Turk J Elec Eng & Comp Sci, Vol.19, No.1, 2011, © TÜBİTAK
doi:10.3906/elk-0912-344

A novel and efficient algorithm for adaptive filtering: Artificial bee colony algorithm

Nurhan KARABOĞA, Mehmet Bahadır ÇETİNKAYA
Department of Electronics Engineering, Faculty of Engineering, Erciyes University, 38039, Melikgazi, Kayseri, TURKEY
e-mail: nurhan_k@erciyes.edu.tr, cetinkaya@erciyes.edu.tr

Abstract

The uni-modal error surfaces and intrinsic stable behaviors of adaptive finite impulse response (FIR) filters make gradient based algorithms very effective in the design of these filters. Gradient based design methods are well developed for the design of adaptive FIR filters and widely applied to the distinct areas such as noise cancellation, system identification and channel equalization. However, the studies on adaptive infinite impulse response (IIR) filters are not as common as adaptive FIR filters since the stability during the adaptation process may not be ensured in some applications, and the convergence to the optimal design is not always guaranteed due to their multi-modal error surface structures. Gradient based design approaches may often get stuck at a local minimum in a multi-modal error surface and the stability of the designed filter can not be ensured. However, global optimization algorithms based approaches are able to converge to the global minimum in a multi-modal error surface and ensure the stability of the adaptive IIR filter. One of the most recently proposed swarm intelligence based global optimization algorithms is the artificial bee colony algorithm, which simulates the intelligent foraging behavior of honeybee swarms. In this work, a novel approach based on artificial bee colony algorithm is introduced for the design of adaptive FIR and adaptive IIR filters. Simulations are realized for the noise cancellation problem and the performance of the proposed approach is compared to that of some known gradient and evolutionary based approaches.

Key Words: *Artificial bee colony, Particle swarm optimization, Differential evolution, Adaptive filter design, Noise cancellation*

1. Introduction

Digital signal processing systems has become an active area of research in recent decades due to their low cost, reliability, accuracy, small physical sizes and flexibility [1]. One of the most common digital signal processing systems is the digital filters. Non-adaptive filters cannot process time-varying nonstationary signals and they require *a* priori knowledge of the statistics of the signal to be processed. Contrary to non-adaptive structures, if the input data varies with respect to time or there is no prescribed specification for the variation, adaptive filters are needed [1, 2].

Adaptive filters are associated to one of two major groups by their impulse response: finite impulse response (FIR) and infinite impulse response (IIR) filters. Due to their uni-modal error surfaces and intrinsic stable behavior, gradient based algorithms, such as least-mean-square (LMS) and normalized least-mean square (NLMS), are very effective in the design of adaptive FIR filters [3, 4].

However, since the gradient based algorithms try to find the global minimum of the error surface by moving in the direction of the negative gradient, approaches based on these algorithms may lead the filter to a local minimum when the error surface is multi-modal as such in IIR filters [4]. IIR filters can provide a much better performance than the FIR filters having the same number of coefficients [4, 5].

Despite this important advantage of IIR filters over FIR filters, there are two main problems encountered in the design of IIR filters: they might have a multi-modal error surface and the filter might become unstable during the adaptation process [6, 7]. The stability of the filter can be handled by limiting the parameter space in a suitable value range. The multi-modal error surface of IIR filter causes the gradient based algorithms to be stuck at local minima and not converge to the global optimum.

In order to overcome this problem and find the global optimum solution, approaches based on global optimization algorithms such as genetic algorithm (GA), simulated annealing (SA), tabu search (TS), differential evolution (DE), particle swarm optimization (PSO) and artificial bee colony (ABC) algorithms can be used for designing adaptive IIR filters. Among these algorithms, DE, PSO, TS, SA and GA based design methods have been described and applied to the adaptive filter design [3, 5, 8–11].

ABC offers an efficient swarm intelligence based algorithm. However, to the best of our knowledge, ABC algorithm has not been used to design adaptive filters although it has been employed in the design of nonadaptive filters [12].

In this work, a new approach based on ABC algorithm is presented for both FIR and IIR adaptive filter structures and an improved version of ABC algorithm is applied for adaptive noise cancellation. Also, a new approach for finding the optimal step size value of LMS-type algorithms is proposed in the design of adaptive FIR filters.

The remainder of this paper is organized as follows. Section 2 presents a brief review to the artificial bee colony algorithm and its improved version. In section 3, the adaptive noise cancellation problem is introduced, then an improved approach for finding the optimal step size of LMS-type algorithms is presented. Finally, the proposed approach based on artificial bee colony algorithm for solving adaptive noise cancellation problem is described and simulation results are discussed.

2. Artificial bee colony algorithm

2.1. Basic ABC

Swarm intelligence has become in recent years a research interest for many scientists in various areas. The swarm intelligence can be defined as any attempt for designing algorithms or distributed problem-solving devices inspired by the collective behavior of insects and other animal societies [13]. The classical examples of swarm are bee colony swarming around their hive; a colony of ants; a flock of birds; an immune system consisting of a swarm of cells; and a crowd as a swarm of people. In 2005, D. Karaboga introduced a bee swarm algorithm called artificial bee colony algorithm for numerical optimization problems [14]; and B. Basturk and D. Karaboga compared the performance of ABC with that of some other well-known population based optimization algorithms [15]. ABC have been employed by several researchers to solve various problems in different research areas [16–20].

Flowchart of the artificial bee colony algorithm is given in Figure 1 [12]. Each cycle of the search consists of three steps after initialization stage: placing the employed bees onto the food sources and calculating their nectar amounts; placing the onlookers onto the food sources and calculating the nectar amounts; and determining the scout bees and placing them onto the randomly determined food sources. In the ABC, a food source position represents a possible solution to the problem to be optimized. At initialization, a set of food source positions are randomly produced and values of the algorithm control parameters are assigned. The nectar amount retrievable from food source corresponds to the quality of the solution represented by that food source. So the nectar amounts of the food sources existing at the initial positions are determined. In other words, the quality (values) of the initial solutions are calculated. Each employed bee is moved onto her food source area to determine a new food source within the neighborhood of the present location, and then its nectar amount is evaluated. If the new nectar amount is higher, then she forgets the previous amount and remembers the new one. After the employed bees complete their search, they come back into the hive and share their information about the nectar

