Progress In Electromagnetics Research C, Vol. 94, 45–57, 2019

Two-Way Pattern Synthesis of MIMO Radar with Sidelobe Reduction and Null Control via Improved Whale Optimization Algorithm

Pengliang Yuan*, Chenjiang Guo, Guofeng Jiang, and Qi Zheng

Abstract—This paper proposes an improvement to the whale optimization algorithm (WOA), which is based on the Levy flight technique. The improvement allows improved whale optimization algorithm (IWOA) to have a good diversity of population, faster convergence and overcome premature. The test by using different benchmark functions is conducted to demonstrate the effectiveness of improvement on the algorithm performance. Finally, IWO algorithm is applied to optimize the problems of two-way pattern of MIMO radar system, involving the achievements of sidelobe reduction, deep null, and wide null at the prescribed directions. The obtained numerical results demonstrate that IWOA can not only efficiently fulfill the expected deep nulls and wide nulls at the prescribed directions, but also enable the peak sidelobe level to retain in the smaller level at the same time than several state-of-the-art algorithms.

1. INTRODUCTION

In recent years, MIMO radar has received much attention due to superior characteristics [1–12]. Unlike a traditional phased array system, MIMO radar allows emitting orthogonal or noncoherent waveforms at transmitter, and these waveforms can be extracted at receiver by a set of matched filters. In the case of emitting orthogonal waveforms, a virtual array can be formed by exploiting the MIMO concept. Due to the feature of virtual array, MIMO radar has many superior characteristics, such as good detective ability, high resolution, and accurate localization.

However, interference is always an inevitable problem in practice= and would cause the distortion of signals, ultimately degrade the performance of entire system. Generally in terms of antenna array, there are two approaches available to overcome this problem: one is to suppress the peak sidelobe level (Psll) [3, 13, 14], and the other is to form the deep nulls at the directions of interferences in the radiation pattern [13]. Herein we only focus on the two-way pattern [2] of MIMO radar.

Presently, some evolutionary algorithms have been used to solve the problems of sidelobe suppression, deep null, and capacity of system associated with the array configuration, such as generic algorithm (GA) [2, 15], differential particle swarm optimization (DPSO)) [3], galaxy-based search algorithm [16], chaotic differential evolution(DE) [17], and chaotic whale optimization algorithm(CWOA) [18]. For the optimization of system capacity, some evolutionary algorithms have also been utilized to improve the capacity of system by optimizing the array configuration [15, 16, 19, 20]. Ref. [19] uses GA to optimize the element spacing to achieve the optimal capacity of system. In [15], a hybrid algorithm, integrating with GA and Taguchi algorithm, is proposed to improve the capacity of a MIMO wireless system for the linear array with the different geometrical configurations, hence the

Received 15 April 2019, Accepted 5 June 2019, Scheduled 12 July 2019
* Corresponding author: Pengliang Yuan (yuanpengliang@mail.nwpu.edu.cn).
The authors are with the School of Electronics and Information, Northwestern Polytechnical University, Xi'an, Shannxi 710072, China.

rectangular array is taken in further consideration in [16]. For the optimization of sidelobe suppression and deep null, generic algorithm (GA) is adopted to suppress the Psll of pattern by optimizing the positions of elements in the literature [2]. This method obtains a desired Psll, but suffers from a heavier computational cost. On the basis of [2], a hybrid algorithm called differential particle swarm optimization (DPSO) algorithm is proposed in [3], in which the crossover and mutual operator from the differential evolution algorithm are integrated into particle swarm optimization scheme, to use the superiority of DE operators to improve the diversity of population and avoid premature of PSO. Its experimental results demonstrate that the ability of global optimization is greatly improved, and the diversity of population is enhanced. In [17], chaotic optimization is introduced into DE scheme to enhance the diversity of population of DE. The basic idea of this algorithm is to make full use of the diversity property of chaotic optimization to reduce the risk of trapping into local optimality, further to avoid the premature of algorithm. Due to the emerging of WOA, the CWOA integrating the chaotic optimization with WOA is used to optimize the problem of sidelobe suppression [18].

In 2016, [21] proposes WOA inspired by the hunting behavior of humpback whales. The WOA is very competitive, in comparison with several state-of-the-art meta-heuristic algorithms [21]. So far, WOA has been applied in general array synthesis [22]. However, each evolutionary algorithm has its own advantage and disadvantage, and WOA directly applied in the synthesis problem of MIMO radar is not always effective. Therefore, we propose an improved whale optimization algorithm called IWOA, which is applied to solve the optimization problem of MIMO radar, involving the sidelobe reduction, null control, and wide null. Since we need to consider the two arrays simultaneously for synthesis the problem of MIMO radar, the synthesis of wide null in MIMO radar using evolutionary algorithms is thus a challenge problem.

This paper is organized as follows. Section 2 formulates the specific problem mathematically. The description and performance evaluation of the proposed algorithm are presented in Section 3. The simulation results are shown in the Section 4. Section 5 contains our conclusion.

2. PROBLEM FORMULATION

Considering a narrow-band MIMO radar system, the transmitter consists of M antennas, and the receiver consists of N antennas. These antennas are all linear omnidirectional antennas. Then the direction functions of transmitter and receiver, $f_T(u)$ and $f_R(u)$, can be stated as follows

$$f_{\mathrm{T}}(u) = \left|\sum_{i=1}^{M} w_i^{\mathrm{T}} \exp\left(j2\pi d_i^{\mathrm{T}} u/\lambda\right)\right| \tag{1}$$

$$f_{\mathrm{R}}(u) = \left|\sum_{i=1}^{N} w_i^{\mathrm{R}} \exp\left(j2\pi d_i^{\mathrm{R}} u/\lambda\right)\right| \tag{2}$$

