

## Research Article

# A Multiuser Detector Based on Artificial Bee Colony Algorithm for DS-UWB Systems

Zhendong Yin, Xiaohui Liu, and Zhilu Wu

*School of Electronics and Information Engineering, Harbin Institute of Technology, Harbin 150001, China*

Correspondence should be addressed to Zhendong Yin; [zgcZR2005@yahoo.com.cn](mailto:zgcZR2005@yahoo.com.cn)

Received 3 June 2013; Accepted 10 July 2013

Academic Editors: P. Agarwal, V. Bhatnagar, J. Yan, and Y. Zhang

Copyright © 2013 Zhendong Yin et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Artificial Bee Colony (ABC) algorithm is an optimization algorithm based on the intelligent behavior of honey bee swarm. The ABC algorithm was developed to solve optimizing numerical problems and revealed promising results in processing time and solution quality. In ABC, a colony of artificial bees search for rich artificial food sources; the optimizing numerical problems are converted to the problem of finding the best parameter which minimizes an objective function. Then, the artificial bees randomly discover a population of initial solutions and then iteratively improve them by employing the behavior: moving towards better solutions by means of a neighbor search mechanism while abandoning poor solutions. In this paper, an efficient multiuser detector based on a suboptimal code mapping multiuser detector and artificial bee colony algorithm (SCM-ABC-MUD) is proposed and implemented in direct-sequence ultra-wideband (DS-UWB) systems under the additive white Gaussian noise (AWGN) channel. The simulation results demonstrate that the BER and the near-far effect resistance performances of this proposed algorithm are quite close to those of the optimum multiuser detector (OMD) while its computational complexity is much lower than that of OMD. Furthermore, the BER performance of SCM-ABC-MUD is not sensitive to the number of active users and can obtain a large system capacity.

## 1. Introduction

Since the concept of ultra-wideband (UWB) technology was put forward by FCC in the 1990s, UWB technique has drawn a lot of attention in the theoretic research, industrial application, and many other areas because of its attractive features such as high data transmission rate, low power density, high interference resistance, and strong multipath resolution [1–3].

Multiuser detection (MUD) is a method to eliminate the effect of multiple access interference (MAI). The multiple accesses are commonly divided into two paths: time hopping (TH-UWB) and direct sequence (DS-UWB) [4]. In recent years, the increase in demand for multiple access applications with high data transmission rates has prompted the development of multiuser detection (MUD) techniques in such systems to suppress the MAI and improve the system performance. Optimum multiuser detection (OMD) was provided by Verdu in 1986 [5]; OMD makes the BER performance of multiple access system approximate to single-user system. But it has a high computational complexity, so it has poor

real-time characteristic confines in engineering. Therefore, suboptimal detectors which may approximate OMD's BER performance with an acceptable computational complexity have become a focus of research. A code-aided interference suppression method was introduced for NBI restriction in DS-UWB systems [6]. A multiuser frequency-domain (FD) turbo detector was employed which combines FD turbo equalization schemes with soft interference cancellation [7]. Adaptive MUD methods using the recursive least square (RLS) principles were proposed in [8, 9]. However, the trade-off problem between BER performance and computational complexity still exists.

Swarm intelligence is a research branch that models the population of interacting agents or swarms that are able to self-organize. Several modern heuristic algorithms have been developed for solving combinatorial and numeric optimization problems [10]. Artificial Bee Colony (ABC) algorithm is an optimization algorithm based on the intelligent behavior of honey bee swarm [11, 12], and some extended and efficient

ABC algorithms are presented to improve the performance of ABC [13, 14].

It has been demonstrated that ABC algorithm for numerical optimization problems has more superior performance than those based on heuristics algorithm [15]. ABC algorithm has been widely used in neural network training practice [16], path optimization [17], and back analysis [18]. However, no literatures have been reported that ABC was used in the multiuser detection (MUD) fields. In order to possess a good BER performance and a low time complexity, we investigate an efficient multiuser detector by selection of initial states based on code mapping for the Artificial Bee Colony algorithm (SCM-ABC-MUD) in DS-UWB systems. As a kind of swarm intelligence methods, ABC is selected here for its significant ability to search for the global optimal value and to adapt its searching space automatically [19, 20]. And its basic motivation is to find the global optimum by simulating the bees' behaviors [21]. This proposed algorithm makes use of the thought of ABC's iterative optimization and the result of the suboptimal detectors based on code mapping. Simulation results show that the BER performance and near-far effect (NFE) resistance capability of this algorithm are better than those of matched filter (MF), DEC, and MMSE detectors and even close to OMD.

The paper is organized as follows. In Section 2, the MUD model is introduced and a suboptimal detector based on code mapping is proposed. Then in Section 3, the basic principles of the Artificial Bee Colony (ABC) algorithm and the proposed SCM-ABC-MUD algorithm are illustrated. In Section 4, simulation experiments that compare the performance of different MUD algorithms are analyzed, followed by conclusions given in Section 5.