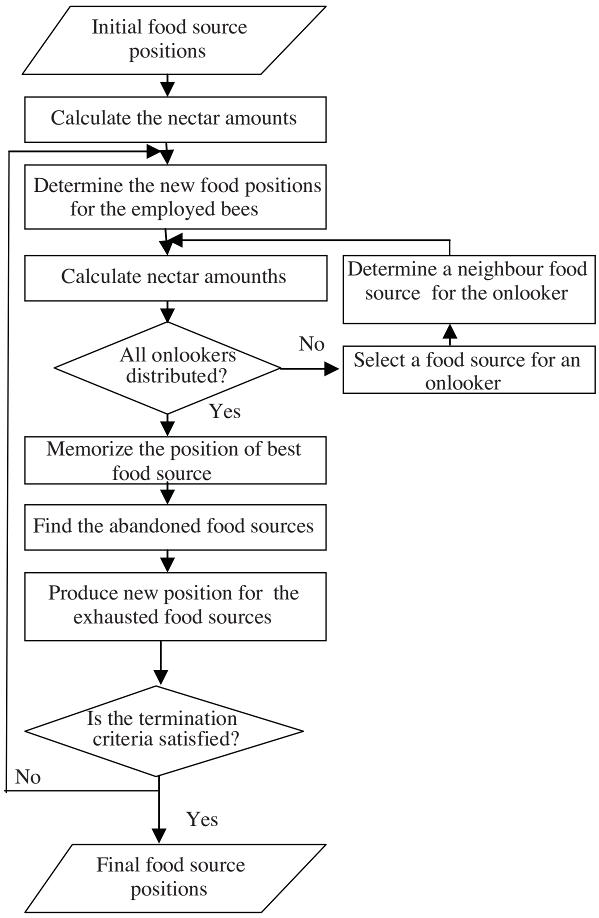


Figure 1. Flow chart of the ABC algorithm.

amounts of their sources with the onlookers waiting on the dance area. All onlookers successively determine a food source area with a probability based on their nectar amounts. Each onlooker determines a neighbor food source within the neighborhood of the one to which she has been assigned and then its nectar amount is evaluated.

A honey bee colony has scouts that are the colony's explorers who do not need any guidance while looking for food. In the ABC algorithm, if a solution representing a food source cannot be improved by a predetermined number of trials, it means that the associated food source has been exhausted by the bees and the employed bee of this food source becomes a scout. The position of the abandoned food source is replaced with a randomly produced food position. The number of trials for releasing a food source is equal to the value of "limit," which is an important control parameter of the ABC algorithm. These three steps are repeated until the termination criteria are satisfied.

Assume that w_i is the position of the i^{th} food source (i^{th} solution to the problem) and $f(w_i)$ represents its nectar amount (the quality of solution). Let $P(c) = \{w_i(c) \mid i = 1, 2, \ldots, s\}$ (c denotes the cycle; s, the number of food sources around the hive) represent the population of food source positions. If the nectar amount of a food source is high, the probability with that the source is chosen by an onlooker bee becomes high proportionally. The probability can be defined as

$$p_i = \frac{f(w_i)}{\sum\limits_{k=1}^{s} f(w_k)}, \tag{1}$$

where s is the number of food sources (number of solutions in the population) and is also equal to the number of employed bees in the colony. An onlooker bee selects a food source region depending on the probabilities calculated and determines a neighbour food source around the chosen one. This process is repeated until all onlookers are distributed among the food sources (solutions). In this way, the onlookers are recruited to the food sources with high nectar amount (the solutions with high fitness value) that have been determined by the employed bees. Assume that the position of the food source selected by the onlooker is w_i. The neighbor food source position of w_i might be determined as the following:

$$w_i(c+1) = w_i(c) + \phi_i(w_i(c) - w_k(c)) \tag{2}$$

where ϕ_i is a randomly generated number in the interval [-1, +1] and k is a randomly produced index different from i.

If the nectar amount $f(w_i(c+1))$ is greater than $f(w_i(c))$, then the artificial bee memorizes $w_i(c+1)$ and shares her information with onlooker bees; and the position $w_i(c)$ of the food source i is changed as $w_i(c+1)$, otherwise $w_i(c)$ is kept same. As mentioned before, every food source has only one employed bee; therefore the number of employed bees is equal to the number of food sources. If the position w_i cannot be improved through the predetermined number of trials "limit" of bees, the food source i is abandoned and its employed bee becomes a scout. The scout starts to search for a new food source randomly; and after finding a new source, its new position is accepted to be $w_i(c+1)$.

2.2. Modified ABC

Basic ABC algorithm has only three control parameters: colony size (number of onlookers and employed bees or food sources), maximum cycle number (termination criteria) and the limit value. In a basic ABC, a neighbor food source position around a food source is determined by changing only one parameter of the present position. This process reduces the convergence speed of ABC during the initial phase of search. In order to avoid this undesirable characteristic of basic ABC, the neighbor source position of a food source might be determined by changing more than one parameter. The ABC producing neighbor solutions in this way is called modified ABC. Modified ABC has got one more control parameter compared to basic ABC, called the modification rate-MR, which controls the frequency of parameter change in the production of a neighbor solution. The recommended value for this control parameter lies in the range [0, 1]. Modified ABC has been used for constraint optimization problems by Karaboga and Basturk in 2007 [21]. In this paper, the modified ABC is first time used for adaptive IIR filter design which is an unconstraint optimization problem.

3. Adaptive noise cancellation

Noise cancellation is a basic problem which has important applications in such areas as speech processing, echo cancellation, signal enhancement, antenna array processing, biomedical signal and image processing. Noise cancellation is the extraction of a desired signal from a noisy, corrupted signal by negating the noise [13]. Figure 2 represents the structure of an adaptive noise canceller.

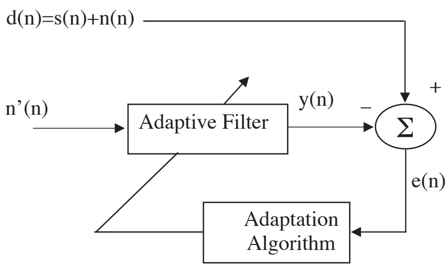


Figure 2. Adaptive noise canceller.

As seen from the Figure 2, there are four signals: the information signal $s(n)$ (an electrocardiogram (ECG) signal, in this work); the noise signal $n(n)$ corrupting the desired signal $d(n)$; and the reference signal $n'(n)$. ECG signals usually suffers from the sinusoidal noise that varies around 50 Hz produced by the line voltage; therefore, in the simulations carried out for adaptive FIR filter design, sinusoidal noises with different frequencies are added to the ECG signal. However, in simulations realized for adaptive IIR filter design, an additive white Gaussian noise (AWGN) is preferred because of its widespread use in the literature and some important features that will be mentioned in the next chapter. The reference signal (input signal) $n'(n)$ is, similarly, a correlated version of the $n(n)$. The error signal $e(n)$, which is equal to the difference between the noisy signal and the adaptive filter output $y(n)$, is computed and fed back to the adaptation algorithm to adjust the adaptive filter coefficients in order to minimize a given cost function. The Mean Squared Error (MSE) cost function to be minimized can be defined as

$$J(n) = \mathbf{E}[|e(n)|^2]. \tag{3}$$

In this equation, **E** denotes the expected value. For each algorithm, the quality of the solution i in the population is calculated by using the formula

$$fit(i) = \frac{1}{1 + J_i(w)}, \tag{4}$$

where $J_i(w)$ is the cost function defined by the equation 3.