where $j = \sqrt{-1}$, w_i^{T}, and d_i^{T} are the ith excitation and location of transmitter subarray; w_i^{R} denotes the ith excitation of receiver subarray; d_i^{R} stands for the ith location of receiver subarray; $\{\cdot\}^T$ is the transpose of a vector; λ is the operating wavelength. Let $u = \sin\theta - \sin\theta_0$, where θ is the beam direction of plane wave, uniformly divided by k, i.e., $\theta = \{\theta_i | i = 1, \ldots, k\}$, and θ_0 is the direction of maximum beam. For the linear array, θ and θ_0 are limited in $[-\pi/2, \pi/2]$, therefore the variable $u \in [-2, 2]$. In the transmitter, assuming that the maximum steering of main lobe is located at $u = 0$, then $f_{\mathrm{T}}(u) = f_{\mathrm{T}}(-u)$. Since the pattern in transmitter is symmetric about $u = 1$, we have $f_{\mathrm{T}}(1 + \Delta u) = f_{\mathrm{T}}(1 - \Delta u)$. Similar to the transmitter, the receiver has $f_{\mathrm{R}}(u) = f_{\mathrm{R}}(-u)$, $f_{\mathrm{R}}(1 + \Delta u) = f_{\mathrm{R}}(1 - \Delta u)$. The two-way pattern of MIMO radar [3, 7] associated with u can be written as

$$f(u) = f_{\mathrm{T}}(u) \cdot f_{\mathrm{R}}(u) = \left|\sum_{i=1}^{M}\sum_{i=1}^{N} w_i^{\mathrm{T}} w_i^{\mathrm{R}} \exp\left(j2\pi\left(d_i^{\mathrm{T}} + d_i^{\mathrm{R}}\right) u/\lambda\right)\right| \tag{3}$$

To compute more efficiently in Matlab software, we adopt Kronecker operator $\otimes$ to rewrite Eq. (3) as follows

$$f(u) = \left|\sum_{i=1}^{M} w_i^{\mathrm{T}} \exp\left(j2\pi d_i^{\mathrm{T}} u/\lambda\right)\right| \otimes \left|\sum_{i=1}^{N} w_i^{\mathrm{R}} \exp\left(j2\pi d_i^{\mathrm{R}} u/\lambda\right)\right| = f_{\mathrm{T}}(u) \otimes f_{\mathrm{R}}(u) \tag{4}$$

This is because the computation load of using Eq. (4) to compute is much less than that of using Eq. (3), which has been proved in [3].

3. IMPROVED WHALE OPTIMIZATION ALGORITHM

The main steps of IWOA algorithm are summarized in Algorithm 1, and the corresponding parameters are specified in Table 1. In the following, we only complete some details as supplement to Algorithm 1.

Algorithm 1: Pseudo code of the IWOA algorithm

Input: input parameters $t, t_{\max}$, l, r
Output: X^*
Initialize the whales population $X_i(i = 1, 2, \ldots, n)$
Calculate the fitness of each search agent
X^* =the best search agent
while $t < t_{\max}$ **do**
  for *each search agent* **do**
    Update α, A, C, l and p
    if $p <$ th **then**
      if $|A| \leq 1$ **then**
        Update the position of the current search agent by $D = |C \cdot x^*(t) - X(t)|$
        $X(t+1) = X^*(t) - A \cdot D$
      else
        Update the position of the current search agent by the (5)
      end
    else
      Update the position of the current search agent by $D^{'} = |X^*(t) - X(t)|$
      $X(t+1) = D^{'} \cdot e^{bl} + \cos(2\pi l) + X(t)$
    end
  end
  Correct the search agent beyond the search space
  Calculate the fitness of each search agent
  Update X^* if there is a better solution
  $t = t + 1$
end

At the beginning of initialization, we select a known solution as an initial population, since a better initial population will contribute to fast convergence of algorithm [23]. On the other hand, it will also relieve the sensitivity of the evolutionary algorithm to an initial population.

Compared with some traditional algorithms, WOA has demonstrated to have good performance; however, it has some demerits, such as slow convergence, low precision, and being easily trapped into local optimum because of insufficient diversity of population [24]. Levy flight is an effective method to implement the large jump in the search space and is used in many optimization techniques to produce random step size in the design region [25–27]. Moreover, using Levy flight can improve the fast moving ability of WOA, pull WOA to jump out of the stagnation on the problem, and reduce the number of spiral paths or number of iterations. Thus, it can get better various search spaces for detecting the global optimal. Herein the update of position vector for search agent is based on the Sine Cosine Algorithm (SCA), proposed by Mirjalili in 2016 [28]. SCA makes full use of the periodic behavior of sine and cosine trigonometric functions to generate a new candidate solution around the other, so as to enhance the capability of exploitation.

To obtain better diversity, Levy flight distribution [29] is used and defined by the following:

$$X_i(t+1) = \text{Levy}(X_i(t)) + \alpha \cdot \sin(\tau) \left|\gamma X^* - X_i(t)\right| \tag{5}$$

Table 1. Parameters specification.

	Symbol	Quantity	Value
(5)	α	control parameter	$2(1-t/t_{\max})$
	τ	random number	$[0, 2\pi]$
	γ	random number	$[0, 2]$
(8)	*scale*	scale factor	0.01
(9)	β	control parameter	1.5
Algorithm1	t	iteration times	$[1, t_{\max}]$
	$t_{\max}$	maximum of iteration	100
	l	random number	$(-1, 1)$
	r	random number	[0,1]
	b	random number	$(0, 10)$
	C	control parameter	$2 \cdot r$
	α	control parameter	$2(1-t/t_{\max})$
	A	control parameter	$2\alpha \cdot r - \alpha$
	th	threshold of probability	0.5
	p	random number	$[0, 1]$

where X_i is the position vector of the ith individual; α, τ and γ are the parameters to obtain the position of the new solution, the specifications of α, τ, and γ are listed in Table 1. In Eq. (5), Levy$(X_i(t))$ can be defined by the following:

$$\text{Levy}(X_i(t)) = X_i(t) + step \oplus \text{random}(\text{size}(X_i(t))) \tag{6}$$

where random(size$(X_i(t))$) is the dimension of vector and equals the dimension of the objective problem. The term *step* is obtained by the following:

$$step = stepsize \oplus X_i(t) \tag{7}$$

where parameter *stepsize* is given by the following:

$$stepsize = scale \times s \tag{8}$$

where *scale* is the scale factor, $s = \frac{u}{v^{1/\beta}}$, where u and v are Gaussian distributions, respectively, i.e., $u \sim N(0, \sigma_u^2), v \sim N(0, \sigma_v^2)$, where $\sigma_v^2 = 1$, σ_u^2 is defined by the following:

$$\sigma_u^2 = \left\{ \frac{\Gamma(1+\beta)\sin(\pi\beta/2)}{\Gamma((1+\beta)/2)\beta 2^{(\beta-1)/2}} \right\}^{1/\beta}, \text{where} \quad \beta \in (0, 2] \tag{9}$$

where Γ stands for the standard Gamma function.

