heavisine signal	Doppler signal	Bump signal	Block signal	Mean	STD
Conventional method	5.496	5.451	5.647	5.501	5.524	0.085
NM method	1.286	1.172	1.316	1.331	1.276	0.072
UKF method	<i>1.135</i>	<i>1.052</i>	<i>1.208</i>	<i>1.161</i>	<i>1.139</i>	<i>0.065</i>
GA method	1.607	1.447	1.934	2.424	1.853	0.431
PSO method	5.094	8.047	5.031	4.953	5.781	1.512

Fonts in italics denote that there are all significant differences between the UKF method and any one of other methods by paired *t*-tests ($p < 0.01$)

3.3 Optimization performance comparison for real ECG signals

Results in Section 3.3 show that the proposed AFD implementation methods can provide the better performance for four representative signals compared with the conventional exhaustive search. To verify the performance of these proposed methods for real signals, simulations of real ECG signals are carried out. Table 5 illustrates the comparisons of computation time between optimization methods. For all records, the computations based on the proposed methods are faster than that based on the conventional exhaustive search. In these proposed methods, the UKF-based AFD method uses least computation time for most records. The PSO-based AFD method performs worst compared to other proposed methods.

4 Discussions

According to results in Section 3, the UKF method based AFD can provide the best performance for all four representative and real ECG signals. It is reasonable that the UKF can achieve the good performance. The UKF method can produce the optimization results within a small number of iterations, and therefore, not too many points need to be evaluated. Figure 8 illustrates the comparisons of the convergence of proposed optimization methods. The UKF method has the highest convergence rate and most accurate searching result at the beginning iteration level,

Table 4 Reconstruction energy error of different optimization algorithms for different signals

	Heavisine signal	Doppler signal	Bump signal	Block signal
Conventional method	1.1×10^{-6}	7.3×10^{-7}	0.1×10^{-6}	0.6×10^{-3}
NM method	1.4×10^{-6}	3.8×10^{-7}	0.1×10^{-6}	2.1×10^{-3}
UKF method	8.0×10^{-6}	10.0×10^{-7}	1.2×10^{-6}	4.4×10^{-3}
GA method	9.5×10^{-6}	6.8×10^{-7}	2.4×10^{-6}	1.7×10^{-3}
PSO method	1.5×10^{-6}	1.3×10^{-7}	1.6×10^{-6}	1.1×10^{-3}