In order to calculate the quality of a possible solution, a moving scheme is employed for computing $J(n)$. The cost function is calculated by using a block of K samples (K=1000) and the data block is shifted by 1 sample after each cycle or iteration.

The control parameter values of the ABC based [22] and other evolutionary algorithms used in the simulations are given in Table 1.

Table 1. Control parameter values used in the simulations.

Modified ABC	ABC	PSO	DE
Colony size = 30	Colony size = 30	Swarm size = 30	Population size = 30
limit value =180 Modification Rate = 0.7	limit value = 180	Inertia factor, ω= 0.4 Cognitive factor, c_1= 2 Social factor, c_2= 2	Crossover rate = 0.8 Scaling factor = 0.6
$X_{max} = 1$ X_{min} = -1	$X_{max} = 1$ X_{min} = -1	$X_{max} = 1$, X_{min} = -1 V_{max}=0.5, V_{min}=-0.5	$X_{max} = 1$ X_{min} = -1

In the simulations of adaptive FIR filter design, the sampling frequency of ECG signal is chosen as $f_s = 500$ Hz and it has been sampled at 1000 points. For a realistic simulation, instead of adding a fixed 50 Hz sinusoidal noise to entire signal, the signal is divided into five equal intervals and each interval is respectively corrupted by the sinusoidal noises with different frequencies of 46 Hz, 48Hz, 50Hz, 52Hz and 54Hz.

For 50 Hz sinusoidal noise, the time period between the two sequential sample value of the information signal can be obtained as $T = 1/f_s = 2$ ms. The period of the sinusoidal noise produced by the line will be $T = 1/f$=20 ms. As a result, into a one period of a sinus wave, 10 ECG samples will occur. So, for 1 ECG sample, the discrete time frequency of the 50 Hz sinusoidal line voltage will be 0.1 radian/sample. Hence, the expression of the 50 Hz sinusoidal noise occurs as $\sin(2 \cdot \pi \cdot 0.1 \cdot n)$.

LMS algorithm requires the choice of a most appropriate value for the step size parameter (μ) that affects the stability, steady state MSE and convergence speed of the algorithm.

In this study, the LMS algorithm, which uses a fixed step size value during the optimization, and the NLMS algorithm, whose step size is changing adaptively at each iteration and maintaining the desired stability, are used. In this paper, a new approach for finding the optimal step size value for LMS and optimal initial step size value for NLMS is proposed. In the improved approach, firstly, the value range of the step size parameter is determined and then the step size is increased with a step value of 0.001 at each iteration. The algorithm is run for each step size and then the sum of the MSE values for the current step size is calculated individually. By using these data a relation between the step size and total MSE value is proposed and from this relation the optimal value of the step size parameter for the problem is determined. Here, it is very important to note that in NLMS algorithm the μ value is adaptively updated at each iteration. So, the μ_{NLMS} calculated with the proposed method given above is the optimal initial μ value.

3.1. Using adaptive FIR filter

The transfer function of an adaptive FIR filter with the degree of N can be described by,

$$H(z) = \sum_{i=0}^{N} b_i z^{-i}. \tag{5}$$

In the above equation, b_i represents the time-varying parameters of the adaptive FIR filter which are being updated by adaptation algorithm so as to minimize the MSE value at each iteration. LMS, NLMS, ABC and modified ABC algorithms are used to the design adaptive FIR filter and their performances compared. The degree of the designed filter is chosen as 7 and the ECG signal is corrupted by the sinusoidal noises with different frequencies as discussed in the previous chapter. The proposed method for finding the optimal step size value was used for both LMS and NLMS algorithms and obtained results are given in Figure 3.

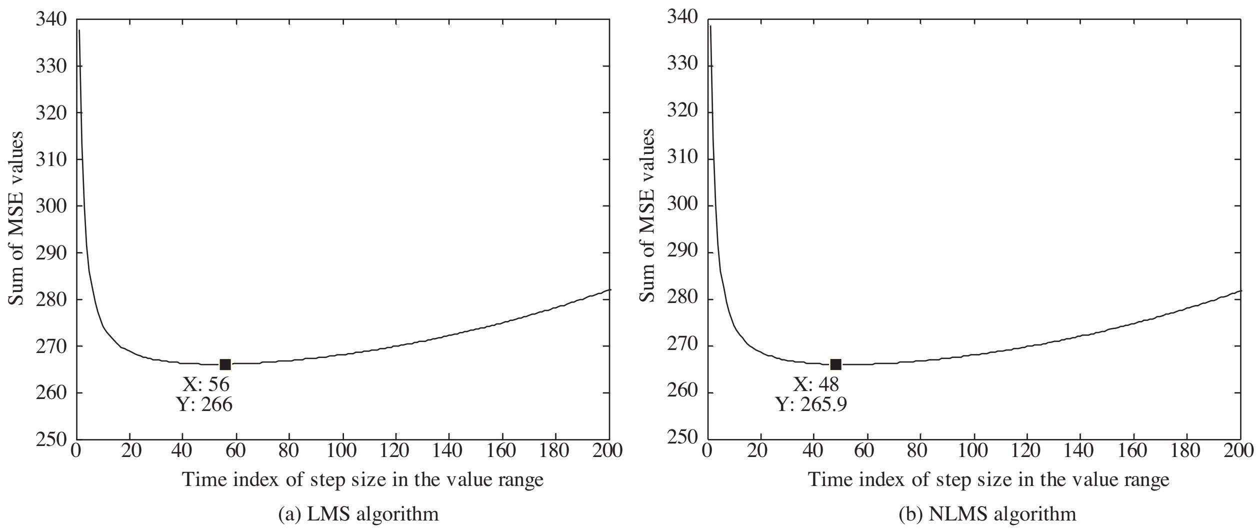


Figure 3. Optimal step size values found for LMS and NLMS algorithms.

In simulation, the value range of the step size parameter for LMS algorithm was taken as μ=0.001:0.001:1. As seen from the figure given for LMS algorithm, for the fifty-sixth value of this interval, namely for $\mu = 0.056$, the sum of the MSE values becomes minimum ($\sum MSE = 266$). So, $\mu = 0.056$ is the optimal step size value of LMS algorithm for this problem. Similarly, μ=0.001:0.001: 2 is the value range for μ_{NLMS} and as seen from the figure obtained for NLMS, the forty-eighth value of this range, that is $\mu = 0.048$, produces the minimum total MSE ($\sum MSE = 265.9$)and hence it is the best initial step size value of NLMS algorithm. Figure 4 shows the information signal and corrupted signal being equal to the sum of information signal and sinusoidal noise.