## 2. MUD Models and Methods

**2.1. Multiuser DS-UWB System Model.** In theory, a  $K$ -user synchronous DS-UWB system under the additive white Gaussian noise (AWGN) channel is considered which is not subjected to the frequency selective multipath. And assume each user employs the binary phase-shift key (BPSK) modulation. Then the  $k$ th user's transmit signal can be written as

$$x_k(t) = \sum_{i=1}^M \sum_{j=0}^{N_c-1} d_k(i) c_k(t - (i-1)T_s) p(t - (i-1)T_s - jT_c), \quad (1)$$

where  $M$  is the length of bits per packet and BPSK symbols  $d_k(i) \in \{-1, 1\}_{i=1}^M$  are spread with the specific PN codes  $c_k(t)$ , which are the binary bit stream valued only by  $-1$  or  $1$ .  $T_s$  is the symbol duration,  $T_c$  is the pulse repetition period,  $N_c$  equals to  $T_s/T_c$ , and  $p(t)$  represents the transmitted pulse waveform generally characterized as the second derivative of Gaussian pulse

$$p(t) = \left[ 1 - 4\pi \left( \frac{t - t_d}{\tau_m} \right)^2 \right] \exp \left[ -2\pi \left( \frac{t - t_d}{\tau_m} \right)^2 \right], \quad (2)$$

where  $t_d$  and  $\tau_m$  are the pulse center and the pulse shape parameter.

The total received signal composed by different signals of all users is

$$r(t) = v(t) + n(t) = \sum_{k=1}^K A_k x_k(t) + n(t), \quad (3)$$

where  $A_k$  is the amplitude of the  $k$ th received signal and  $n(t)$  is zero-mean additive white Gaussian noise with the unilateral power spectral density of  $N_0$ .

### 2.2. Classical Multiuser Detectors

**2.2.1. Matched Filters (MFs).** The traditional receiver of a DS-UWB system consists of a pulse demodulator and a set of matched filters (MFs) corresponding to each user. Let the output of a bank of single-user MFs be a  $K$ -dimensional vector  $\mathbf{y} = [y_1, y_2, \dots, y_K]^T$ , the vector  $\mathbf{b} = [b_1, b_2, \dots, b_K]^T$  represent the output of sign detectors, the vector  $\mathbf{d} = [d_1, d_2, \dots, d_K]^T$  denotes the correct bits of each user, and the vector  $\mathbf{n} = [n_1, n_2, \dots, n_K]^T$  denotes the output of noise from matched filters which is a zero mean Gaussian random. So, the output of the MFs can be represented as follows:

$$\mathbf{y} = \mathbf{R}\mathbf{A}\mathbf{b} + \mathbf{n}, \quad (4)$$

$$\mathbf{b} = \text{sgn}(\mathbf{y}), \quad (5)$$

where  $R = (r_{ij})_{K \times K}$  denotes the cross-correlation matrix, in which  $r_{ij} = \sum_{l=0}^{N_c-1} c_i(l)c_j(l)$ , and  $\mathbf{A} = \text{diag}(A_1, A_2, \dots, A_K)$  in which the diagonal element  $A_k$  ( $k \in [1, K], k \in N$ ) represents the signal amplitude of the  $k$ th user.

**2.2.2. Optimum Multiuser Detection (OMD).** According to the theory of OMD, the optimum detection result satisfies the following expression:

$$\mathbf{b}_{\text{OMD}} = \arg \left\{ \max_{\mathbf{b} \in \{-1, 1\}} \left( 2\mathbf{b}^T \mathbf{A}\mathbf{y} - \mathbf{b}^T \mathbf{A}\mathbf{R}\mathbf{A}\mathbf{b} \right) \right\}. \quad (6)$$

It is known that the selection of this optimal solution  $\mathbf{b}_{\text{OMD}}$  in the  $K$ -dimensional Euclidean solution space is generally a nondeterministic polynomial hard problem, but the computational complexity of the OMD method is  $O(K^2)$ , and  $K$  is the number of active users.

**2.2.3. Suboptimal Multiuser Detection Based on Code Mapping (SCM).** In order to get a suboptimal solution, the candidate bits set output from the matched filters mapped to a one-dimensional feature space using a mapping function.

Let

$$F(\mathbf{b}) = \frac{1}{2} \mathbf{b}^T \mathbf{A}\mathbf{R}\mathbf{A}\mathbf{b} - \mathbf{b}^T \mathbf{A}\mathbf{y}. \quad (7)$$

According to (6), if the elements in  $\mathbf{b}$  are all right, the value of  $F(\mathbf{b})$  will achieve the minimum. Making a partial derivation of (7), we get

$$\frac{\partial F}{\partial \mathbf{b}} = \mathbf{H}\mathbf{b} - \mathbf{A}\mathbf{y}. \quad (8)$$

By expanding (8), we get a  $K$ th-order linear equations given as follows:

$$\begin{aligned} \frac{\partial F}{\partial b_1} &= \sum_{j=1}^K A_1 A_j r_{1j} b_j - A_1 y_1, \\ \frac{\partial F}{\partial b_2} &= \sum_{j=1}^K A_2 A_j r_{2j} b_j - A_2 y_2, \\ &\vdots \\ \frac{\partial F}{\partial b_K} &= \sum_{j=1}^K A_K A_j r_{Kj} b_j - A_K y_K. \end{aligned} \quad (9)$$

Now, we take into account that MAI is the main interference resource which largely affects the performance of the system. Let  $L(b_i) = \sum_{j=1}^K A_i A_j r_{ij} b_j - A_i y_i$ ,  $i = 1, 2, \dots, K$ . Bringing the candidate codes set  $\mathbf{b}$  into (9), there are two situations illustrated as follows.