In Algorithm 1, the choice of b mainly depends on the feature of problem. Generally, b will be limited between 0 and 10. We choose $b = 4$ according to our observation in the experiments. Also, IWOA will dramatically worsen in performance if b is more than 8. Additionally, the threshold th can be tuned, but there is only a slight effect on the performance of IWOA. th equals 0.5. We still take $th = 0.5$ [21] in our experiments.

The improved algorithm IWOA is tested by solving the eight benchmark functions F_1-F_8, listed in Table 2. Such functions are used in [21, 30–32]. Table 2 also gives the number of design variables V_{no}, dynamic range of optimization variables *Range*, and the optimum $f_{\min}$.

For the sake of comparison, the population size and maximum iteration are consistent with those of [21], i.e., population size = 30, $t_{\max} = 500$. Functions F_1–F_4 have only one global optimum to evaluate the capability of exploitation for meta-heuristic algorithm. F_5–F_8 are multi-modal functions including many local optima, whose numbers increase exponentially with the number of design variables. Functions F_5–F_8 can evaluate the capability of exploration for meta-heuristic algorithm.

Table 2. Table of testbench functions.

Function	V_{no}	Range	$f_{\min}$
$F_1(x) = \sum_{i=1}^{n} x_i^2$	30	$[-100, 100]$	0
$F_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	30	$[-10, 10]$	0
$F_3(x) = \sum_{i=1}^{n} \left(\sum_{j=1}^{i} x_j \right)^2$	30	$[-100, 100]$	0
$F_4(x) = \max_i \{ \lvert x_i \rvert, 1 \leq i \leq n \}$	30	$[-100, 100]$	0
$F_5(x) = \sum_{i=1}^{n} \left[x_i^2 - 10\cos(2\pi x_i) + 10 \right]$	30	$[-5.12, 5.12]$	0
$F_6(x) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2} \right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n} \cos(2\pi x_i) \right) + 20 + e$	30	$[-32, 32]$	0
$F_7(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}} \right) + 1$	30	$[-600, 600]$	0
$F_8(x) = \frac{\pi}{n}\left\{ 10\sin(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 [1 + 10\sin^2(\pi y_{i+1}] + (y_n - 1)^2 \right\}$ $+ \sum_{i=1}^{n} u(x_i, 10, 100, 4)\}$			
$y_i = 1 + \frac{x_i + 1}{4}, u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a \\ 0, & -a < x_i < a \\ k(-x_i - a)^m, & x_i < -a \end{cases}$	30	$[-50, 50]$	0

For each benchmark function, starting from different populations randomly generated, IWOA will run 100 times,

The corresponding results are shown in Table 3. To facilitate comparison, we also provide the statistical results of WOA, PSO, Gravitational Search Algorithm (GSA) [33], DE, and Fast Evolutionary Programming (FEP) [34]. As can be seen, IWOA is the most efficient algorithm or the second best one, and exhibits a better performance than other algorithms. IWOA has the ability of good exploitation and exploration.

Table 3. Average and standard deviation (Std) for F_1-F_8 benchmark functions with different algorithms.

	Algorithms	F1	F2	F3	F4	F5	F6	F7	F8
Average	IWOA	**2.76E-82**	**1.51E-50**	**6.49E-11**	7.00E-02	**0**	**2.07E-15**	**0**	**8.2425E-18**
	WOA	1.41E-30	1.06E-21	5.39E-07	0.072581	**0**	7.4043	0.000289	0.339676
	PSO	0.000136	0.042144	7.01E+01	1.086481	46.70423	0.276015	0.009215	0.006917
	GSA	2.53E-16	0.055655	8.97E+02	7.35487	25.96841	0.062087	27.70154	1.799617
	DE	8.20E-14	1.50E-09	6.80E-11	**0**	69.2	9.70E-08	**0**	7.9E-15
	FEP	5.70E-04	8.10E-03	0.016	0.3	0.046	0.018	0.016	9.2E-06
Std	IWOA	**3.89E-82**	**4.52E-50**	**7.27E-11**	8.43E-02	**0**	**1.67E-15**	**0**	**6.25E-18**
	WOA	4.91E-30	2.39E-21	2.93E-06	0.39747	**0**	9.897572	0.001586	0.214864
	PSO	0.000202	4.54E-02	2.21E+01	0.317039	11.62938	0.50901	0.007724	0.026301
	GSA	9.67E-17	0.194074	3.19E+02	1.741452	7.470068	0.23628	5.040343	0.95114
	DE	5.90E-14	9.90E-10	7.40E-11	**0**	38.8	4.20E-08	**0**	8E-15
	FEP	0.00013	0.00077	1.40E-02	0.5	0.012	0.0021	0.022	3.6E-06

The convergence curves of IWOA, GSA, PSO, and SSA are provided in Fig. 1 to see the convergence rates of these algorithms. Herein the average of the best solution is attained in each iteration over 30 runs. As can be seen in the figure, IWOA shows different behaviors when optimizing the benchmark functions. Firstly, the convergence rate is accelerated with the increase of iteration. It is evident in F_5, F_6, and F_7. This results from the adaptive mechanism in IWOA that helps to move to the promising regions of the search space and rapid convergence. Secondly, the final iterations all reach better optima, and these results show that the IWOA benefits from the modification and obtains better diversity. In summary, it seems that IWOA has the success convergence rate in optimizing the benchmark functions.