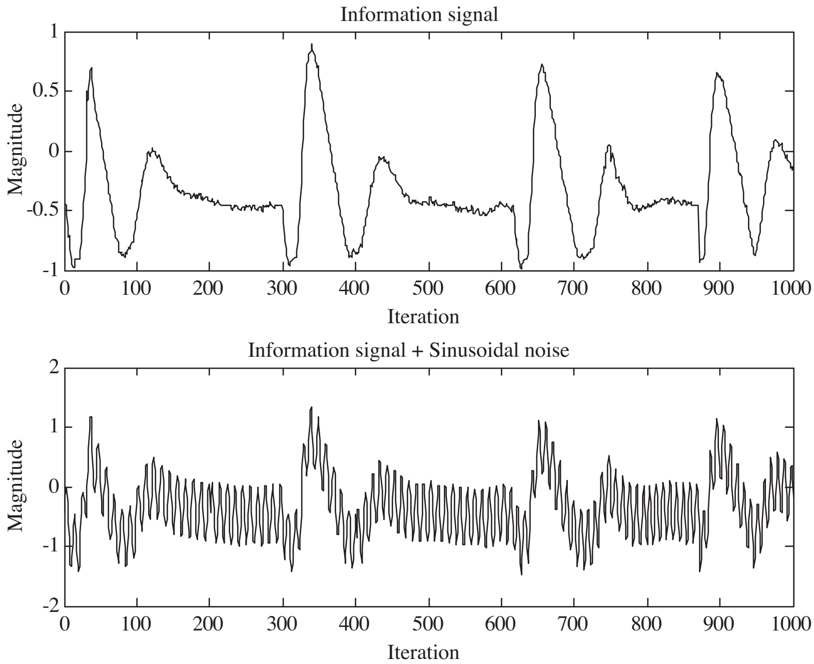


Figure 4. The main signals used in the simulations for adaptive FIR filter design.

When the corrupted signal in Figure 4 is filtered using the adaptive FIR filters designed by the LMS, NLMS, ABC and modified ABC algorithms, the filtered ECG signals obtained are those shown in Figure 5. It is clear from the figures that, except the transient points in which the frequency of the applied noise changes, both the LMS and ABC based algorithms have similar performances. But in the transient points the performance of the ABC based algorithms is much better than the LMS based algorithms.

To compare the performance of the algorithms, Figure 6 represents the convergence speed of the algorithms. Population based algorithms perform CS (colony size) number of computations at each cycle while the gradient based algorithms perform only one computation at each iteration. In this paper, the colony size for ABC based algorithms is chosen as 30, namely, ABC based algorithms perform 30 computation at each cycle. LMS and NLMS algorithms perform 1 computation at each iteration, as a result, 1000 iteration of gradient based algorithms corresponds to 34 cycle of ABC based algorithms. For a fair comparison, from the Figure 6(c), when the 34^{th} cycle of the ABC based algorithms and 1000^{th} iteration of the gradient based algorithms are compared it is clearly seen that ABC based algorithms have better performances in terms of the convergence speed and minimum MSE error. The minimum MSE value and the best convergence speed is obtained when the modified ABC algorithm is used. Although, the MSE performance of LMS and NLMS algorithms is very close to each other, the convergence speed of the NLMS algorithm is better than the LMS algorithm. The filter parameters and the averaged MSE values of 20 random runs are given in Table 2. From the table it is clear that, ABC based approaches can also converge to better averaged MSE values.

Figure 7 demonstrates the evolution of the parameters for ABC and modified ABC algorithms during the adaptation process for the run in which the minimum MSE value is obtained. As seen from the figure given for basic ABC, it is observed that the evolution might show instantaneous changes. However, the evolution process of the parameters for modified ABC algorithm seems more stable since the values are changing gradually.