- (1) No wrong code in  $\mathbf{b}$ . Based on the theory of extreme value, if MAI is the only interference resource without AWGN and the elements in  $\mathbf{b}$  are all correct, the result of (9) strictly equals  $\mathbf{0}$ . And, in the condition of high SNR, bringing (4) to  $L(b_k)$ , we can see that  $L(b_k) = -A_i n_i \sim N(0, A_i^2 N_c N_0 / 2)$  ( $k = 1, 2, \dots, K$ ) when there is no error bit in the candidate codes set  $\mathbf{b}$ .
- (2) Wrong codes exist in  $\mathbf{b}$ . Supposing  $b_i$  ( $i \in [1, K]$ ,  $i \in N$ ) is the wrong code (in other words,  $-b_i$  is the correct code), the other codes are correct. Substituting it to the  $i$ th equation of (9), we get

$$\begin{aligned} L(b_i) &= \sum_{\substack{j=1 \\ j \neq i}}^K A_i A_j r_{ij} b_j - A_i y_i + A_i^2 r_{ii} b_i \\ &= \sum_{\substack{j=1 \\ j \neq i}}^K A_i A_j r_{ij} b_j - A_i y_i + A_i^2 r_{ii} (-b_i) + 2A_i^2 r_{ii} b_i \\ &= 2A_i^2 r_{ii} b_i - A_i n_i. \end{aligned} \quad (10)$$

As for  $k$  which does not equal  $i$ , we get

$$L(b_k) = 2A_i A_k r_{ik} b_i - A_k n_k, \quad k = 1, 2, \dots, K, \quad k \neq i. \quad (11)$$

According to (10) and (11), it can be seen when the  $i$ th user's code is wrong,  $|L(b_i)| \gg |L(b_k)|$ ,  $k = 1, 2, \dots, K, k \neq i$ . Therefore, the function  $L(\mathbf{b})$  can obviously differentiate the wrong codes and the right codes through the absolute value of it. In addition,  $L(\mathbf{b})$  is a  $K$ th-order linear equation which can get the result of  $L(\mathbf{b})$  without complex computations.

It is accessible to map  $b$  into a one-dimensional feature space  $|L(\mathbf{b})|$  to identify the wrong codes in the candidate set. Figure 1 shows an example of the feature space mapping in the scenario of 10 users, with 6 dB signal-to-noise ratio (SNR).

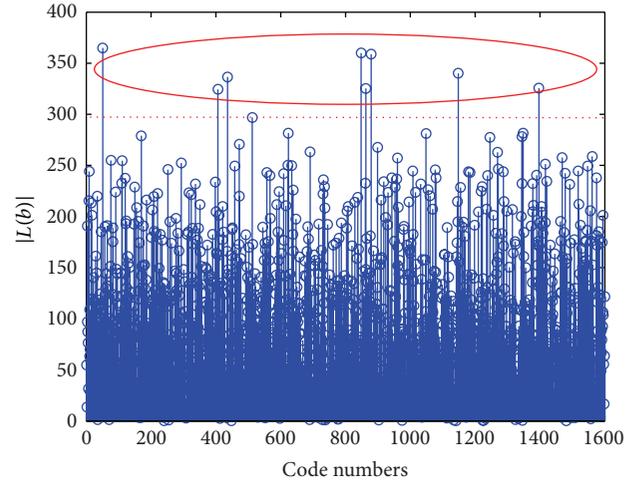


FIGURE 1: Relationship between code numbers and mapping function  $|L(\mathbf{b})|$ .

In Figure 1, it is clear that the wrong codes and the right ones have a significant difference in the feature space mapped by  $|L(\mathbf{b})|$ . Almost all the values of  $|L(\mathbf{b})|$  corresponding to the wrong codes are greater than 280; in contrast the values of the right ones are smaller than 280. C-means clustering approach [22] is used to classify the candidate codes into right bits and wrong ones. The C-means clustering is a popular unsupervised clustering algorithm based on the partition of data, which can finish the classification without training samples.

In conclusion, the whole proposed approach can perform the multiuser detection with 3 steps. First, map the candidate bits set  $\mathbf{b}$  into a one-dimensional feature space by mapping function  $|L(\mathbf{b})|$ . Second, aggregate and pick out the wrong bits among the candidate bits set by code clustering. And finally, correct the wrong bits to obtain a suboptimal solution.

### 3. The Proposed SCM-ABC-MUD Algorithm

**3.1. The Basic Principle of ABC Algorithm.** In the ABC algorithm, the colony of artificial bees consists of three kinds of bees: employed bees, onlookers, and scouts. In the algorithm, the number of employed bees is equal to the number of food sources around the hive. The employed bee whose food source is exhausted by the employed and onlooker bees becomes a scout. The position of a food source represents a possible solution of the optimization problem, and the nectar amount of a food source corresponds to the quality (fitness) of the associated solution. The number of the onlooker bees or the employed bees is equal to the number of food sources in the population.

The general scheme of the ABC algorithm is as follows.