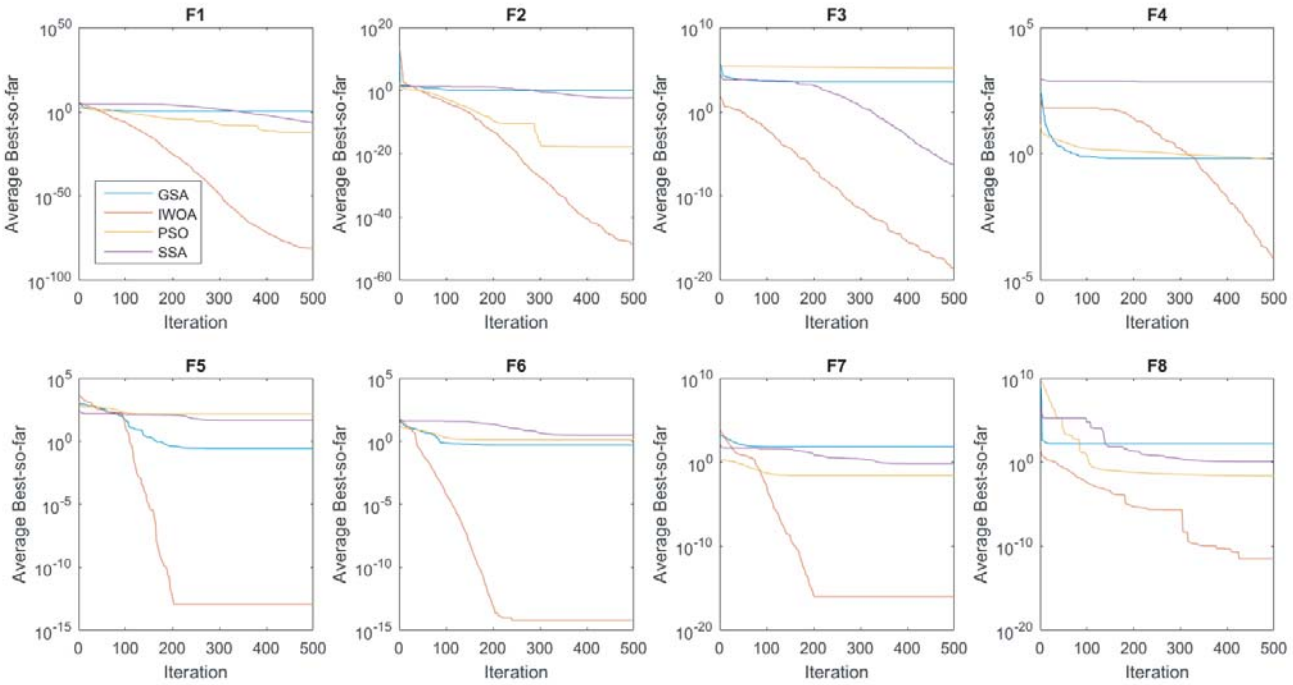

Figure 1. Comparison of convergence curves between IWOA and other algorithms.

4. SIMULATION RESULTS

Herein we present three cases on synthesis of the array problems. The computer configuration and parameters specification used in simulation are listed in Table 4. The parameter configuration is as follows: the elevation angle $\theta \in [-90°, 90°]$ is divided by 1024. One hundred independent runs are repeated for reliability in the next examples. In consideration of the effect of mutual coupling, tthe minimum spacing will be restricted not to be lower than $\lambda/2$.

Table 4. Computer configuration and parameters specification.

Computer configuration			
RAM	2GB DDR		
Language	Matlab 2016		
CPU	2.66 GHz Intel Core2^{TM}		
Parameters specification			
Iteration	Initial population	Aperture (T/R)	No. elements (T/R)
100	60	50λ	25

4.1. Synthesis of Psll with Spacing

This case is to optimize the spacing of both subarrays for the lower Psll on pattern, thus the spacings of transmitter and receiver, $d_i^{\mathrm{T}}, d_i^{\mathrm{R}}$, are viewed as the variables of optimization. This problem is subject to the given excitation, the fixed aperture and the constrained spacing. In summary, it can be formulated

in the form of optimization problem as follows

$$\begin{aligned} \text{find} \quad & \left(d_i^{\mathrm{T}}, d_i^{\mathrm{R}}\right) \\ \min \quad & fitness \\ \text{s.t.} \quad & \ell_{\mathrm{T}} = \ell_{\mathrm{R}} = 50\lambda, \\ & \left|w_i^{\mathrm{T}}\right| = \left|w_i^{\mathrm{R}}\right| = 1,\ d_i^{\mathrm{T}},\ d_i^{\mathrm{R}} \succeq \lambda/2. \end{aligned} \tag{10}$$

where $\ell_{\mathrm{T}}, \ell_{\mathrm{R}}$ respectively denote the apertures of transmitter and receiver; $w_i^{\mathrm{T}}, w_i^{\mathrm{R}}$ are the excitations of both subarrays; and the fitness function $fitness$ is defined by

$$\begin{aligned} fitness &= \sum_{i=1}^{n} \frac{1}{\Delta\theta_i} \int_{\theta_{il}}^{\theta_{iu}} \left|f(\theta_s) - \mathrm{Psll}_d\right|^2 d\theta, \text{where} \\ \Delta\theta_i &= |\theta_{iu} - \theta_{il}|, \theta_s \in \theta_{\text{sidelobe region}} \end{aligned} \tag{11}$$

where θ_{il} and θ_{iu} represent the upper bound and lower bound of the ith angular interval θ_i, and $\mathrm{Psll_d}$ denotes the desired Psll.

The simulation results by IWOA are shown in Table 5 and Fig. 2. Table 5 provides Psll and Psll_{ave} of [2, 3, 7] to compare with that of IWOA. It can be observed that IWOA reaches the best Psll -32.82 dB, which outperforms the other three algorithms. Fig. 2 gives the resultant radiation pattern obtained by IWOA. Here we only compare it with those of GA [2], because the specifically numerical results of DPSO and CDE on patterns are not provided in [3, 7]. As can be clearly seen, the dynamic varying range of IWOA is less than that of GA over the sidelobe region.

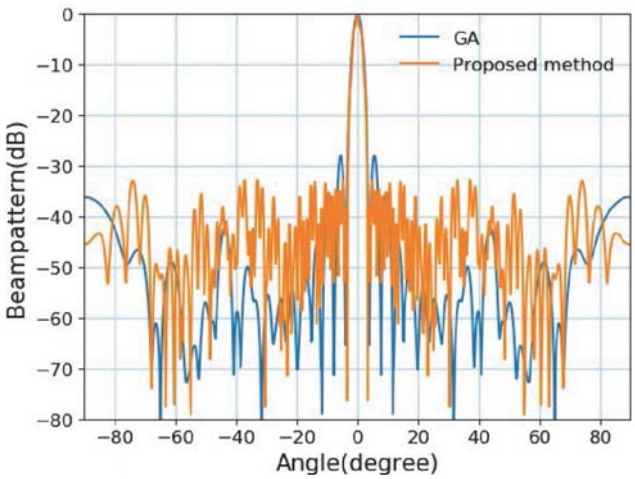

Figure 2. Resultant radiation patterns obtained by GA and IWOA.