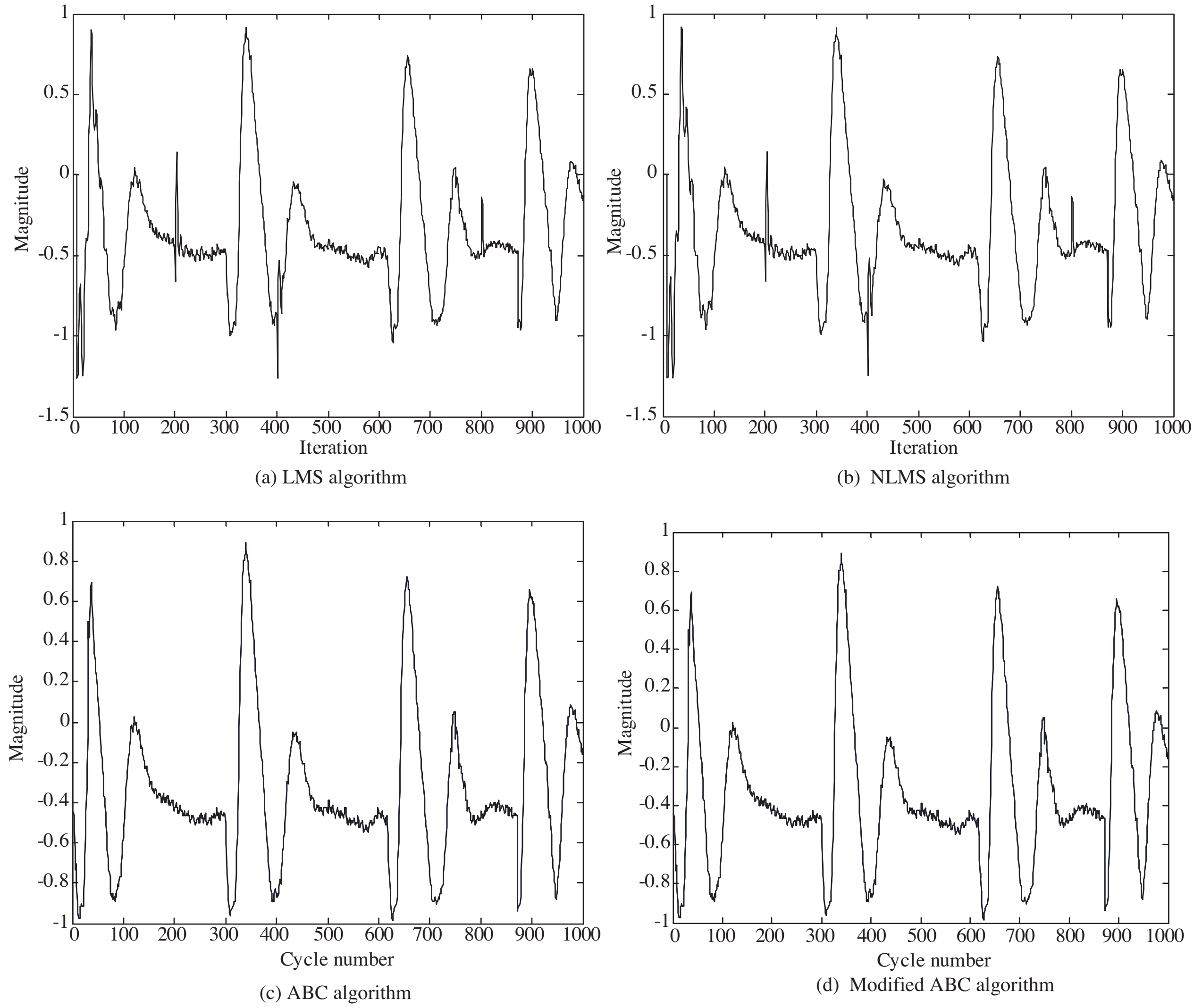
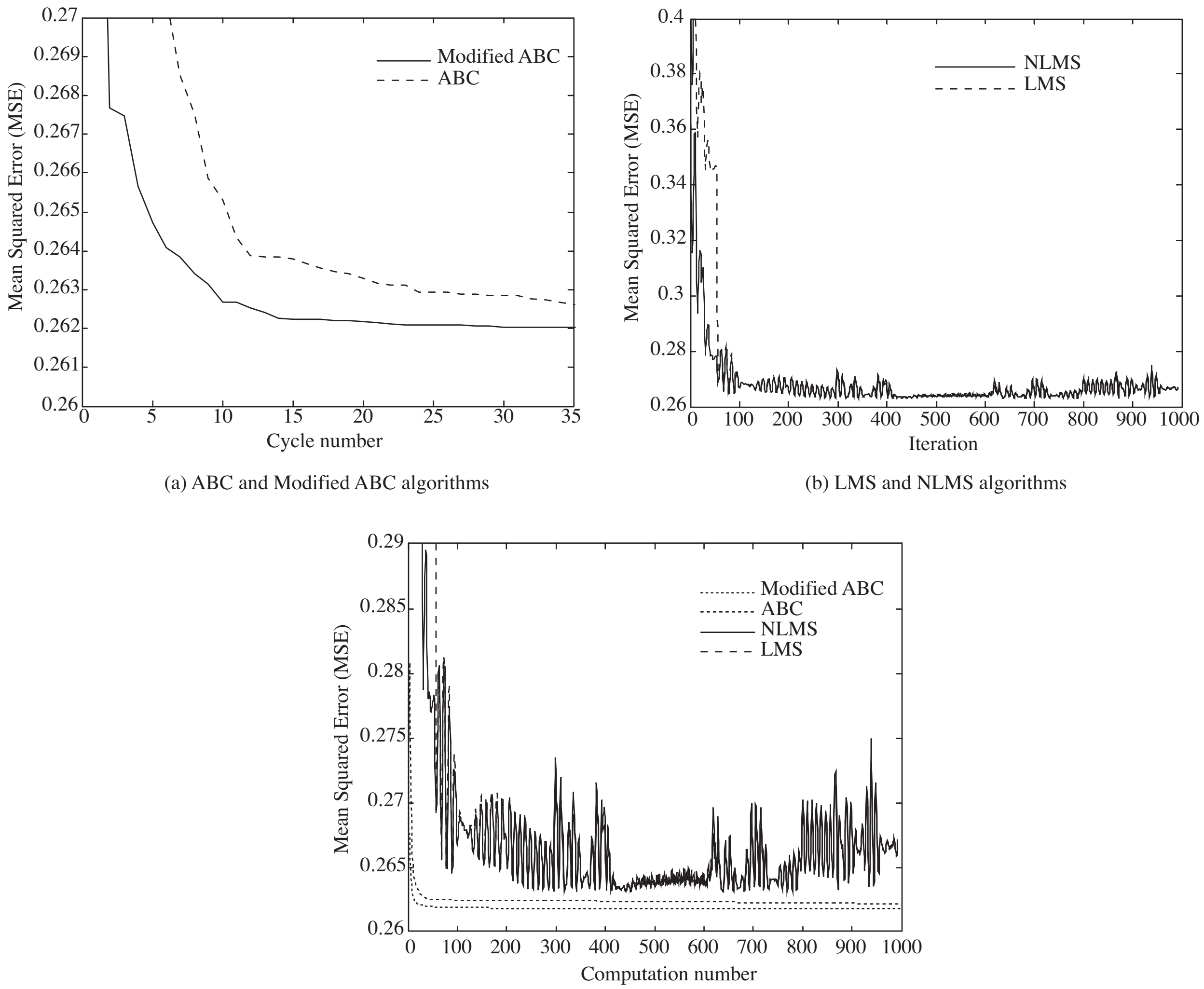


Figure 5. The resulting ECG signals after filtering the sinusoidal noises with different frequencies by using adaptive FIR filters designed by LMS, NLMS, ABC and modified ABC algorithms.

Table 2. The averaged MSE values and the best parameter values found by LMS, NLMS, ABC and modified ABC algorithms.

	LMS (MSE=0.267163)	NLMS (MSE=0.266381)	ABC (MSE=0.261926)	Modified ABC (MSE=0.261774)
b_0	0.2454	0.2471	-0.5134	-0.5951
b_1	0.1011	0.1034	0.3863	0.3304
b_2	-0.0686	-0.0665	-0.2264	-0.1315
b_3	-0.2003	-0.1983	-0.1192	-0.2984
b_4	-0.2589	-0.2589	-0.3423	-0.3088
b_5	-0.2408	-0.2427	-0.1899	-0.1156
b_6	-0.167	-0.1688	0.5261	0.3487
b_7	-0.072	-0.0707	-0.6528	-0.6356



(a) ABC and Modified ABC algorithms

(b) LMS and NLMS algorithms

(c) Performance comparison of evolutionary and gradient based algorithms

Figure 6. Cost function value versus number of iterations averaged over 20 random runs for ABC, modified ABC, LMS and NLMS algorithms.