**Step 1.** The first step is initialization. In this step, a set of food source positions is generated and distributed randomly. After initializing the solution population, the fitness value of each food source position is evaluated. In ABC algorithm, the fitness of each solution is the value of objective function. The ABC has some control parameters: food sources number (the number of food sources equal to the employed bees); limit,

the value of predetermined number of cycles; and MCN, the maximum cycle number. Initially, the values of these control parameters are assigned.

*Step 2.* After initialization, the population is evaluated and is subjected to repeated cycles of the search processes of the employed bees, the onlooker bees, and scout bees. Employed bees search for new food sources ( $V_m$ ) having more nectar within the neighborhood of the food source ( $X_m$ ) in their memory. They find a neighbor food source and then evaluate its profitability (fitness). They can determine a neighbor food source  $V_m$  using the formula given by

$$v_{mi} = x_{mi} + \theta_{mi} (x_{mi} - x_{ki}), \quad (12)$$

where  $x_k$  is a randomly selected food source,  $i$  is a randomly chosen parameter index, and  $\theta_{mi}$  is random number between  $[-1, 1]$ .

*Step 3.* After producing the new food source  $V_m$ , its fitness is calculated and a greedy selection is applied between  $V_m$  and  $X_m$ . If the fitness of the new food source position is better than the old food source position, the employed bee moves from the old food source position to the new food source position, and the new food source takes place of old one in the population and becomes a new member.

The fitness value of the solution,  $fit_m(X_m)$ , might be calculated for minimization problems using the following formula:

$$fit_m(X_m) = \begin{cases} \frac{1}{1 + f_m(X_m)}, & \text{if } f_m(X_m) \geq 0, \\ 1 + \text{abs}(f_m(X_m)), & \text{if } f_m(X_m) \leq 0, \end{cases} \quad (13)$$

where  $f_m(X_m)$  is the objective function value of solution  $X_m$ .

*Step 4.* Unemployed bees consist of two groups of bees: onlooker bees and scouts. Employed bees share their food source information with onlooker bees waiting in the hive, and then onlooker bees probabilistically choose their food sources depending on the probability values calculated using the fitness values provided by employed bees.

Calculate the probability values  $P_i$  for the solutions using fitness of the solutions by

$$P_i = \frac{fit_m(X_m)}{\sum_{m=1}^{SN} fit_m(X_m)}, \quad (14)$$

where SN represents the number of the food sources position.

After a food source  $X_m$  for an onlooker bee is probabilistically chosen, a neighborhood source  $V_m$  is determined by Step 2, and its fitness value is computed. As in Step 3, a greedy selection is applied between  $V_m$  and  $X_m$ . Hence, more onlookers are recruited to richer sources, and positive feedback behavior appears.

*Step 5.* If a predetermined number of trials cannot further improve a candidate solution represented by a food source then that food source, is considered abandoned and the employed bee associated with that food source becomes a

scout. The new solution discovered by the scout can be defined by

$$x_{mi} = l_i + \text{rand}(0, 1) * (u_i - l_i), \quad (15)$$

where  $l_i$  and  $u_i$  are the lower and upper bounds of the parameters  $x_{mi}$ . The abandoned food source is replaced by the randomly new food source.

*Step 6.* If a termination condition is reached, the process is stopped and memorizes the best solution achieved so far. Otherwise the algorithm returns to Step 2.

The general algorithmic structure of ABC optimization approach can be summarized as follows [23]:

(i) *Initialization Phase*

(ii) *Repeat*

Employed Bees Phase

Onlooker Bees Phase

Scout Bees Phase

Memorize the best solution achieved so far

(iii) *Until* (Cycle = Maximum Cycle Number).

*3.2. The Discretization of Behavior Models.* The mathematical model of OMD is considered as a combinatorial optimization problem, and ABC has a strong global searching capability to solve this problem. The optimization function for OMD is shown as (6), which is a discrete optimization function; for this reason, the behavior models of ABC should be discretized.

The discretization is defined as follows.

- (1) The expression of artificial bee's state. In this solution space, the state of each artificial bee is encoded by  $-1$  or  $+1$ . If there are  $K$  active users in this DS-UWB MA system, thus the state is a  $K$ -dimensional vector, like  $X_0 = (x_1, x_2, \dots, x_K)^T$ , where  $x_i \in \{-1, +1\}$  and  $i = 1, 2, \dots, K$ .
- (2) The food quality or the fitness function for artificial bees is the criterion of OMD in (6).
- (3)  $X_c$  is the new food sources for each step; if the element of  $X_c$  is  $x_{ci} > 0$ , then assign  $x_{ci} = +1$ ; otherwise,  $x_{ci} = -1$ .
- (4) Initialization. The initial state of each artificial bee is selected randomly in the discrete space with  $2^K$  likely solutions. But the ABC is an iterative optimization algorithm, which means that the selection of initial states has a great effect on the iteration times and the convergence rate. In this paper, we choose a sub-optimal multiuser detection based on code mapping (SCM) and its variations as initial state values of artificial bees, which can ensure the artificial bees gather near the optimal value of the optimization problem because the suboptimal solution is quite close to the optimal value in the solution space. Besides, this selection method of initial states can speed up the convergence rate and reduce the iteration times for optimization.

3.3. *The Specific Steps of the Initial Value Selection.* The specific steps of the initial value selection are given as follows.