Table 5. Performance comparison with different algorithms.

Algorithm	GA [2]	DPSO [3]	CDE [7]	IWOA
Psll (dB)	-28.65	-30.23	-30.93	$\mathbf{-32.82}$
Psll_{ave} (dB)	–	-29.15	–	$\mathbf{-41.26}$

4.2. Synthesis of Psll and Null with Spacing and Excitation

This case is based on the multi-objective optimization problem of MIMO radar, i.e., for the lower Psll and deeper nulls. The excitations and spacings of transmitter and receiver, $d_i^{\mathrm{T}}, d_i^{\mathrm{R}}, w_i^{\mathrm{T}}, w_i^{\mathrm{R}}$, are all taken

as the optimization variables to optimize simultaneously. The numbers of the variables and objectives are twice as many as Example A. This problem can be formulated as follows

$$\begin{aligned} \text{find} \quad & \left(d_i^{\mathrm{T}}, d_i^{\mathrm{R}}, w_i^{\mathrm{T}}, w_i^{\mathrm{R}}\right) \\ \min \quad & fitness \\ \text{s.t.} \quad & \ell_{\mathrm{T}} = \ell_{\mathrm{R}} = 50\lambda, \\ & 0 \prec w_i^{\mathrm{T}}, w_i^{\mathrm{R}} \prec 1,\ d_i^{\mathrm{T}}, d_i^{\mathrm{R}} \succeq \lambda/2. \end{aligned} \tag{12}$$

where the fitness function $fitness$ can be defined by

$$fitness = \alpha|\mathrm{Psll} - \mathrm{Psll_d}| + \beta|\mathrm{Null} - \mathrm{Null_d}| \tag{13}$$

where $\mathrm{Null_d}$ denotes the desired null level, and α, β are the weight coefficients used to tune. In this case, $\mathrm{Psll_d}$, $\mathrm{Null_d}$ are equal to $-33\,\mathrm{dB}$, $-95\,\mathrm{dB}$, respectively. The weight coefficients $\alpha = 0.9, \beta = 0.1$.

After the iteration process is finished, the simulation results are shown in Table 6, which shows the best results obtained by IWOA, comparing Psll, Psll_{ave}, Null, and Null_{ave} with those of DPSO [3], GA [2], and CDE [7]. It can be seen that IWOA achieves better results.

Table 6. Performance comparison with different algorithms in Example B ($\mathrm{Psll}_d = -33\,\mathrm{dB}$, $\mathrm{Null}_d = -95\,\mathrm{dB}$).

Algorithms	Psll	Psll_{ave}	Null	Null_{ave}
IWOA	**−33.12**	**−41.61**	**−94.99**	**−55.81**
DPSO [3]	−32.2	−28.32	−79.03	–
GA [2]	–	–	–	–
CDE [7]	–	–	–	–

Then we further investigate the influence of the varying desired nulls on the resultant Psll in the presence of a fixed desired Psll. Table 7 gives the resultant results when null levels are decreased by the step of $-20\,\mathrm{dB}$ from $-50\,\mathrm{dB}$ to $-90\,\mathrm{dB}$, whose radiation patterns are plotted in Fig. 3. As can be observed, when we decrease the desired null Null_d, the real null level Null will be decreased. Thus in practice, we have to make a tradeoff between the lower Psll and lower Null.

Table 7. Performance comparison under the varying desired nulls (Null_d) and the same desired Psll (Psll_d) (dB).

Null_d	Psll_d	Psll	Psll_{ave}	Null	Null_{ave}
−50	−33	**−33.05**	−39.96	−65.91	−53.23
−70	−33	−32.99	−39.89	−69.99	−58.71
−90	−33	−32.58	**−41.01**	**−89.99**	**−59.12**

4.3. Imposing Wide Null by Optimizing the Spacing and Excitation for Both Subarrays

This case is based on imposing wide nulls at the specified angles on radiation pattern to reject the interferences. We consider not only the case of imposing a wide null, but also the case of imposing two wide nulls.

For imposing a wide null, assume that the inference is located in $[-40^\circ, -35^\circ]$. Thus to overcome such interference by imposing wide null, a wide null should be placed in $[-40^\circ, -35^\circ]$ and is assumed below $-40\,\mathrm{dB}$. At the same time, we expect the obtained Psll to be as small as possible. Several related parameters are assigned as follows: $\mathrm{Psll}_d = -25\,\mathrm{dB}$, $\mathrm{Null}_d = -40\,\mathrm{dB}$.

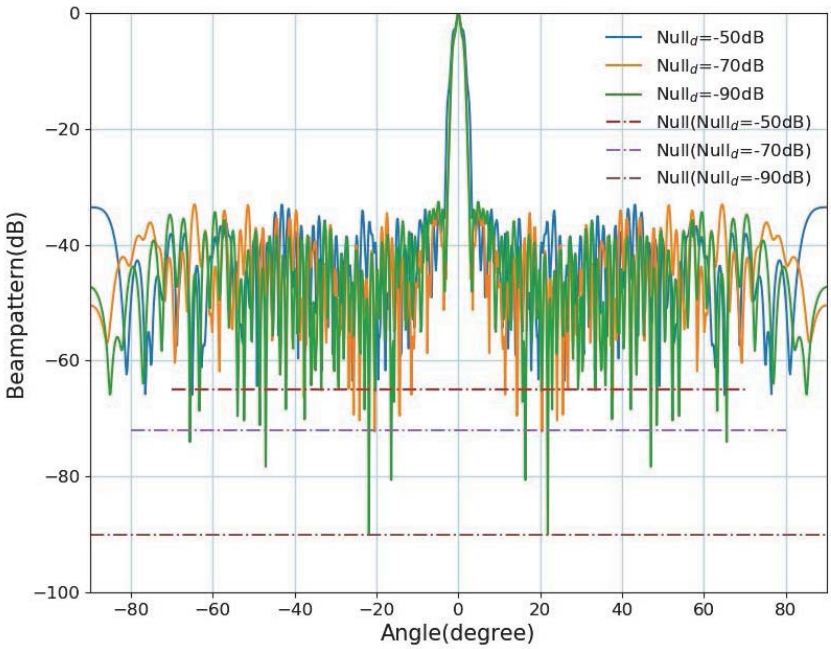

Figure 3. Resultant radiation patterns under the varying desired nulls.

