Progress In Electromagnetics Research M, Vol. 37, 21–30, 2014

Synthesis of Thinned Array with Side Lobe Levels Reduction Using Improved Binary Invasive Weed Optimization

Chao Liu and Hua-Ning Wu*

Abstract—As a very powerful optimization algorithm, invasive weed optimization has been widely applied to continuous optimization problems in electromagnetic (EM) field. However, the optimization of a thinned array can be formulated as a discrete-variable optimization problem with solutions encoded as binary strings. Therefore, in this paper, an improved binary invasive weed optimization (IBIWO) is proposed to design a thinned array with minimum side lobe levels. To evaluate the performance of the proposed algorithm, two examples have been presented and solved. Simulation results of the proposed thinned arrays obtained by IBIWO are compared with published results to verify the effectiveness of the proposed method.

1. INTRODUCTION

Thinning an array involves the removal (turning off) of some elements from a periodic or uniformly spaced array to create a desired pattern. An element connected to the feed network is 'turned on', and an element connected to a matched load is 'turned off' [1, 2]. The main motivation to use thinning is the reduction in cost, weight and power consumption. And thinned arrays present the advantage of easiness of realization, as different elements usually lie on a regular grid, operate with equal amplitude, and are directly connected to the amplifiers [3]. Hence, the synthesis of arrays using thinning is under active research by many groups. Statistical methods were widely used to tackle the design problem of antenna array [4]. Most of these classical optimization methods (such as Newton methods, down-hill and conjugate gradient) are not well suited for thinning an array, because they can only optimize a few continuous problems and often get stuck in local optimal. Therefore, many stochastic, probabilistic or evolutionary optimization approach, such as simulated annealing [5], genetic algorithm [1], immune algorithm [6], ant colony optimization [7], different evolutions [8], particle swarm optimization algorithm [9] were used and shown to be effective for the synthesis of thinned arrays. Some other approaches [10, 11] have also been proposed.

In 2006, a derivative-free, metaheuristic algorithm proposed by Mehrbian and Lucas in [12], known as the Invasive Weed Optimization, mimicking the ecological behavior of colonizing weeds. Since its inception, IWO has found successful application in many electromagnetic problems like design of Printed Yagi Antenna [13], E-shaped MIMO antenna [14], multi-feed reflector antennas [15], Broadband Patch Antenna [16], Conformal Phased Arrays [17], Circular Antenna Arrays [18] etc.. Results obtained using IWO for these problems are encouraging. However, the nature of reproduction operators in classical IWO limits its application. In fact, the optimization of a thinned array can be formulated as a discrete-variable optimization problem with solution encoded as binary strings. In [19, 20], binary invasive weed optimization (BIWO) has been proposed and applied to search best solutions of typical benchmark functions. To the best of our knowledge, the application of the binary version of IWO for antenna design has not yet been reported. The main contribution of this paper is to employ an improved binary

Received 24 April 2014, Accepted 7 May 2014, Scheduled 15 June 2014
* Corresponding author: Hua-Ning Wu (wuhuaning007@163.com).
The authors are with the Department of Electronic Engineering, Naval University of Engineering, Wuhan 430033, China.

IWO to synthesize thinned arrays with lower SLL. The adaptive dispersion mechanism is adapted to improve the search ability of binary IWO.

The rest of the paper is organized as follows. A formulation of the thinned array pattern synthesis as an optimization task is discussed in Section 2. Section 3 gives a comprehensive overview of the proposed IBIWO algorithm. Section 4 presents the simulation results, and in Section 5 conclusions are presented.

2. THINNED ARRAY

According to the structure shown in Figure 1, where there are $2M$ isotropic elements placed symmetrically along the x-axis, and array factor AF at θ angle in XZ plane for a linear antenna array can be expressed as [7, 9]:

$$AF = \sum_{m=-M}^{m=M} A_m \exp\left(j * \left(\frac{2\pi}{\lambda} x_m * \sin\theta + \varphi_m\right)\right) \tag{1}$$

where x_m, I_m and φ_m are the position, excitation amplitude and phase of the mth element, respectively. In thinned array, I_m is 0 if the state of the mth element is 'off', and I_m is 1 if it is 'on'. In our case, the distance between elements is 0.5λ, and there is no element located at the axis origin. All elements have a uniform excitation phase ($\varphi_m = 0$). Thus, Equation (1) can be rewritten as [7, 9]