*Step 1.* Execute matched filter detection and code mapping to get a suboptimal solution first. Assign the result  $b_{1 \times K} = (b_1, b_2, \dots, b_K)$  to the first artificial bee as its initial state value, where  $b_i \in \{-1, +1\}$  and  $i = 1, 2, \dots, K$ .

*Step 2.* Change an element  $b_i$  of the result  $b_{1 \times K}$  randomly,  $b_i^* = -b_i$ , and then assign this new result  $b_{1 \times K}^*$  which changes from  $b_{1 \times K}$  to the second artificial bee as its initial state value.

*Step 3.* Repeat Step 2 and assign these new results to other artificial bees.

*Step 4.* If all artificial bees have been assigned initial state values, the procedure of initial value selection comes to the end. And we can start the ABC to research the optimal value for MUD. In consideration of the previous statements, the overall structure of this proposed SCM-ABC-MUD detector is shown in Figure 2.

Figure 2 shows the block diagram of the algorithm. The implementation of this detector can be summarized as follows: firstly, the output of a bank of matched filter receivers is fed to suboptimal detectors based code mapping and clustering; secondly, the detection results of these suboptimal detectors are used to construct an initial solution space; finally, the ABC is executed in this space.

## 4. Simulation and Discussion

In order to analyze the proposed SCM-ABC-MUD algorithm, Monte Carlo simulations are utilized and the values of these majority parameters are assigned as Table 1.

4.1. *The BER Performance versus SNR.* The BER versus SNR curves with perfect power control in the AWNG are depicted in Figure 3. And there are 10 users in the system. As for ABC, we fix the number of initializing food sources (equal to employed bees) that is 3; limit, the value of predetermined number of cycles, that is 3; and MCN, the maximum cycle number, that is 20.

The performances of matched filter (MF), decor relating (DEC) [24], minimum mean square error (MMSE) [25], SCM-ABC-MUD, and OMD detectors are compared and depicted in Figure 3.

Figure 3 shows that the BER performance of SCM-ABC-MUD is superior to those of other suboptimal detectors including MF, DEC, and MMSE, and it even coincides with that of OMD. The reason is that this proposed SCM-ABC-MUD algorithm can make a search within a simplified solution space constructed by the solutions of these suboptimal detectors rather than a random search.

4.2. *The BER Performance versus User Numbers  $K$ .* The BER performances of these detectors with different number of active users  $K$  are analyzed in this experiment. The SNR of the AWGN channel is 5 dB. And the control parameters of ABC

TABLE 1: Simulation parameters.

System	DS-UWB
Modulation mode	BPSK
Spreading codes (SC)	m sequences
The length of SC	255
Communication channel	AWGN
The number of testing information bits	3200000
The width of UWB pulse	0.8 ns
The pulse repetition period	$\approx 2$ ns
Limit	3
Initializing food source number	3

are same as the BER performance versus SNR. Figure 4 shows the result of this experiment. According to Figure 4, the performance of SCM-ABC-MUD is the best of the suboptimum MUD except OMD. While the number of users increases, ABC needs a new group of control parameters. However, in this experiment, they are unchanged, resulting in the performance difference between SCM-ABC-MUD and OMD.

4.3. *The NFE Resistant Capability Comparison.* In this experiment, the SNR of the first user's received signal is assigned at 5 dB all the time, while the SNR of other users changes from 0 dB to 10 dB with per step of 1 dB. Besides, other parameters are set as same as those in the BER performance versus SNR. Figure 5 shows the first user's BER performances of different MUD detectors. It can be seen from Figure 5 that OMD has the best near-far effect resistant ability, and the SCM-ABC-MUD is very close to OMD.

4.4. *BER Performance versus Different Initial States.* As an iterative optimization algorithm, the convergence rate is as important as the optimization capability. And the selection of initial states has a great effect on the iteration times and the convergence rate. In this experiment two initial state values of artificial bees are employed as a comparison. One has random initial state values which is named as ABC-MUD; the other is SCM-ABC-MUD which is assigned a group of initial state values using the suboptimal solution of the code mapping-based multiuser detection. The BER performance is studied for both algorithms with the maximum cycle number (MCN) of 5, 15, and 20, respectively. Figures 6 and 7 are the convergence rates of ABC-MUD and SCM-ABC-MUD separately.

From Figures 6 and 7, the BER performance of SCM-ABC-MUD is much better than ABC-MUD with the same iteration times, and the former can achieve OMD's BER performance with 20 times iteration which is the global optimal solution. It can be concluded that the convergence rate of SCM-ABC-MUD is much quicker than that of ABC-MUD. Meanwhile, ABC-MUD only improves its BER performance a little than matched filter (shown in Figure 3). It is because that SCM-ABC-MUD makes full use of the suboptimal solution, the initial value of SCM-ABC-MUD is more close to OMD