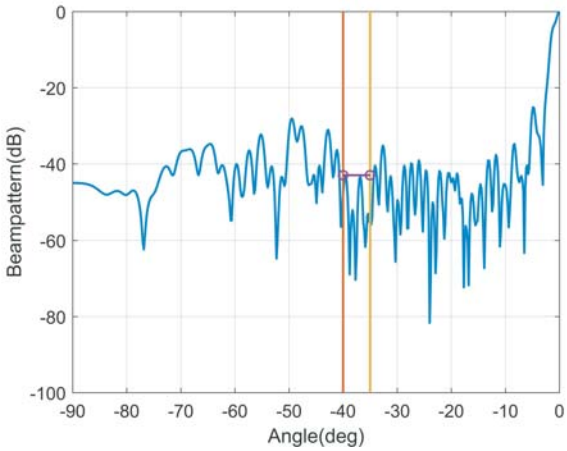

Figure 4. Radiation pattern with a wide null locating in $[-40°, -35°]$.

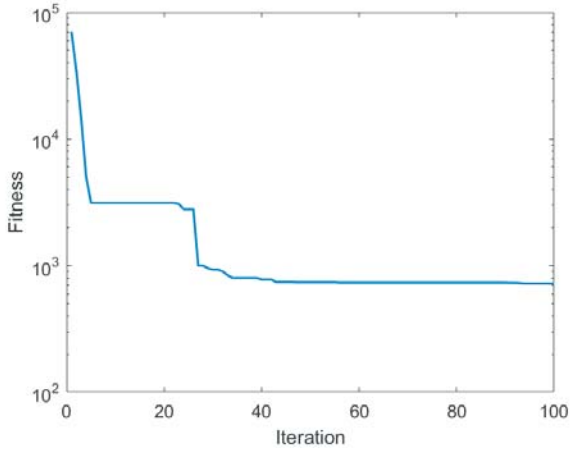

Figure 5. Converge curve to implement a wide null locating in $[-40°, -35°]$ (Pattern corresponding to Fig. 4).

In this optimization, the fitness function $fitness$ [35] will be defined by the following

$$fitness = \sum_{i=1}^{n} \frac{1}{\Delta\theta_i} \int_{\theta_{il}}^{\theta_{iu}} |\delta(\theta)|^2 \, d\theta + \sum_{j=1}^{m} |f(\theta_j)|, \text{where}$$
$$\delta(\theta_i) = \begin{cases} \text{Null}_d - f(\theta_i), \text{if } \theta_i \in [-40°, -35°] \\ 0, \text{otherwise.} \end{cases} \tag{14}$$

where n and m stand for the number of θ_i and the desired null.

In the right-hand side of Eq. (14), the first term aims at suppressing the sidelobe level of radiation pattern, and the second term is used to control the null at the prescribed directions.

The simulation results are shown in Table 8, Fig. 4, and Fig. 5. Table 8 gives the excitation and position of each element, together with the related metric measurements, including Psll, Psll_{ave}, $\text{Null}_{\min}$, and Null_{ave}. The corresponding radiation pattern is depicted in Fig. 4. As can be seen, a wide null located in $[-40°, -35°]$ is implemented as expected, and meets the required Psll as well. Fig. 5 shows the convergence curve with iteration, as can be seen in the figure. IWOA has a rapid convergence rate that is consistent with the previous test result.

For imposing the two wide nulls, assume that the inferences are located in $[-40°, -35°]$ and $[-65°, -60°]$. To overcome such interferences by imposing the wide nulls, we need to place the two wide nulls in the locations of the corresponding interferences. To exhibit the flexibility of the method, one wide null located in $[-40°, -35°]$ will be required to be below -43 dB, which remains the same with the case of the single wide null, and the other wide null located in $[-65°, -60°]$ will be required to be below -50 dB. At the same time, we hope the obtained Psll as small as possible. In this case, $\text{Psll}_d = -25$ dB, $(\text{Null}_d \in [-40°, -35°], \text{Null}_d \in [-65°, -60°]) = (-43\,\text{dB}, -50\,\text{dB})$.

The simulated results are shown in Table 9, Fig. 6, and Fig. 7. Table 9 gives the excitation and position of each element, together with the related metric measurements, including Psll, Psll_{ave}, $\text{Null}_{\min}$, and Null_{ave}. The corresponding radiation pattern is depicted in Fig. 6. As can be seen, the two wide

Table 8. Positions and excitations of elements with null in $[-40°, -35°]$ of Fig. 4.

No.	d_{T}	d_{R}	w_{T}	w_{R}
1	0.00	0.00	0.15	0.13
2	1.39	1.34	0.14	0.14
3	2.06	2.02	0.12	0.13
4	2.75	2.72	0.15	0.12
5	3.45	3.85	0.13	0.15
6	4.12	4.55	0.15	0.13
7	4.81	5.23	0.13	0.11
8	5.48	5.88	0.12	0.13
9	6.15	6.55	0.11	0.13
10	6.84	7.25	0.13	0.13
11	7.51	7.95	0.13	0.12
12	8.66	8.63	0.13	0.11
13	9.33	9.30	0.13	0.13
14	10.00	10.00	0.12	0.14
15	10.66	10.66	0.15	0.23
16	11.31	11.35	0.10	0.13
17	11.98	12.05	0.13	0.12
18	13.13	12.71	0.12	0.13
19	14.26	13.38	0.12	0.23
20	14.91	14.05	0.14	0.13
21	15.57	14.70	0.14	0.23
22	16.24	15.36	0.11	0.12
23	17.37	16.04	0.12	0.15
24	18.51	16.71	0.13	0.12
25	50.00	50.00	0.13	0.11

Results	Psll	Psll_{ave}	$\text{Null}_{\min}$	Null_{ave}
(dB)	26.83	−40.34	−62.56	−49.09

Table 9. Positions and excitations of elements with both nulls in $[-40°, -35°]$ and $[-65°, -60°]$ of Fig. 6.