3.2. Using adaptive IIR filter

The transfer function of an adaptive IIR filter can be given as

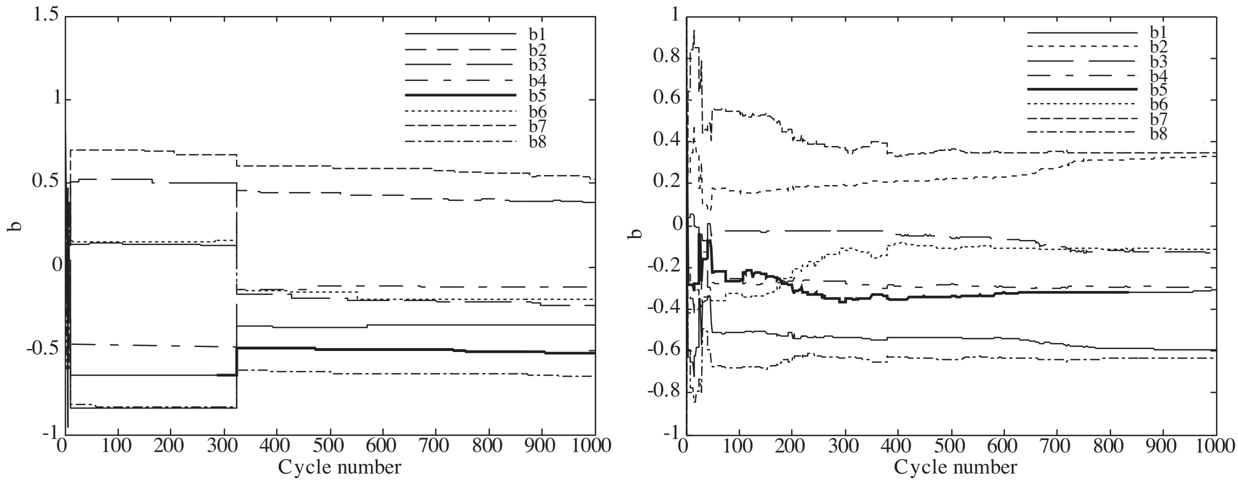
$$H(z) = \frac{\sum_{i=0}^{M} b_i z^{-i}}{1 + \sum_{i=1}^{N} a_i z^{-i}}, \tag{6}$$

where b_i and a_i represent the coefficients of the filter, respectively and $N\,(\geq M)$is the filter order. The time-variable coefficient vector can be determined as the string form

$$w = [b_0\; b_1\; ...\; b_M\;\; a_1\; a_2\; ...\;\; a_N]^T. \tag{7}$$

To satisfy the stability of designed adaptive IIR filter, all poles of the filter must be located inside the unit circle in the z-domain.

In the simulation, an additive white Gaussian noise is preferred as a noise signal because of its power spectrum properties. White noise is defined as an uncorrelated noise process with equal power at all frequencies. In theoretical concept, it has an infinite power that covers an infinite range of frequencies. By adding the white Gaussian noise to the information signal, the information signal is corrupted and, as a result, a noisy signal is obtained. In this section, simulations are realized by using ABC, modified ABC, PSO and DE algorithms and then performances of the algorithms are compared. The information and corrupted signals are given in Figure 8.



(a) ABC algorithm (b) Modified ABC algorithm

Figure 7. Evolution of the parameters of the adaptive FIR filter for ABC and modified ABC algorithms.

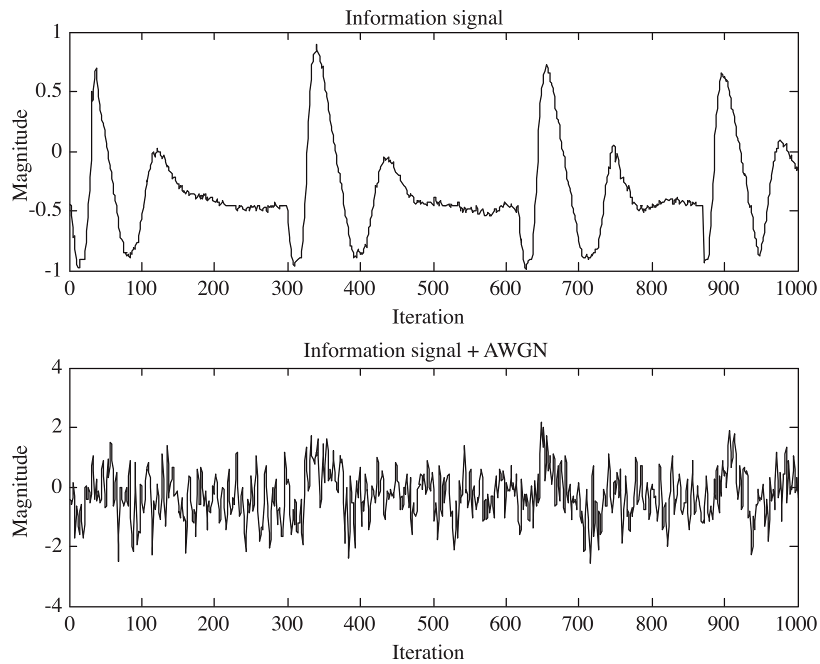


Figure 8. The main signals used in the simulations for adaptive IIR filter design.

After the ECG signal with the white Gaussian noise is filtered by using the adaptive IIR filters designed using ABC, modified ABC, PSO and DE algorithms, the ECG signals obtained for each algorithm is presented in Figure 9. It is clear from the figure that all of the adaptive IIR filters designed by the algorithms successfully remove the noise from the information signal.