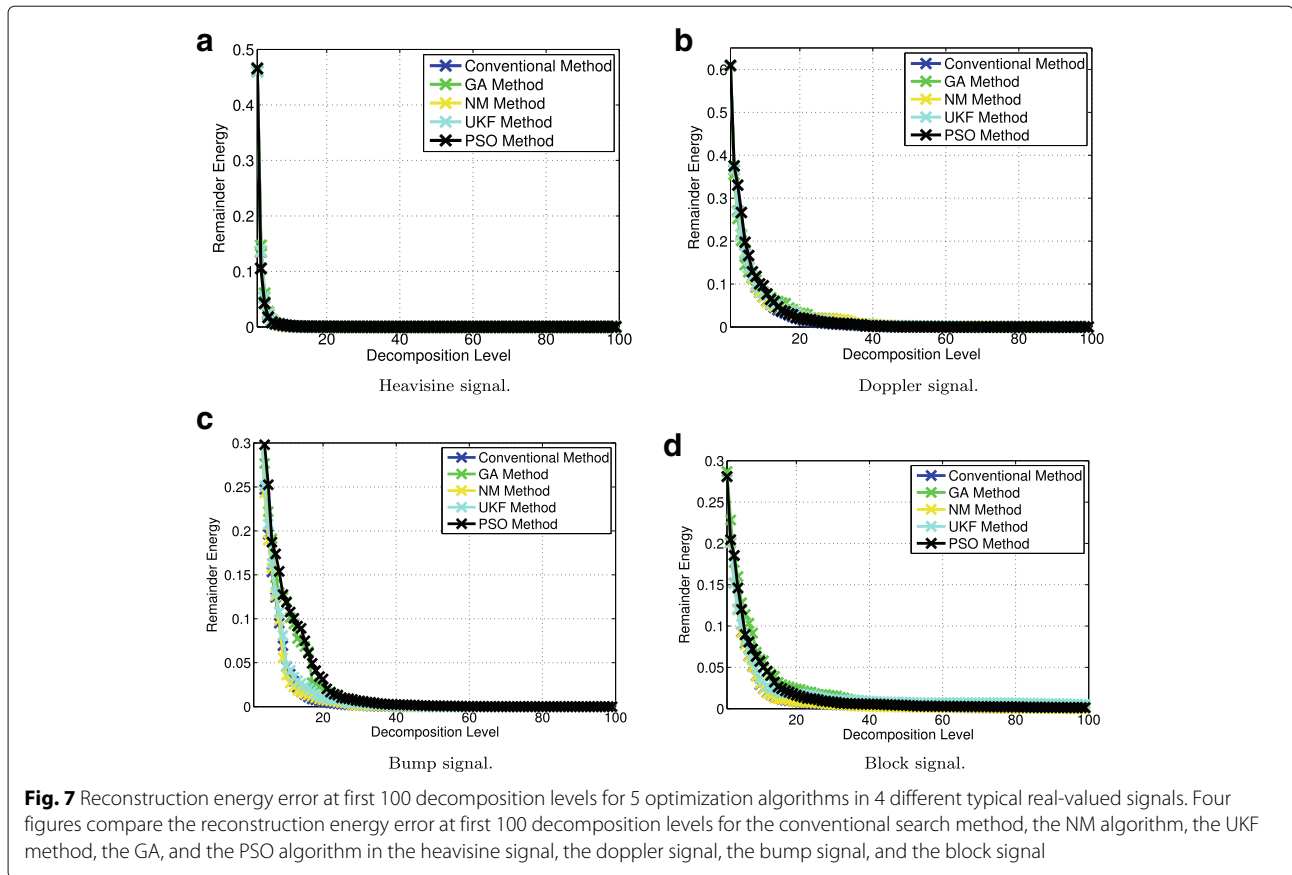


Fig. 7 Reconstruction energy error at first 100 decomposition levels for 5 optimization algorithms in 4 different typical real-valued signals. Four figures compare the reconstruction energy error at first 100 decomposition levels for the conventional search method, the NM algorithm, the UKF method, the GA, and the PSO algorithm in the heavisine signal, the doppler signal, the bump signal, and the block signal

Table 5 Computation time of optimization algorithms for real ECG signals

Real ECG records	Conventional method	NM method	UKF method	GA method	PSO method
100	2.895	0.722	<i>0.539</i>	1.156	2.211
101	2.982	0.618	<i>0.526</i>	1.019	2.015
102	2.871	0.594	<i>0.541</i>	1.022	2.071
103	2.798	0.554	<i>0.519</i>	1.002	2.010
104	2.793	<i>0.547</i>	<i>0.578</i>	0.928	1.994
105	2.824	0.546	<i>0.545</i>	0.966	1.986
106	3.172	<i>0.543</i>	0.989	1.348	2.148
107	3.293	0.595	<i>0.495</i>	1.122	2.046
108	3.294	0.577	<i>0.533</i>	1.013	2.059
109	2.863	0.563	<i>0.543</i>	0.959	2.067
119	2.775	<i>0.518</i>	0.546	0.945	1.994
201	3.029	<i>0.557</i>	0.565	0.996	2.520
202	3.059	0.585	<i>0.563</i>	0.981	2.049
203	2.942	0.831	<i>0.654</i>	0.999	2.117
205	3.011	0.903	<i>0.714</i>	1.113	2.366
207	2.907	0.607	<i>0.528</i>	1.025	2.117
208	2.958	<i>0.560</i>	0.606	1.033	2.067
209	2.888	0.573	<i>0.533</i>	1.271	2.093
213	2.847	0.620	<i>0.588</i>	0.961	2.077
Mean	2.958	0.611	<i>0.584</i>	1.045	2.106
STD	0.155	0.101	0.110	0.111	0.134

Fonts in italics denote that there are all significant differences between the UKF method and any one of other methods by paired t-tests ($p < 0.001$)

which means that the UKF method can achieve the pre-set threshold of the optimization error within a small iteration number. Therefore, the UKF does not need to evaluate large number of function values in the optimization process. Figure 9 shows the total number of evaluated objective function values in the AFD based on proposed optimization methods. It can be seen that the UKF requires the smallest number of points to search the suitable a_n sequence. Such small number of iterations and small number of objective function evaluations will lead the UKF-based AFD achieve the short computation time.