$$AF = 2\sum_{m=1}^{M} I_m \cos(\pi * (m - 0.5)\sin\theta) \tag{2}$$

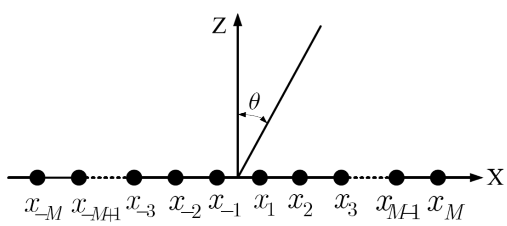


Figure 1. Geometry of a symmetric linear array with $2M$ element.

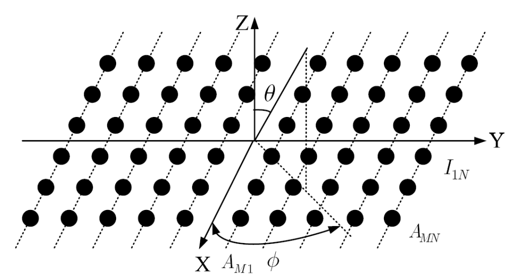


Figure 2. Geometry of a symmetric planar array with $2M \times 2N$ element.

Figure 2 shows a planar array with $2M \times 2N$ elements. Assuming the same considerations as in the linear array, the array factor in this structure is given by [7, 9]:

$$AF = 4\sum_{m=1}^{M}\sum_{n=1}^{N} A_{mn} \cos\left[\pi (m - 0.5)\sin\theta\cos\phi\right] \cdot \cos\left[\pi (n - 0.5)\sin\theta\sin\phi\right] \tag{3}$$

where ϕ is the azimuth angle with respect to x-axis and θ the elevation angle with respect to the Z-axis. The amplitude of excitation A_{mn} and the spacing between elements are both symmetrical about the X and Y axes. The spacing is fixed equal to 0.5λ. Thus, the array factor can be simplified by computing a quarter of the rectangular array.

In order to control the array pattern as desired, different parameters of the far field pattern must be considered in the fitness function. The first and most important parameter is the normalized maximum side lobe level (MSL) which is desired to be as low as possible. The normalized maximum side lobe level of the antenna array can be given by Equation (4).

$$F_{MSL}(I) = \max_{\forall\theta\in R}\left\{20\log\left|\frac{AF(I,\theta)}{AF_{\max}}\right|\right\} \tag{4}$$

where R represents the side lobe region excluding the main beam. In this paper, Equation (5) is used as the objective function to suppress SLL.

$$f(I) = F_{MSL}(I) \tag{5}$$

For the planar array, the fitness function is the sum of the maximum MSL in $\Phi = 0^\circ$ and $\Phi = 90^\circ$ planes, which can be expressed as:

$$f(I) = F_{MSL}(I, \Phi = 0^\circ) + F_{MSL}(I, \Phi = 90^\circ) \tag{6}$$

We need to find which array elements should be enabled or disable ($A_{mn} = 1$ or $A_{mn} = 0$) to get the desired radiation pattern characteristics. Hence, the thinned array synthesis problem can be formulated as the following 0–1 integer optimization problem:

$$\min(f(I)) \quad \text{s.t } I_{nm} \in \{0, 1\}, \quad m = 1, 2, 3 \ldots M, \quad n = 1, 2, 3 \ldots N$$

3. INVASIVE WEED OPTIMIZATION ALGORITHM

3.1. Continuous IWO

Invasive Weed Optimization (IWO) is a meta-heuristic algorithm that mimics the colonizing behavior of weeds. The IWO algorithm may be summarized as four steps, and more details can be found in [12]:

(I) Initialization: In this step, the solutions are randomly initialized and dispersed in the given d-dimensional search space.