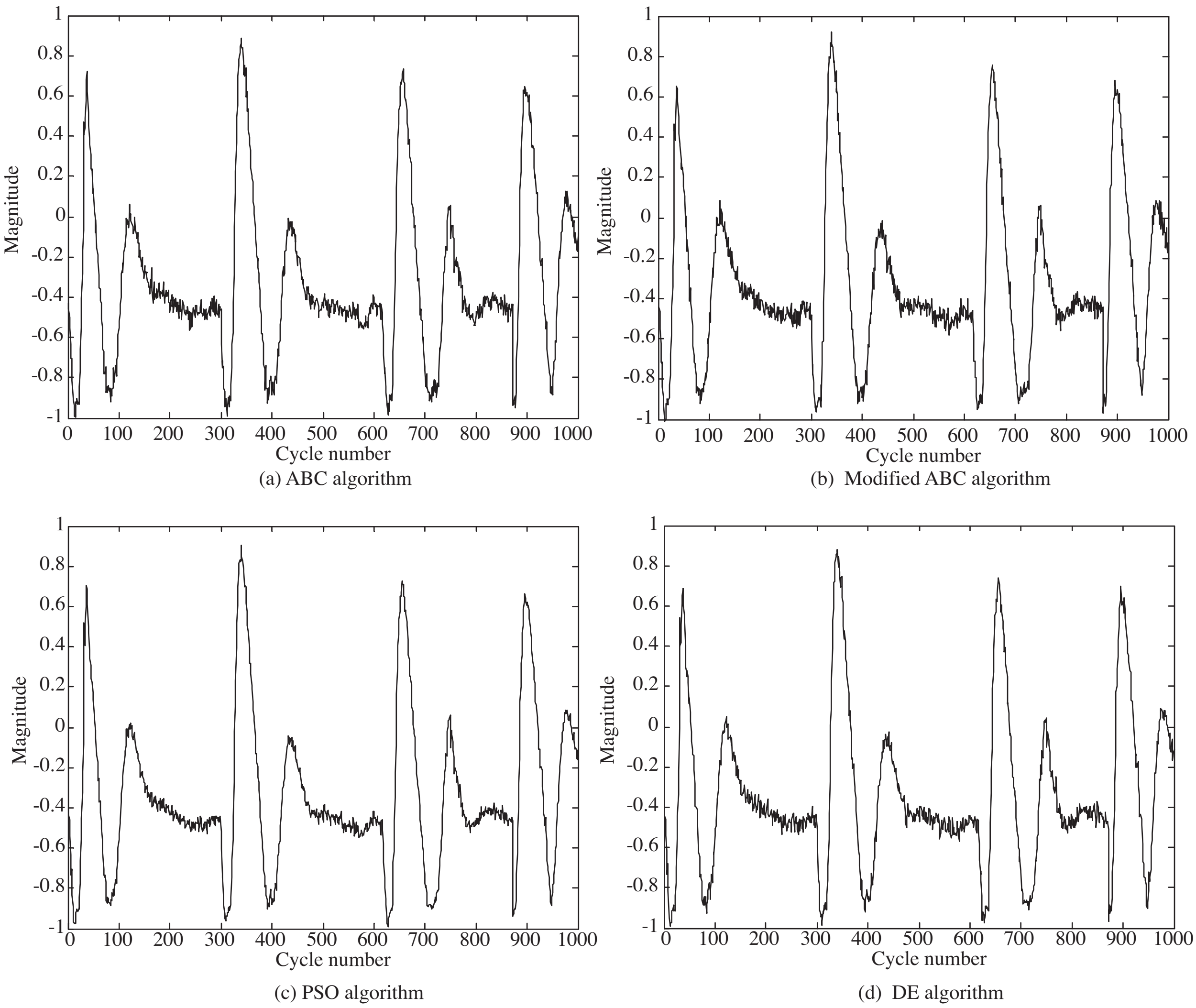


Figure 9. The resulting ECG signals after filtering the AWGN by using ABC, modified ABC, PSO and DE algorithms.

Figure 10 demonstrates the convergence speeds of the algorithms; each algorithm produces similar results in terms of the mean squared error, but the modified ABC converges to the global solution faster than the other algorithms. The modified ABC needs approximately 35 cycles to converge although ABC and PSO needs 100 cycles and DE needs 180 cycles.

Figure 11 and Figure 12, respectively, demonstrates the evolution of the parameters for ABC and modified ABC algorithms during the adaptation process for the run in which the minimum MSE value is obtained.

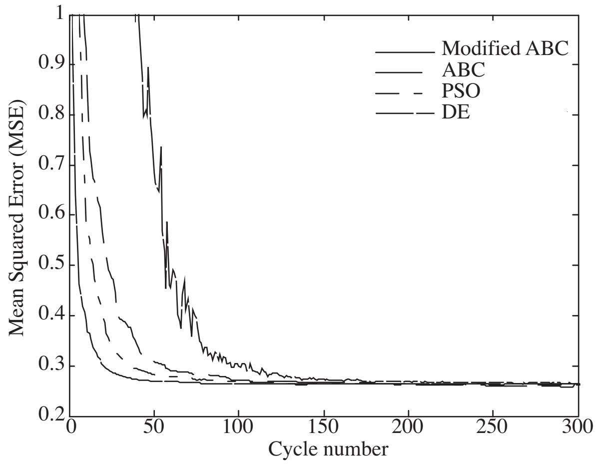


Figure 10. Cost function value versus cycle number averaged over 20 random runs for ABC, modified ABC, PSO and DE algorithms.

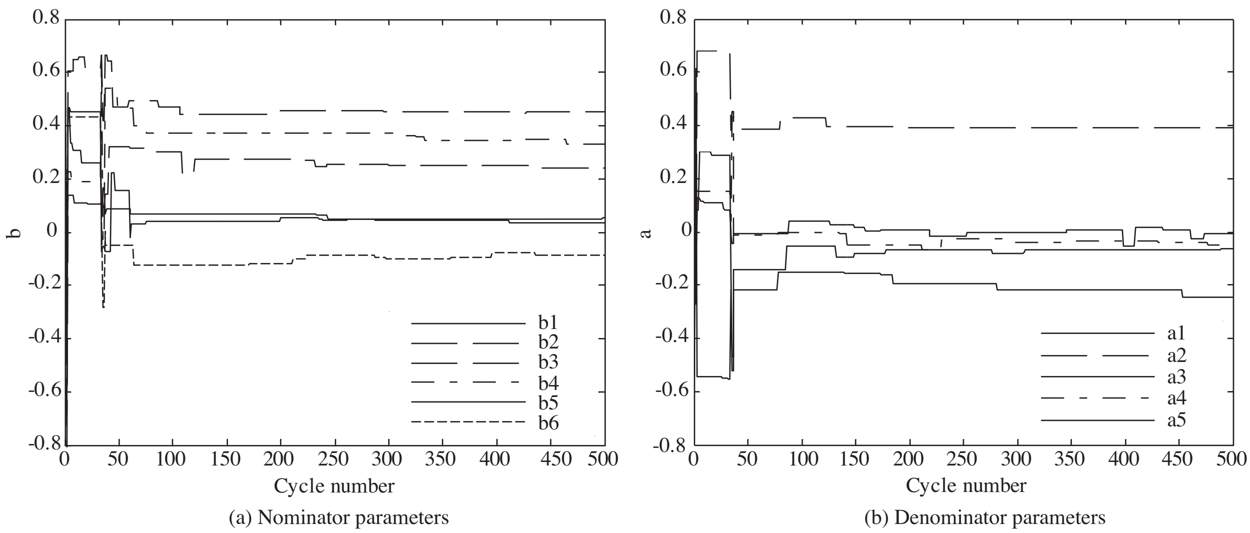
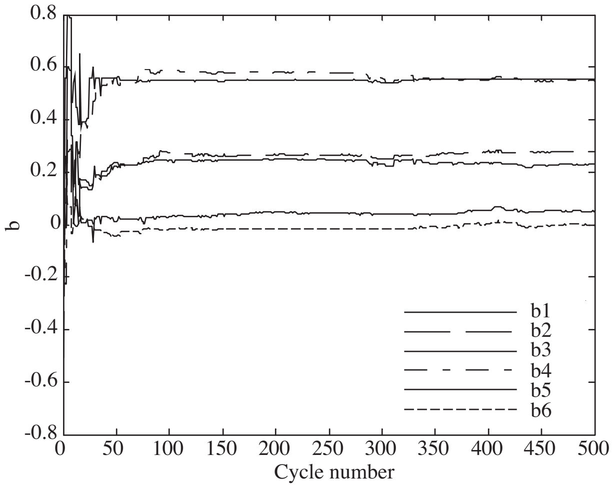
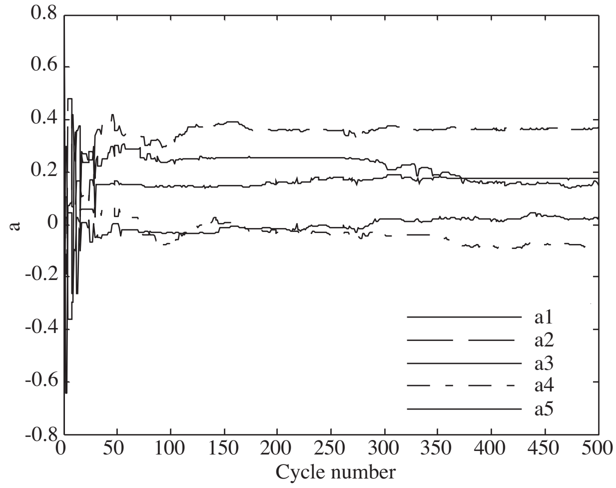


Figure 11. Evolution of the nominator and denominator parameters of the adaptive IIR filter for ABC algorithm.