In the UKF, besides the parameters N_{rand} and N_{iter} mentioned in the Section 3.1, there are other two parameters, i.e., the spread of sigma points β and the secondary scaling parameter κ , as shown in Algorithm 3. These two parameters determine the scaling parameter λ defined as [25]

$$\lambda = \beta^2(L + \kappa) - L. \tag{12}$$

In this paper, simulations are carried out with $\beta = 0.001$ and $\kappa = 0$ which is suggested in [25]. These two parameters will also affect the optimization results of the UKF method. Figure 10a and b illustrate the effects of β for the convergence of the UKF method where \bar{X}_i denotes the estimated observation in the i th iteration that can be considered as the updated optimization result obtained from

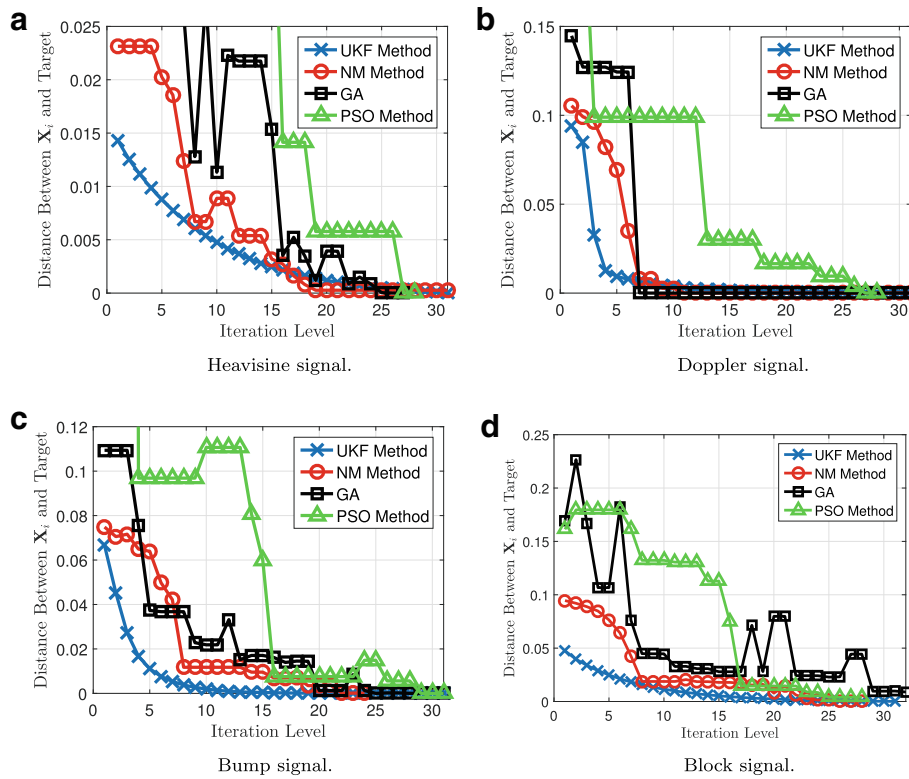


Fig. 8 Differences of the searching results \mathbf{X}_i in the i th iteration of four proposed optimization methods and final optimization results of the conventional exhaustive search method. Four figures compare the convergences of the optimization results for the NM algorithm, the UKF method, the GA, and the PSO algorithm in the heavisine signal, the doppler signal, the bump signal, and the block signal

the i th iteration, as well as \mathbf{P}_X^i and $\mathbf{P}_X^{\text{final}}$ denote the covariances of sample points \mathbf{X} in the i th and the final iterations that can be considered as the updated descent steps obtained from the i th iteration and the final iteration. It can be seen that the error between $\bar{\mathbf{X}}_i$ and the optimum is small at the beginning iteration level when β is small. In addition, $\bar{\mathbf{X}}$ and \mathbf{P}_X achieve the small values faster when β is smaller. Except the parameter β , Fig. 10c and d show the effects of κ for the convergence of $\bar{\mathbf{X}}$ and \mathbf{P}_X . It can be seen that, when κ is close to 0, $\bar{\mathbf{X}}$ and \mathbf{P}_X can converge to the small values faster though the differences between

\mathbf{P}_X^i and $\mathbf{P}_X^{\text{final}}$ are not smallest at the beginning iteration level when $\kappa = 0$. Based on these simulation results, the suggested selections of κ and β in [25] are also suitable for the AFD.