(II) Reproduction: In this step the parent weed produces seeds depending on its own fitness as well as the colony's lowest and highest fitness. So if the weed is better fitted then it will have more offspring and if less fitted then less offspring

(III) Spatial distribution: the generated seeds are randomly scattered over the d-dimensional search space by perturbing them with normally distributed random numbers with zero mean and a variable variance. The standard deviation for a particular iteration can be given as in Equation (7):

$$\delta_{cur} = \frac{(iter_{\max} - iter)^n}{iter_{\max}^n} (\delta_{initial} - \delta_{final}) + \delta_{final} \tag{7}$$

where $\delta_{initial}$ and δ_{final} are initial value and final value of standard deviation, δ_{cur} is the standard deviation of current iteration, n is the nonlinear index, $iter_{\max}$ is the maximum iteration, $iter$ is current iteration. The position of the new seed can be given as in Equation (8):

$$x_{son} = x_{parent} + sd = x_{parent} + randn(0, 1) * \delta_{cur} \tag{8}$$

(IV) Competitive Exclusion: The new seeds produced grow to flowering weeds and are placed together with parent weeds in the colony. Some kind of competition between plants is needed for limiting maximum number of plants in a colony. Weeds with worst fitness are eliminated until the maximum number of weeds $P_{\max}$ in the colony is reached The steps 1 to 4 are repeated until the maximum number of iterations has reached

3.2. BIWO

The presentation used in IWO is a real-valued vector. To be able to use IWO for discrete problems, two aspects need to be changed. Firstly, adopt the appropriate coding method for characterizing the actual problem. In general, the weed in BIWO will be replaced by the binary coding sequence $(x_1, x_2 \ldots x_{n-1}, x_n)$, each x_i takes value of 1 or 0. Secondly, binary space spread should be employed. In IWO algorithm, the generated seeds are randomly scattered over the d-dimensional search space by perturbing them with normally distributed random numbers with zero mean and a variable standard deviation. But this continuous spread method will not make sense in the binary coding sequence. We take a mutation mechanism to create seed in the BIWO. Each bit in the parent weed will be mutated with a probability p_i^k which is calculated by the following sigmoid function.

$$p_i^k = \frac{1}{1 + e^{-d_i^k + 6}} + \frac{1}{1 + e^{d_i^k + 6}} \tag{9}$$

where, d_i^k is the spread distance generated by the normally distributed function $N(0, \sigma_{cur})$. When the mutation probability of each bit in a weed is calculated, then a uniformly distributed random numbers in the range $[0, 1]$ will be generated to specify the mutation bits through following equations

$$\overline{d_i^k} = \begin{cases} 1 & \text{if } \left(\rho < f(d_i^k)\right) \\ 0 & \text{else} \end{cases} \tag{10}$$

$$x_i^{k+1} = \bmod\left(x_i^k + \overline{d_i^k}, 2\right) \tag{11}$$

In literature [20], the recommended value of $\delta_{initial}$ and δ_{final} of BIWO is 3 and 15.

3.3. IBIWO

As mentioned in Section 3.1, we know that the function of δ_{cur} defines the exploration ability and exploitation ability of the algorithm, acts as both diversification and intensification components of BIWO, and has a great effect on final solutions. In early iterations, the bigger δ_{cur} will help the algorithm to explore the solution space as much as it can. A good diversification will make the final solution near global optimum. The algorithm will use this component to identify most potential spaces where the global optimum may lie in. A good intensification will help the algorithm to exploit the potential areas and find the global optimum. It will increase the convergence speed of the algorithm and search a better final solution. Hence, it is very important to keep an efficient balance between diversification and intensification of the algorithm. But, BIWO algorithm uses a fix δ_{cur} to produce seeds related to each weed and suffers from the lack of fine balance between exploration and exploitation.

In order to overcome the drawbacks of BIWO, adaptive dispersion mechanism is integrated in the BIWO algorithm. The adaptive dispersion mechanism is that the δ_{cur} of the current generation distribute linearly among the weeds as weed with the highest fitness achieves the lowest δ_{cur}, and the lowest fitness achieves the highest δ_{cur}, which can be represented by Equation (12). j is the index of weeds in the colony sorted according to their fitness, and σ_{cur} can be calculated by Equation (1). p_{sum} is the sum number of weeds in the current generation and σ_j the δ_{cur} of j-th weeds to produce seeds. Hence, the plant with lower fitness will have the chance to produce good seeds in current generation. In addition, this process will increase the diversification of algorithm and improve the search ability of the colony so that the algorithm will explore the search space effectively.