No.	d_{T}	d_{R}	w_{T}	w_{R}
1	0.00	0.00	0.14	0.13
2	1.00	1.00	0.10	0.10
3	1.50	1.50	0.10	0.10
4	2.00	2.00	0.10	0.16
5	2.50	2.50	0.10	0.10
6	3.00	3.00	0.10	0.10
7	3.50	3.75	0.10	0.14
8	4.00	4.25	0.17	0.16
9	4.50	4.75	0.11	0.10
10	5.27	5.25	0.10	0.10
11	6.02	5.75	0.10	0.10
12	6.52	6.25	0.10	0.15
13	7.02	7.08	0.15	0.10
14	7.52	7.73	0.11	0.10
15	8.02	8.23	0.10	0.10
16	8.52	8.73	0.10	0.10
17	9.02	9.23	0.15	0.10
18	9.52	9.76	0.13	0.10
19	10.18	10.42	0.14	0.10
20	10.68	10.92	0.15	0.10
21	11.18	11.42	0.10	0.13
22	11.68	11.92	0.16	0.10
23	12.18	12.59	0.10	0.10
24	12.68	13.09	0.13	0.10
25	50	50	0.1	0.1

Results	Psll	Psll_{ave}	$\text{Null}_{\min}$	Null_{ave}
(dB)	−26.16	−39.41	−69.84	−52.01

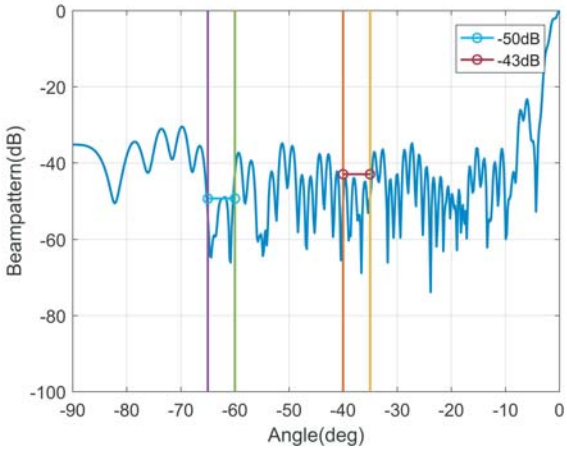

Figure 6. Radiation pattern with the two wide nulls in $[-40°, -35°] \cup [-65°, -60°]$.

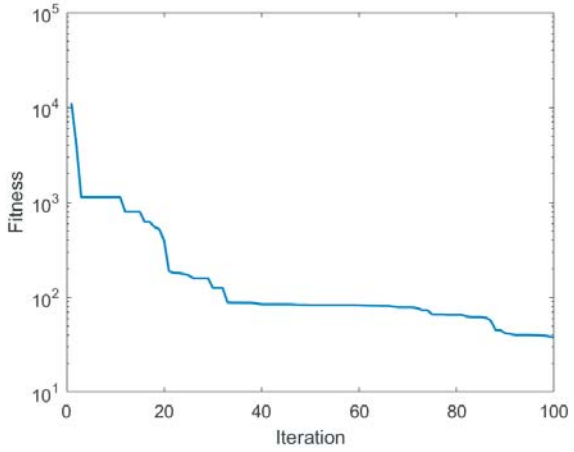

Figure 7. Converge curve to implement the two wide nulls locating in $[-40°, -35°] \cup [-65°, -60°]$ (Pattern corresponding to Fig. 6).

nulls located in $[-40°, -35°] \cup [-65°, -60°]$ are implemented as expected and also reach the required Psll, respectively. Fig. 7 shows the convergence curve with iteration, and it shows that IWOA improves the solution continually.

5. CONCLUSION

In this study, we adapt WOA to optimize the problems of two-way pattern of MIMO radar, involving the achievements of sidelobe suppression, deep null, and wide null. The test of IWOA based on the benchmark functions is conducted and compared with those of several state-of-the-are algorithms. The obtained results show that the improvement enhances the performance in population diversity, exploration and exploitation capabilities, and exhibits a better convergence rate. Finally, examples on MIMO radar are given, and the obtained numerical results validate that IWOA is able to efficiently optimize the synthesis problem of MIMO radar system. Additionally, the effect of varying desired nulls on Psll is also investigated.

ACKNOWLEDGMENT

This work was supported in part by the China Scholarship Council and the Excellent Doctorate Cultivating Foundation of Northwestern Polytechnical University. We would also like to greatly thank Qingping Lei for giving suggestions to this paper.

REFERENCES

1. Cheng, Z., Z. He, R. Li, and Z. Wang, "Robust transmit beampattern matching synthesis for MIMO radar," *Electronics Letters*, Vol. 53, No. 9, 620–622, 2017.
2. Zhang, Z., Y. Zhao, and J. Huang, "Array optimization for MIMO radar by genetic algorithms," *2nd Int. Congress Image Signal Processing, Conference Proceedings*, Vol. 1, 1–4, 2009.
3. Jiang, Y., J. Dong, F. Liu, Y. Yue, and R. Shi, "Pattern synthesis of MIMO radar using differential particle swarm optimization algorithm," *Optik — International Journal for Light and Electron Optics*, Vol. 126, No. 24, 5781–5786, 2015.
4. Pandey, N. and L. P. Roy, "Convex optimisation based transmit beampattern synthesis for MIMO radar," *Electronics Letters*, Vol. 52, No. 9, 761–763, 2016.