5 Conclusion

In order to improve the computation efficiency of the AFD, four typical optimization algorithms, including the UKF method, the NM algorithm, the GA, and the PSO algorithm, are adopted in basis search and compared to the conventional exhaustive search method. The

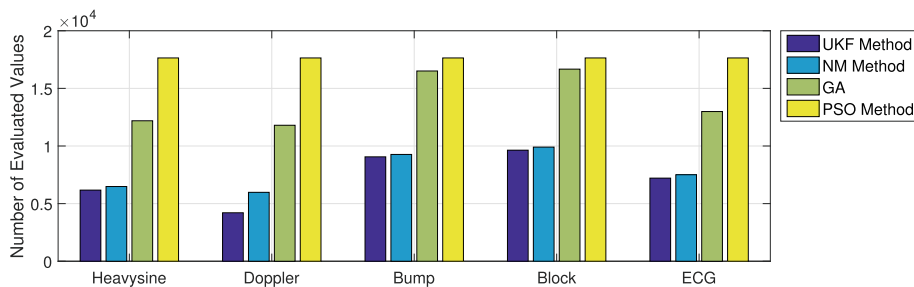


Fig. 9 Total number of evaluated objective function values in four optimization methods-based AFD. This figure compares the number of evaluated objective function values for the NM algorithm, the UKF method, the GA, and the PSO algorithm in the heavisine signal, the doppler signal, the bump signal, and the block signal

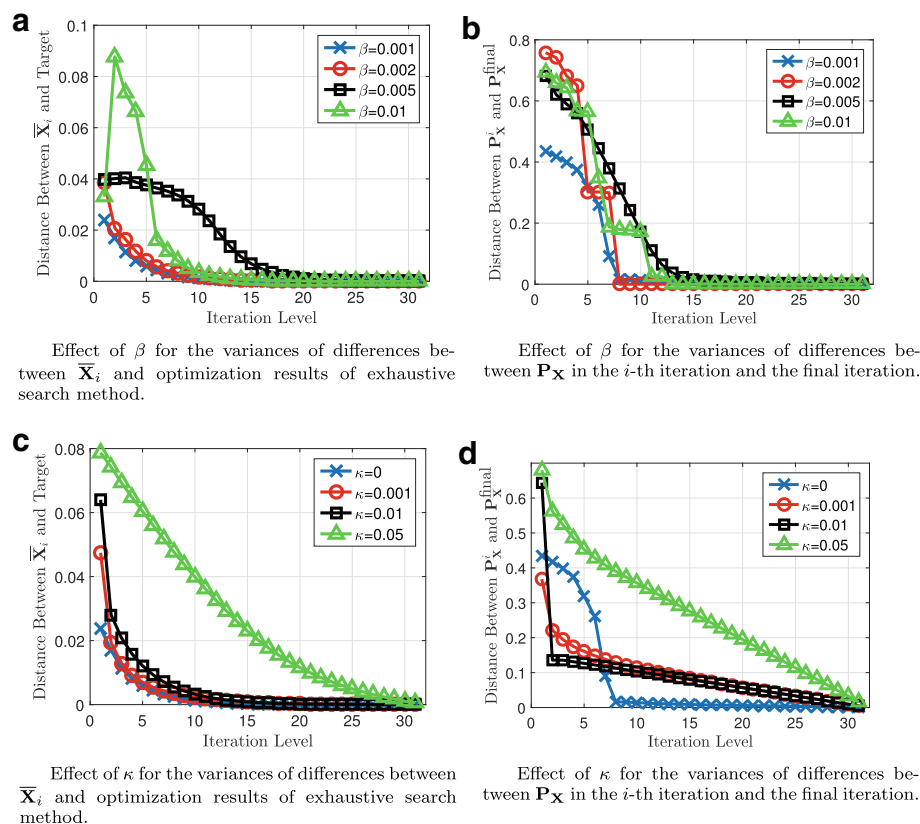


Fig. 10 Effects of parameters for the convergence of estimated observation $\bar{\mathbf{X}}$ and covariance \mathbf{P}_X in the UKF. Four figures compare the effects of parameters β and κ for the estimated observation $\bar{\mathbf{x}}_i$ and covariance \mathbf{P}_X in the UKF method