$$\sigma_j = (\sigma_{cur}) \times \left(\frac{j}{p_{sum}}\right) \tag{12}$$

The pseudo-code of the novel binary IBIWO is given as follows:

Randomly produce weeds P_0 in binary coding sequence;
For each $iter < iter_{\max}$ do
Calculate each weed's fitness value by Equations (5) or (6);
Sort all fitness values;
Record maximum and minimum value;
Calculate the number of seeds by Equation (1);
Calculate each seed's standard variance by Equations (7) and (12);
Produce seeds by Equations (9)–(11);
Add the seed to population;
If $((Num = |X|) > P_{\max})$ then;
Sort the population X of their fitness;
Save population of weed with best fitness until $Num = P_{\max}$;
End if
End for

4. SIMULATION RESULTS

In this section, we use two thinned array cases to evaluate the search ability of the proposed algorithm. In both cases, the array with all elements 'on' is used as initial solutions. All simulations are conducted in a Windows 7 Professional OS environment using 12-core processors with Intel Xeon (R), 3.33 GHz, 72GB RAM, and the codes are implemented in Matlab 7.10.

The first case discussed here is to thin a linear array with 100 elements symmetrically spaced 0.5λ apart along the x-axis with its center at the origin in order to generate a broadside symmetric pattern [1, 7, 9]. In [7], Quevedo-Teruel and Rajo-Iglesias utilize the ACO algorithm to the pattern synthesis of linear thinned array. In [1], Haupt designs the same array using GA. Wang et al. use chaotic binary PSO algorithm in the linear thinned array design and obtain a better result than the other algorithms [9]. In this work, IBIWO is utilized to design the thinned array with lower MSL. For comparison, BIWO is also applied to optimize the same problem. The values of parameters of IBIWO and BIWO are listed in Table 1. Because of the randomness nature, all the experiments have been run 100 times with 300 iterations independently. The stopping criterion of each run is to complete the number of iterations. The best and average results are presented in this section.

Table 1. Parameters of IBIWO and BIWO.

Parameter	Value	Parameter	Value
initial size of colony	20	maximum number of seed	5
maximum size of colony	50	minimum number of seed	0
$\delta_{initial}$	15	dimensional	50
δ_{final}	3	$iter_{\max}$	300

According to the symmetrical structure shown in Figure 1, only 50 elements are to be optimized. In this case, fitness function shown as Equation (5) is minimized using the IBIWO. Figure 3 shows the best pattern obtained by the IBIWO, and the result is compared with the initial value with all elements turned on. From Figure 3, we notice that the MSL with the full 100-element linear array (all elements are turned on) is -13.73 dB, and the MSL is lowered to -22.12 dB after thinning by IBIWO.

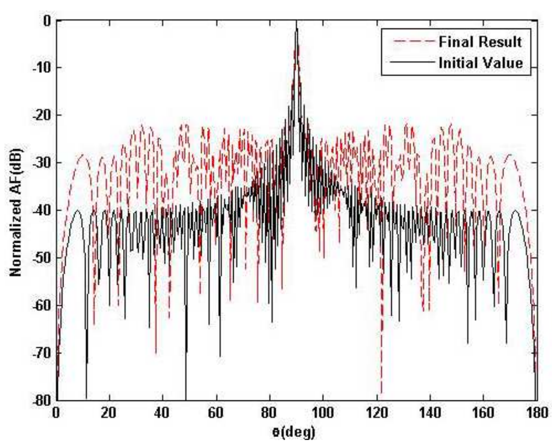


Figure 3. Radiation pattern compared with the initial value.

Figure 4 gives the element status of the best thinned array with minimum MSL obtained by IBIWO algorithm. As shown in Figure 4, the number of turned off elements is 22.

To further verify the performance of the IBIWO, it is compared with the GA [1], ACO [7], BDE [8], BPSO [9], CBPSO [9] and BIWO [19]. The obtained array patterns using ACO, BIWO and IBIWO are presented in Figure 5. Figure 6 represents a comparison of the results obtained by the IBIWO algorithm

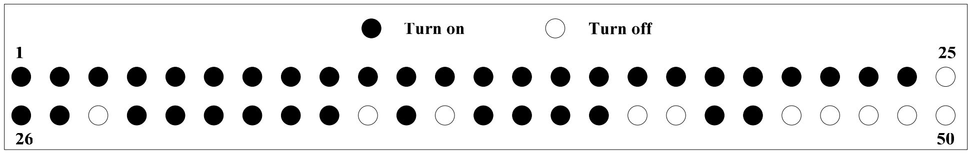


Figure 4. The elements status obtained by IBIWO.