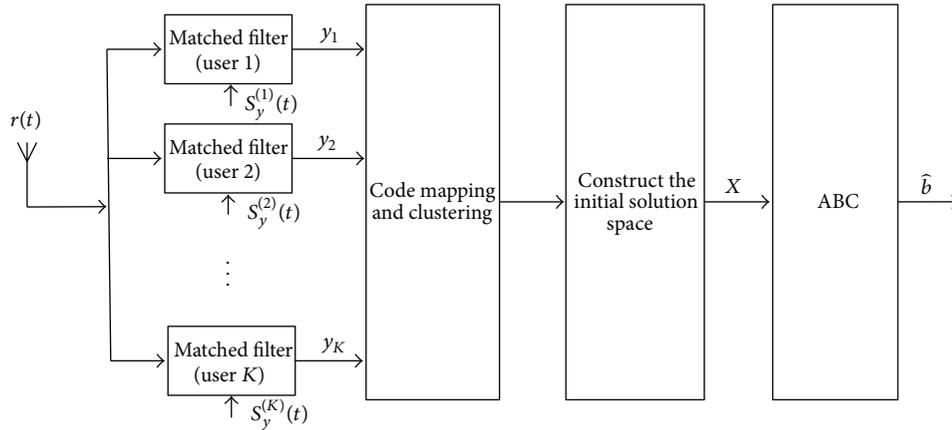


FIGURE 2: General schematic diagram of the SCM-ABC-MUD detector.

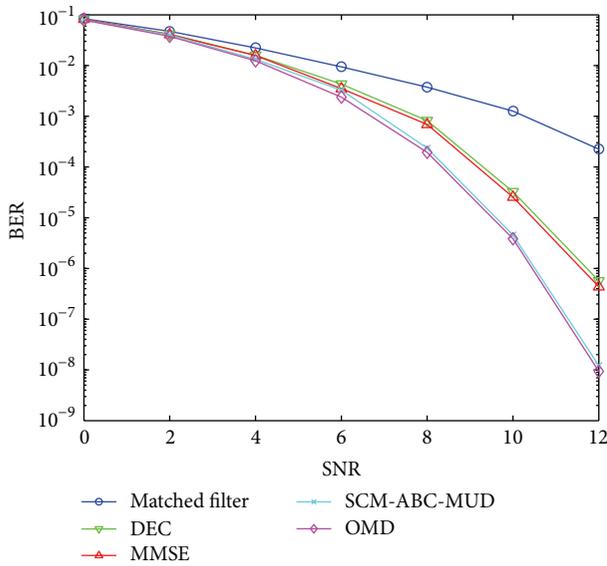
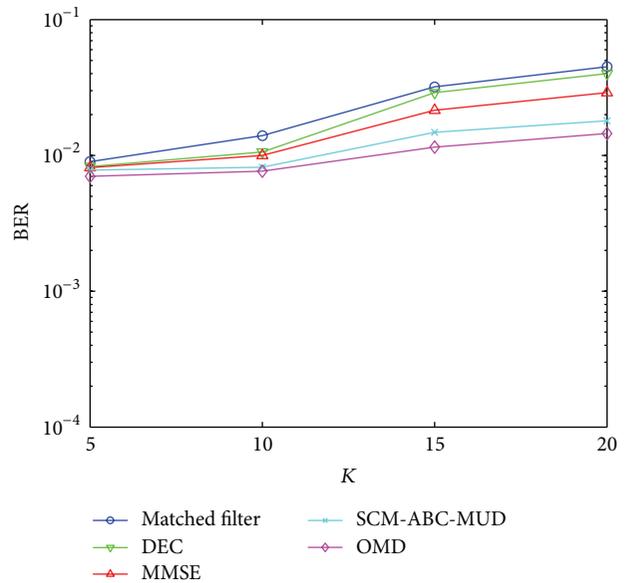


FIGURE 3: BER performance versus SNR.

FIGURE 4: BER performance versus user numbers  $K$ .

than ABC-MUD, and so it can achieve the optimal value of OMD much quicker than ABC-MUD.

**4.5. The Computational Complexity of the Proposed SCM-ABC-MUD.** OMD can obtain the best BER performance while the computational complexity performance is too expensive to be implemented in practice. However, the computational complexity of SCM-ABC-MUD is much less than that of the OMD. In a  $K$ -user DS-UWB system, to detect the  $K$ -user vector  $\mathbf{b}$ , the computational complexity of iterations of the OMD is  $2^K$ . The computational complexity of the proposed method includes three parts. The first part is the computational complexity of the code mapping; according to Section 2.2.3, the mapping method is just a calculation of  $L(\mathbf{b})$ , so the computational complexity of the first part can be neglected. The second part is the computational complexity of C-means clustering; the complexity of iterations of the code

clustering is  $K$ . The third part is the complexity of the ABC algorithm. It depends on the convergence rate of SCM-ABC-MUD. In Section 4.4, we have obtained a conclusion that the SCM-ABC-MUD can achieve convergence with a much fewer iterations than OMD. Let the normalized operation time of per vector  $\mathbf{b}$  using matched filter detector equal to 1 in the condition of 10 users. The simulation parameters are the same as Section 4.1. Table 2 lists the relative operation time using OMD, SCM-ABC-MUD, and matched filter.

From Table 2 we can see that the computational complexity of the SCM-ABC-MUD is in the same order of magnitude to the MMSE and DEC and far lower than OMD. This is because the iteration of SCM-ABC-MUD will be converged very soon and costs little quantity of computation. Hence, we can get a conclusion that the SCM-ABC-MUD can get good BER performance with low computational complexity.