5. Rezer, K., W. Gropengießer, and A. F. Jacob, "Particle swarm optimization of minimum-redundancy MIMO arrays," *Microwave Conference, Conference Proceedings*, Vol. 1, 1–4, 2011.
6. Sharifabad, F. K., M. A. Jensen, and A. L. Anderson, "Array beamforming synthesis for point-to-point MIMO communication," *IEEE Transactions on Antennas & Propagation*, Vol. 63, No. 9, 3878–3886, 2015.
7. Jiang, Y., J. Dong, and Z. Shi, "Pattern synthesis of MIMO radar based on chaotic differential evolution algorithm," *Optik — International Journal for Light and Electron Optics*, Vol. 140, 794–801, 2017.
8. Goldsmith, A., S. A. Jafar, N. Jindal, and S. Vishwanath, "Capacity limits of MIMO channels," *IEEE JSAC*, Vol. 21, No. 5, 684–702, 2003.
9. Paulraj, A. J., D. A. Gore, R. U. Nabar, and H. Bolcskei, "An overview of MIMO communications — A key to gigabit wireless," *Proceedings of the IEEE*, Vol. 92, No. 2, 198–218, 2004.
10. Edfors, O. and A. J. Johansson, "Is Orbital Angular Momentum (OAM) based radio communication an unexploited area?," *IEEE Transactions on Antennas & Propagation*, Vol. 60, No. 2, 1126–1131, 2012.
11. Qu, Y., G. Liao, S.-Q. Zhu, X.-Y. Liu, and H. Jiang, "Performance analysis of beamforming for MIMO radar," *Progress In Electromagnetics Research*, Vol. 84, 123–134, 2008.
12. Ghayoula, E., R. Ghayoula, M. Haj-Taieb, J.-Y. Chouinard, and A. Bouallegue, "Pattern synthesis using hybrid fourier-neural networks for IEEE 802. 11 MIMO application," *Progress In Electromagnetics Research*, Vol. 67, 45–58, 2016.
13. Khodier, M. M. and C. G. Christodoulou, "Linear array geometry synthesis with minimum sidelobe level and null control using particle swarm optimization," *IEEE Transactions on Antennas & Propagation*, Vol. 53, No. 8, 2674–2679, 2005.
14. Zhang, C., X. Fu, L. Leo, S. Peng, and M. Xie, "Synthesis of broadside linear aperiodic arrays with sidelobe suppression and null steering using whale optimization algorithm," *IEEE Antennas and Wireless Propagation Letters*, Vol. 17, No. 2, 347–350, 2018.
15. Recioui, A., "Capacity optimization of MIMO wireless communication systems using a hybrid genetic-taguchi algorithm," *Wireless Personal Communications*, Vol. 71, No. 2, 1003–1019, 2013.
16. Recioui, A., "Application of a galaxy-based search algorithm to MIMO system capacity optimization," *Arabian Journal for Science & Engineering*, Vol. 41, No. 9, 3407–3414, 2016.
17. Jiang, Y., J. Dong, and Z. Shi, "Pattern synthesis of MIMO radar based on chaotic differential evolution algorithm," *Optik — International Journal for Light and Electron Optics*, Vol. 140, 794–801, 2017.
18. Yuan, P., C. Guo, Q. Zheng, and J. Ding, "Sidelobe suppression with constraint for MIMO radar via chaotic whale optimisation," *Electronics Letters*, Vol. 54, No. 5, 311–313, 2018.
19. Recioui, A. and H. Bentarzi, "MIMO capacity enhancement in spatially correlated channels using taguchi method," *WSEAS International Conference on Mathematical Methods*, 2010.
20. Recioui, A., "Application of the spiral optimization technique to antenna array design," *Handbook of Research on Emergent Applications of Optimization Algorithms*, 364–385, IGI Global, 2018.
21. Mirjalili, S. and A. Lewis, "The whale optimization algorithm," *Advances in Engineering Software*, Vol. 95, 51–67, 2016.
22. Yuan, P. and Z. Shi, "Synthesis of non-uniform linear array antenna based on WOA," *Radio Engineering*, Vol. 5, No. 10, 53–58, 2017.
23. Plessis, W. P. D. and A. B. Ghannam, "Improved seeding schemes for interleaved thinned array synthesis," *IEEE Transactions on Antennas & Propagation*, Vol. 62, No. 11, 5906–5910, 2014.
24. Zhou, Y., L. Ying, and Q. Luo, "Lvy flight trajectory-based whale optimization algorithm for global optimization," *IEEE Access*, Vol. 5, No. 99, 1–1, 2017.
25. Yan, B., Z. Zhao, Y. Zhou, W. Yuan, J. Li, J. Wu, and D. Cheng, "A particle swarm optimization algorithm with random learning mechanism and levy flight for optimization of atomic clusters," *Computer Physics Communications*, Vol. 219, 79–86, 2017.

26. Hakl, H. and H. Uuz, "A novel particle swarm optimization algorithm with levy flight," *Applied Soft Computing Journal*, Vol. 23, No. Complete, 333–345, 2014.
27. Jensi, R. and G. W. Jiji, "An enhanced particle swarm optimization with levy flight for global optimization," *Applied Soft Computing*, Vol. 43, No. C, 248–261, 2016.
28. Mirjalili, S., "SCA: A sine cosine algorithm for solving optimization problems," *Knowledge-Based Systems*, Vol. 96, 120–133, 2016.
29. Chegini, S. N., A. Bagheri, and F. Najafi, "Psoscalf: A new hybrid pso based on sine cosine algorithm and levy flight for solving optimization problems," *Applied Soft Computing*, Vol. 73, 697–726, 2018.
30. Yao, X., Y. Liu, and G. Lin, "Evolutionary programming made faster," *IEEE Transactions on Evolutionary Computation*, Vol. 3, No. 2, 82–102, 2002.
31. Digalakis, J. G. and K. G. Margaritis, "On benchmarking functions for genetic algorithms," *International Journal of Computer Mathematics*, Vol. 77, No. 4, 481–506, 2001.
32. Yang, X. S., "Firefly algorithm, stochastic test functions and design optimisation," *International Journal of Bio-Inspired Computation*, Vol. 2, No. 2, 78–84(7), 2010.
33. Rashedi, E., H. Nezamabadi-Pour, and S. Saryazdi, "GSA: A gravitational search algorithm," *Information Sciences*, Vol. 179, No. 13, 2232–2248, 2009.
34. Xin, Y., L. Yong, and G. Lin, "Evolutionary programming made faster," *IEEE Trans. Evol. Comput.*, Vol. 3, No. 2, 82–102, 1999.
35. Mahto, S. and A. Choubey, "A novel hybrid iwo/wdo algorithm for interference minimization of uniformly excited linear sparse array by position-only control," *IEEE Antennas & Wireless Propagation Letters*, Vol. 15, No. 3, 250–254, 2016.