maximization problem with one complex-valued variable in the basis search of the AFD is reformulated as the equivalent minimization problem with two real-valued variables. Simulations are conducted to four typical signals, including the heavisine signal, the doppler signal, the bump signal, and the block signal, which can represent spatially variable functions appearing in the signal processing. To verify the performance for real signals, simulations are also carried out for real ECG signals. Comparative results show that the UKF method can achieve the highest computation speed with a fast energy convergence rate for these signals.

Abbreviations

AFD: Adaptive fourier decomposition; GA: Genetic algorithm; MS: Maximal sifting; NM: Nelder-Mead; MSP: Maximal selection principle; PSO: Particle swarm optimization; STD: Standard deviation; UKF: Unscented Kalman filter

Availability of data and materials

Please contact author for data requests.

Funding

This work is supported in part by the Macau Science and Technology Development Fund (FDCT) under projects 036/2009/A, 142/2014/SB, 055/2015/A2 and 079/2016/A2, the University of Macau Research Committee under MYRG projects 069(Y1-L2)-FST13-WF, 2014-00174-FST, 2016-00240-FST, 2016-00053-FST and 2017-00207-FST.

Authors' contributions

All the authors have participated in writing the manuscript. All authors read and approved the manuscript.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

¹Department of Electrical and Computer Engineering, Faculty of Science and Technology, University of Macau, Avenida da Universidade, Taipa, Macau, China. ²Department of Mathematics, Faculty of Science and Technology, University of Macau, Avenida da Universidade, Taipa, Macau, China.

Received: 26 June 2018 Accepted: 2 November 2018

Published online: 06 December 2018

References

1. T. Qian, L. Zhang, Z. Li, Algorithm of adaptive Fourier decomposition. *IEEE Trans. Signal Process.* **59**(12), 5899–5906 (2011). <https://doi.org/10.1109/TSP.2011.2168520>