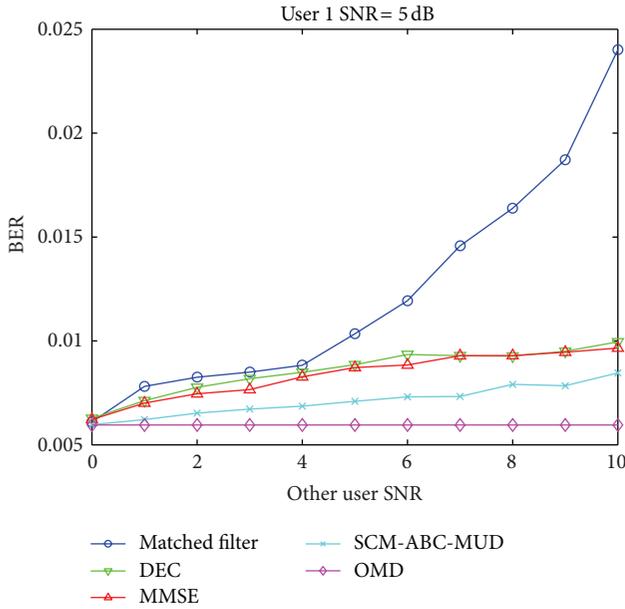


FIGURE 5: Near-far effect resistant of different MUD algorithms.

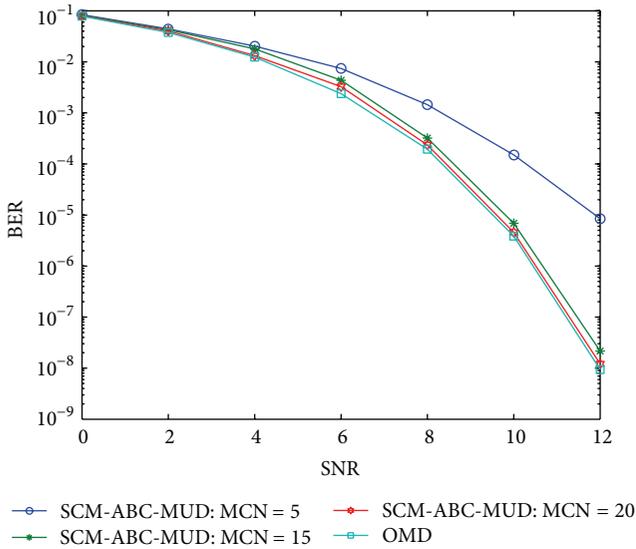


FIGURE 6: The BER performance with different iteration times of SCM-ABC-MUD.

### 5. Conclusions

In this paper, we firstly employed the Artificial Bee Colony algorithm in the DS-UWB MUD. In consideration of the high computational complexity of OMD, the proposed MUD is a hybrid method which combines ABC algorithm and a suboptimal solution of the code mapping-based MUD. First, the bits set output from the matched filters is mapped into a one-dimensional feature space to obtain a suboptimal solution; then the initial solution space is constructed based on the suboptimal solution; finally, the optimal solution is found by operating the different behaviors of artificial bees in solution space. The proposed multiuser detector can make full use

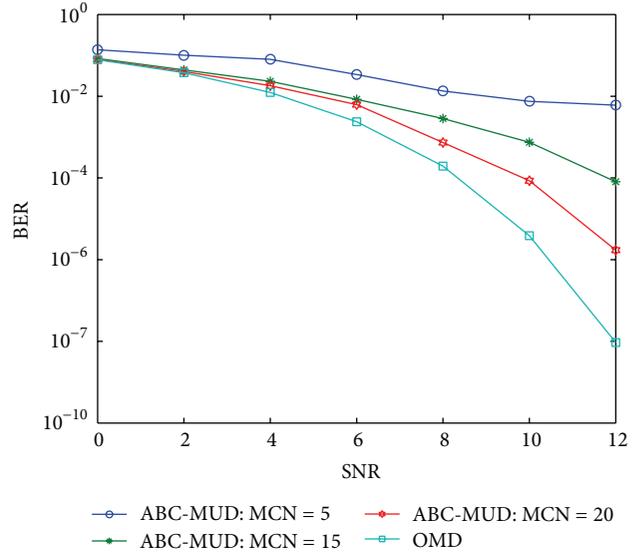


FIGURE 7: The BER performance with different iteration times of ABC-MUD.

TABLE 2: The comparison of computational complexity using different MUD algorithms.

OMD	MMSE	DEC	SCM-ABC-MUD	MF
58.9	1.38	1.30	1.21	1

of the suboptimal solution and advantages of ABC to study the optimal value in the solution space. Simulation results have indicated that the BER performance, user capacity, and the NFE resistant ability of this novel algorithm are quite close to those of OMD, and they are also superior to those of MF, DEC, and MMSE. Furthermore, the convergence rate of SCM-ABC-MUD is better than that of ABC-MUD. And the computational complexity of the SCM-ABC-MUD is much lower than that of OMD.

### Acknowledgments

The research in this paper is supported by the National Natural Science Foundation of China (Grant no. 61102084), Foundation of China Academy of Space Technology (CAST), and the China Postdoctoral Science Foundation (Grant no. 2011M500665).