The parameter values of the best filters designed by the modified and basic ABC algorithms are given in Table 3. The averaged MSE values of 20 random runs in Table 3 show that, the modified ABC algorithm produces the best averaged MSE value.



(a) Nominator parameters



(b) Denominator parameters

Figure 12. Evolution of the nominator and denominator parameters of the adaptive IIR filter for modified ABC algorithm.

Table 3. The parameter values of the best filters designed by ABC, modified ABC, PSO and DE algorithms.

	ABC (MSE=0.264168)	Modified ABC (MSE=0.262445)	PSO (MSE=0.264164)	DE (MSE=0.264288)
a_1	-0.2475	0.1485	-0.2847	-0.1843
a_2	0.3920	0.3733	0.4974	0.4112
a_3	-0.0644	0.1746	-0.0733	0.0149
a_4	-0.0486	-0.0759	-0.0412	-0.0722
a_5	-0.0066	0.0306	0.0249	0.0261
b_0	0.0523	0.0521	0.0528	0.0582
b_1	0.2400	0.2768	0.2511	0.2566
b_2	0.4532	0.5507	0.4419	0.4707
b_3	0.3304	0.5467	0.3476	0.3672
b_4	0.0369	0.2293	0.0698	0.0786
b_5	-0.0871	-0.0025	-0.0455	-0.0534

4. Conclusion

In this work, a novel approach based on ABC algorithm for adaptive FIR and IIR filter design was described for the purpose of noise cancellation. The performance of the proposed modified ABC algorithm was firstly examined for the design of adaptive FIR filters with uni-modal error surface and its performance was compared with basic ABC, LMS and NLMS algorithms. Then the proposed algorithm was used for the design of adaptive IIR filters with multi-modal error surface and its performance was compared to that of basic ABC, PSO and DE algorithms. Simulation results show that, the performance of modified ABC algorithm in terms of the convergence speed and final averaged mean squared error is similar or better than the LMS, NLMS, PSO and DE algorithms although ABC is as simple as PSO and DE algorithms. It can be concluded that, ABC algorithm based approach can successfully be used for designing adaptive FIR and IIR filters for noise cancellation.

References

[1] Paulo S. R. Diniz, Adaptive Filtering Algorithms and Practical Implementations, Springer, USA, 2008.

[2] S. Haykin, Adaptive Filter Theory, Prentice Hall, USA, 2002.

[3] D. J. Krusienski, W. K. Jenkins, Design and performance of adaptive systems based on structured stochastic optimization strategies, IEEE Circuits Systems Magazine 5 (2005) 8-20.

[4] S. C. Ng, S. H. Leung, C. Y. Chung, A. Luk, W. H. Lau, The genetic search approach: A new learning algorithm for adaptive IIR filtering, IEEE Signal Processing Magazine 13 (1996) 38-46.

[5] N. Karaboga, Digital IIR filter design using differential evolution algorithm, EURASIP Journal on Applied Signal Processing 8 (2005) 1-9.

[6] A. Kalinli, N. Karaboga, A parallel tabu search algorithm for digital filter design, COMPEL-The International Journal for Computation and Mathematics in Electrical and Electronic Engineering 24 (2005) 1284-1298.

[7] N. Karaboga, B. Cetinkaya, Design of digital FIR filters using differential evolution algorithm, Circuits Systems and Signal Processing Journal 25 (2006) 649-660.

[8] D. J. Krusienski, W. K. Jenkins, Adaptive filtering via particle swarm optimization, 37^{th} Asilomar Conference on Signals Systems and Computers, 2003, pp. 571-575.

[9] D. J. Krusienski, W. K. Jenkins, Particle swarm optimization for adaptive IIR filter structures, Congress on Evolutionary Computation, 2004, pp. 965-970.

[10] A. Kalınlı, N. Karaboga, A new method for adaptive IIR filter design based on tabu search algorithm, International Journal of Electronics and Communication 59 (2004) 1-7.

[11] S. Chen, B. L. Luk, Adaptive simulated annealing for optimization in signal processing applications, Signal Processing 79 (1999) 117-128.

[12] N. Karaboga, A new design method based on artificial bee colony algorithm for digital IIR filters, Journal of the Franklin Institute-Engineering and Applied Mathematics 346 (2009) 328-348.

[13] A. P. Engelbrecht, Fundamentals of computational swarm intelligence, John Wiley & Sons Publication, Chichester, UK, 2005.

[14] D. Karaboga, An idea based on honey bee swarm for numerical optimization. Technical Report-TR06, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.

[15] D. Karaboga, B. Basturk, A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm, Journal of Global Optimization 39 (2007) 459-471.

[16] R. S. Rao, S.V.L. Narasimham, M. Ramalingaraju, Optimization of distribution network configuration for loss reduction using artificial bee colony algorithm, International Journal of Electrical Power and Energy Systems Engineering 1 (2008) 116-122.

[17] A. Singh, An artificial bee colony algorithm for the leaf-constraint minimum spanning tree problem, Applied Soft Computing 8 (2008) 687-697.

[18] P. W. Tsai, J. S. Pan, B. Y. Liao, S. C. Chu, Interactive artificial bee colony algorithm, International Symposium on Intelligent Informatics, 2008, pp. 247-251.

[19] S. Hemamalini, S. P. Simon, Economic load dispatch with valve-point effect using artificial bee colony algorithm, 32^{th} National Systems Conference, 2008, pp. 17-19.

[20] T. Kurban, E. Besdok, A comparison of RBF neural network training algorithms for inertial sensor based terrain classification, Sensors Journal 9 (2009) 6312-6329.

[21] D. Karaboga, B. Basturk, Artificial bee colony (ABC) optimization algorithm for solving constrained optimization problems, LNCS: Advances in Soft Computing: Foundations of Fuzzy Logic and Soft Computing, 2007, pp.789-798.

[22] B. Akay, D. Karaboga, Parameter Tuning for the Artificial Bee Colony Algorithm, Lecture Notes in Artificial Intelligence, 2009, pp.608-619.