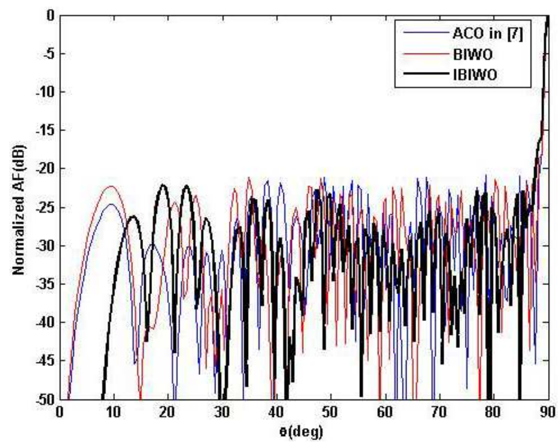
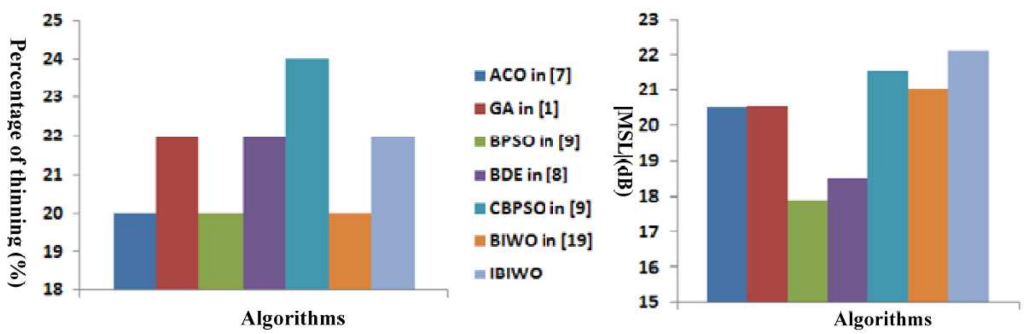


Figure 5. Comparisons of 100-elements thinned linear array pattern obtained by IBIWO and other algorithms.



(a) Percentage of thinning (b) Absolute value of MSL (dB)

Figure 6. The best results obtained by IBIWO and other algorithms.

and the other six algorithms. The results used for comparison are given by [7, 9]. From Figure 6(a) we can clearly know that the percentage of thinning obtained by the IBIWO is 22%, which is more than that of 20% in [7, 9, 19] and 22% in [1, 8] and with lower MSL, except that of 24% obtained by CBPSO in [9]. As shown in Figure 6(b), the absolution value of MSL obtained by IBIWO is larger than that of other algorithms.

To evaluate the efficiency and reliability of the proposed algorithms, the IBIWO algorithm is further compared with the algorithms mentioned before in terms of average convergence speed. For each iteration step, the average fitness value is calculated from 100 fitness values derived at the certain step [9]. Figure 7 shows the variation of the average SLL value as a function of number of iterations. As shown in Figure 7, the IBIWO obtains the best average MSL, and the value is 22.04 dB. And the average convergence iterations up to the best average MSL is 200, which is smaller than that of the ACO, BPSO and BIWO, except for the GA, BDE and CBPSO. The details are listed in Table 2.

The second case discussed here is to design a thinned planar array with 20 × 10 elements, which

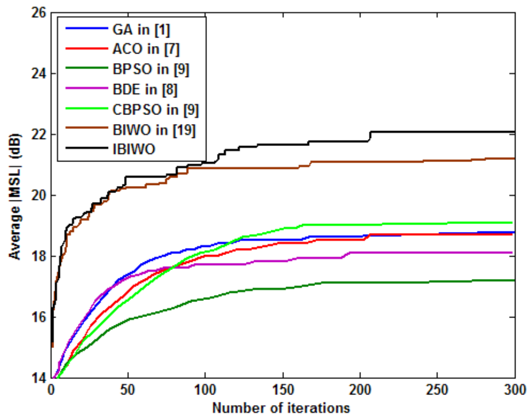


Figure 7. Convergence of the average |MSL| values versus the number of iterations.