2. T. Qian, Adaptive Fourier decompositions and rational approximations, part I: Theory. *Int. J. Wavelets Multiresolut. Inf. Process.* **12**(5), 1461008 (2014). <https://doi.org/10.1142/S0219691314610086>
3. L. Zhang, W. Hong, W. Mai, T. Qian, Adaptive Fourier decomposition and rational approximation – part II: Software system design and development. *Int. J. Wavelets Multiresolut. Inf. Process.* **12**(05), 1461009 (2014). <https://doi.org/10.1142/S0219691314610098>
4. W. Mi, T. Qian, F. Wan, A fast adaptive model reduction method based on Takenaka–Malmquist systems. *Syst. Control Lett.* **61**(1), 223–230 (2012). <https://doi.org/10.1016/j.sysconle.2011.10.016>
5. Z. Wang, F. Wan, C. M. Wong, L. Zhang, Adaptive Fourier decomposition based ECG denoising. *Comput. Biol. Med.* **77**, 195–205 (2016). <https://doi.org/10.1016/j.combiomed.2016.08.013>
6. Z. Wang, L. Yang, C. M. Wong, F. Wan, in *12th Int. Symp. Neural Networks*. Fast basis searching method of adaptive Fourier decomposition based on Nelder–Mead algorithm for ECG signals (Springer, Jeju, South Korea, 2015), pp. 305–314. https://doi.org/10.1007/978-3-319-25393-0_34
7. J. Ma, T. Zhang, M. Dong, A novel ECG data compression method using adaptive Fourier decomposition with security guarantee in e-health applications. *IEEE J. Biomed. Heal. Informatics.* **19**(3), 986–994 (2015). <https://doi.org/10.1109/JBHI.2014.2357841>
8. Q. Chen, T. Qian, Y. Li, W. Mai, X. Zhang, Adaptive Fourier tester for statistical estimation. *Math. Method. Appl. Sci.* **39**(12), 3478–3495 (2016). <https://doi.org/10.1002/mma.3795>
9. C. Tan, L. Zhang, H. T. Wu, A novel Blaschke unwinding adaptive Fourier decomposition based signal compression algorithm with application on ECG Signals. *IEEE J. Biomed. Heal. Inform.*, 1–11 (2018). <https://doi.org/10.1109/JBHI.2018.2817192>
10. T. Qian, Y.-B. Wang, Adaptive Fourier series—a variation of greedy algorithm. *Adv. Comput. Math.* **34**(3), 279–293 (2011). <https://doi.org/10.1007/s10444-010-9153-4>
11. T. Qian, Y. Wang, Remarks on adaptive Fourier decomposition. *Int. J. Wavelets Multiresolut. Inf. Process.* **11**(1), 1350007 (2013). <https://doi.org/10.1142/S0219691313500070>
12. T. Qian, Intrinsic mono-component decomposition of functions: an advance of Fourier theory. *Math. Methods Appl. Sci.* **33**(7), 880–891 (2010). <https://doi.org/10.1002/mma.1214>
13. T. Qian, Cyclic AFD algorithm for the best rational approximation. *Math. Methods Appl. Sci.* **37**(6), 846–859 (2014). <https://doi.org/10.1002/mma.2843>
14. R. R. Coifman, S. Steinerberger, H.-t. Wu, Carrier frequencies, holomorphy, and unwinding. *SIAM J. Math. Anal.* **49**(6), 4838–4864 (2017). <https://doi.org/10.1137/16M1081087>
15. R. R. Coifman, S. Steinerberger, Nonlinear phase unwinding of functions. *J. Fourier Anal. Appl.* **23**(4), 778–809 (2017). <https://doi.org/10.1007/s00041-016-9489-3>
16. G. Plonka, V. Pototskaia, Computation of adaptive Fourier series by sparse approximation of exponential sums. *J. Fourier Anal. Appl.*, 1–29 (2018). <https://doi.org/10.1007/s00041-018-9635-1>
17. A. Kirkbas, A. Kizilkaya, E. Bogar, in *2017 40th International Conference on Telecommunications and Signal Processing (TSP)*. Optimal basis pursuit based on jaya optimization for adaptive fourier decomposition (IEEE, Barcelona, Spain, 2017), pp. 538–543. <https://doi.org/10.1109/TSP.2017.8076045>
18. D. L. Donoho, I. M. Johnstone, Ideal spatial adaptation by wavelet shrinkage. *Biometrika.* **81**(3), 425–455 (1994). <https://doi.org/10.1093/biomet/81.3.425>
19. A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, H. E. Stanley, PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals. *Circulation.* **101**(23), 215–220 (2000). <https://doi.org/10.1161/01.CIR.101.23.e215>
20. G. B. Moody, R. G. Mark, The impact of the MIT-BIH arrhythmia database. *IEEE Eng. Med. Biol. Mag.* **20**(3), 45–50 (2001). <https://doi.org/10.1109/51.932724>
21. S. Y. Park, A. K. Bera, Maximum entropy autoregressive conditional heteroskedasticity model. *J. Econom.* **150**(2), 219–230 (2009). <https://doi.org/10.1016/j.jeconom.2008.12.014>