### References

- [1] M. Z. Win and R. A. Scholtz, "Impulse radio: how it works," *IEEE Communications Letters*, vol. 2, no. 2, pp. 36–38, 1998.
- [2] X. Shen, M. Guizani, H. H. Chen et al., "Guest editorial ultra-wideband wireless communications-theory and applications," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 4, pp. 713–716, 2006.
- [3] J. Ahmadi-Shokouh and R. C. Qiu, "Ultra-wideband (UWB) communications channel measurements—a tutorial review," *International Journal of Ultra Wideband Communications and Systems*, vol. 1, no. 1, pp. 11–31, 2009.

- [4] I. Oppermann, M. Hämäläinen, and J. Iinatti, Eds., *UWB: Theory and Applications*, Wiley, 2005.
- [5] S. Verdu, "Minimum probability of error for asynchronous Gaussian multiple-access channels," *IEEE Transactions on Information Theory*, vol. 32, no. 1, pp. 85–96, 1986.
- [6] C. Wang, M. Ma, R. Ying et al., "Narrowband interference mitigation in DS-UWB systems," *IEEE Signal Processing Letters*, vol. 17, no. 5, pp. 429–432, 2010.
- [7] P. Kaligineedi and V. K. Bhargava, "Frequency-domain turbo equalization and multiuser detection for DS-UWB systems," *IEEE Transactions on Wireless Communications*, vol. 7, no. 9, pp. 3280–3284, 2008.
- [8] G. S. Biradar, S. N. Merchant, and U. B. Desai, "Performance of constrained blind adaptive DS-CDMA UWB multiuser detector in multipath channel with narrowband interference," in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM '08)*, pp. 1–5, November–December 2008.
- [9] Q. Z. Ahmed and L. L. Yang, "Reduced-rank adaptive multiuser detection in hybrid direct-sequence time-hopping ultrawide bandwidth systems," *IEEE Transactions on Wireless Communications*, vol. 9, no. 1, pp. 156–167, 2010.
- [10] M. Woodside-Oriakhi, C. Lucas, and J. E. Beasley, "Heuristic algorithms for the cardinality constrained efficient frontier," *European Journal of Operational Research*, vol. 213, no. 3, pp. 538–550, 2011.
- [11] B. Basturk and D. Karaboga, "An artificial bee colony (ABC) algorithm for numeric function optimization," in *Proceedings of IEEE Swarm Intelligence Symposium*, pp. 12–14, May 2006.
- [12] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," *Journal of Global Optimization*, vol. 39, no. 3, pp. 459–471, 2007.
- [13] W. Zou, Y. Zhu, H. Chen et al., "A clustering approach using cooperative artificial bee colony algorithm," *Discrete Dynamics in Nature and Society*, vol. 2010, Article ID 459796, 16 pages, 2010.
- [14] Y. Xu, P. Fan, and L. Yuan, "A simple and efficient artificial bee colony algorithm," *Mathematical Problems in Engineering*, vol. 2013, Article ID 526315, 9 pages, 2013.
- [15] D. Karaboga and B. Akay, "A comparative study of artificial bee colony algorithm," *Applied Mathematics and Computation*, vol. 214, no. 1, pp. 108–132, 2009.
- [16] D. Karaboga, B. Akay, and C. Ozturk, "Artificial bee colony (ABC) optimization algorithm for training feed-forward neural networks," in *Modeling Decisions for Artificial Intelligence*, pp. 318–329, Springer, Berlin, Germany, 2007.
- [17] L. P. Wong, M. Y. H. Low, and C. S. Chong, "A bee colony optimization algorithm for traveling salesman problem," in *Proceedings of the 2nd Asia International Conference on Modelling and Simulation (AICMS '08)*, pp. 818–823, IEEE, May 2008.
- [18] F. Kang, J. Li, and Q. Xu, "Structural inverse analysis by hybrid simplex artificial bee colony algorithms," *Computers and Structures*, vol. 87, no. 13–14, pp. 861–870, 2009.
- [19] D. Karaboga and B. Akay, "A modified artificial bee colony (ABC) algorithm for constrained optimization problems," *Applied Soft Computing Journal*, vol. 11, no. 3, pp. 3021–3031, 2011.
- [20] B. Akay and D. Karaboga, "A modified artificial bee colony algorithm for real-parameter optimization," *Information Sciences*, vol. 192, pp. 120–142, 2012.
- [21] B. Alatas, "Chaotic bee colony algorithms for global numerical optimization," *Expert Systems with Applications*, vol. 37, no. 8, pp. 5682–5687, 2010.
- [22] C. Xu, P. Zhang, B. Li et al., "Vague C-means clustering algorithm," *Pattern Recognition Letters*, vol. 34, no. 5, pp. 505–510, 2012.
- [23] D. Karaboga, "Artificial bee colony algorithm," *Scholarpedia*, vol. 5, no. 3, p. 6915, 2010.
- [24] S. Im and E. J. Powers, "An iterative decorrelating receiver for DS-UWB multiple access systems using biphasic modulation," in *Proceedings of IEEE Workshop on Signal Processing Systems Design and Implementation*, pp. 59–64, October 2004.
- [25] G. C. Chung, "Multi-user access interference suppression for UWB system," in *Proceedings of the International Conference on Computer & Information Science (ICCIS '12)*, vol. 2, pp. 670–675, June 2012.



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