22. K. Klein, J. Neira, Nelder–Mead simplex optimization routine for large-scale problems: A distributed memory implementation. *Comput. Econ.* **43**(4), 447–461 (2013). <https://doi.org/10.1007/s10614-013-9377-8>
23. J. Nocedal, S. J. Wright, *Numer. Optim.* (Springer, New York, USA, 1999)
24. J. A. Nelder, R. Mead, A simplex method for function minimization. *Comput. J.* **7**(4), 308–313 (1965). <https://doi.org/10.1093/comjnl/7.4.308>
25. E. A. Wan, R. Van Der Merwe, in *Adapt. Syst. Signal Process. Commun. Control Symp. 2000. AS-SPCC. IEEE 2000*. The unscented Kalman filter for nonlinear estimation (IEEE, Albert, Canada, 2002), pp. 153–158. <https://doi.org/10.1109/ASSPCC.2000.882463>
26. S. J. Julier, J. K. Uhlmann, in *SPIE 3068, Signal Process. Sens. Fusion, Target Recognit. VI*. New extension of the Kalman filter to nonlinear systems (SPIE, Orlando, FL, USA, 1997), pp. 182–193. <https://doi.org/10.1117/12.280797>
27. S. Lienhard, J. G. Malcolm, C.-F. Westin, Y. Rathi, A full bi-tensor neural tractography algorithm using the unscented kalman filter. *EURASIP J. Adv. Signal Process.* **2011**(1), 77 (2011). <https://doi.org/10.1186/1687-6180-2011-77>
28. M. S. White, S. J. Flockton, A comparison of evolutionary algorithms for tracking time-varying recursive systems. *EURASIP J. Adv. Signal Process.* **2003**(8), 396340 (2003). <https://doi.org/10.1155/S1110865703303117>
29. R. Salvador, F. Moreno, T. Riesgo, L. Sekanina, Evolutionary approach to improve wavelet transforms for image compression in embedded systems. *EURASIP J. Adv. Signal Process.* **2011**(1), 973806 (2011). <https://doi.org/10.1155/2011/973806>
30. J. Riionheimo, V. Välimäki, Parameter estimation of a plucked string synthesis model using a genetic algorithm with perceptual fitness calculation. *EURASIP J. Adv. Signal Process.* **2003**(8), 758284 (2003). <https://doi.org/10.1155/S1110865703302100>
31. G. Pignatelli, R. Cucchiara, L. Cinque, S. Levialdi, Tuning range image segmentation by genetic algorithm. *EURASIP J. Adv. Signal Process.* **2003**(8), 683043 (2003). <https://doi.org/10.1155/S1110865703303087>
32. E. Elbeltagi, T. Hegazy, D. Grierson, Comparison among five evolutionary-based optimization algorithms. *Adv. Eng. Informatics.* **19**(1), 43–53 (2005). <https://doi.org/10.1016/j.aei.2005.01.004>
33. L. M. Schmitt, Theory of genetic algorithms. *Theor. Comput. Sci.* **259**(1–2), 1–61 (2001). [https://doi.org/10.1016/S0304-3975\(00\)00406-0](https://doi.org/10.1016/S0304-3975(00)00406-0)
34. S. Panda, N. P. Padhy, Comparison of particle swarm optimization and genetic algorithm for FACTS-based controller design. *Appl. Soft Comput.* **8**(4), 1418–1427 (2008). <https://doi.org/10.1016/j.asoc.2007.10.009>
35. L. M. Schmitt, Theory of genetic algorithms II: Models for genetic operators over the string-tensor representation of populations and convergence to global optima for arbitrary fitness function under scaling. *Theor. Comput. Sci.* **310**(1–3), 181–231 (2004). [https://doi.org/10.1016/S0304-3975\(03\)00393-1](https://doi.org/10.1016/S0304-3975(03)00393-1)
36. B. Li, Z. Zhou, W. Zou, W. Gao, Particle swarm optimization based noncoherent detector for ultra-wideband radio in intensive multipath environments. *EURASIP J. Adv. Signal Process.* **2011**(1), 341836 (2011). <https://doi.org/10.1155/2011/341836>
37. R. He, K. Wang, Q. Li, Y. Yuan, N. Zhao, Y. Liu, H. Zhang, A novel method for the detection of R-peaks in ECG based on K-nearest neighbors and particle swarm optimization. *EURASIP J. Adv. Signal Process.* **2017**(1), 82 (2017). <https://doi.org/10.1186/s13634-017-0519-3>
38. M. Clerc, J. Kennedy, The particle swarm – explosion, stability, and convergence in a multidimensional complex space. *IEEE Trans. Evol. Comput.* **6**(1), 58–73 (2002). <https://doi.org/10.1109/4235.985692>
39. IC. Trelea, The particle swarm optimization algorithm: convergence analysis and parameter selection. *Inf. Process. Lett.* **85**(6), 317–325 (2003). [https://doi.org/10.1016/S0020-0190\(02\)00447-7](https://doi.org/10.1016/S0020-0190(02)00447-7)
40. T. Qian, H. Li, M. Stessin, Comparison of adaptive mono-component decompositions. *Nonlinear Anal. Real World Appl.* **14**(2), 1055–1074 (2013). <https://doi.org/10.1016/j.nonrwa.2012.08.017>