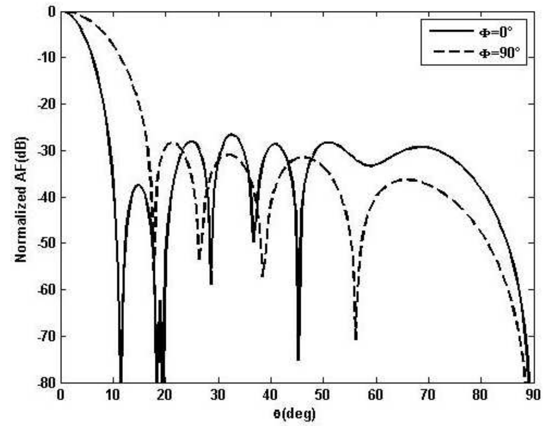


Figure 8. Radiation pattern of 20×10 thinned planar array achieved by the IBIWO in the planes $\Phi = 0°$ and $\Phi = 90°$.

Table 2. Comparisons of the simulation results.

Simulation results	GA In [1]	ACO In [7]	BDE In [8]	BPSO In [9]	CBPSO In [9]	BIWO In [19]	IBIWO
Average maximum |MSL|	19.2	19.1	18.5	17.6	20	21.31	22.04
Average convergence iterations	164	203	193	236	181	242	20

was optimized in [1, 7–9]. In [1], Haupt utilized GA algorithm to a 20×10 planar array. In [7], ACO algorithm was applied to the same problem by Quevodo-Teruel and Rajo-Iglesias. Zhang et al. designed the planar array using the Boolean Differential Evolution (BDE) in [8], and Wang et al. applied the Chaotic Binary Particle Swarm Optimization (CBPSO) for the synthesis of thinned arrays [9]. In this case, IBIWO and BIWO are also utilized to design the thinned array with lower MSL. The values of parameters are the same as the first case.

In this case, the MSLL is suppressed in the planes $\Phi = 0°$ and $\Phi = 90°$. Hence, Equation (6) is selected as the fitness function optimized by IBIWO. The best radiation pattern of the optimized array obtained by the proposed algorithm is plotted in Figure 8. The best fitness obtained by IBIWO is -54.9 dB (MSL $= -26.58$ dB in $\Phi = 0°$ plane and MSL $= -28.32$ dB in $\Phi = 90°$ plane, as shown in Figure 8).

The configuration of the best array is given for a quadrant of the array elements plotted in Figure 9. The number of elements turned off is 24 in the quarter of the rectangular array.

To further verify the performance of the IBIWO, it is compared with the GA [1], ACO [7], BDE [8], BPSO [9], CBPSO [9] and BIWO [19]. The radiation pattern obtained by the IBIWO, ACO, BIWO in $\Phi = 0°$ and $\Phi = 90°$ plane are shown in Figure 10. Figure 11 gives the |MSL| values of comparisons of various algorithms. The comparisons in Figure 11 demonstrate that the IBIWO can achieve the best MSL in $\Phi = 0°$ and $\Phi = 90°$ plane. Figure 12 illustrates that the percentage of thinning obtained by the IBIWO outperforms that of the other six algorithms.

Details of comparative results in terms of the convergence characteristic are carried out and shown in Figure 13 and Table 3. Table 3 shows that the BIWO algorithm can obtain the best average MSLL, and the value is 24.97. The number of iterations up to the best average MSLL of BIWO is 212, which is smaller than BPSO and BIWO, except for that of GA, ACO, CPSO and BDE. From Figure 13, we

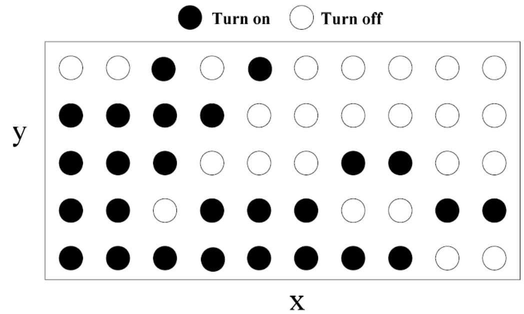
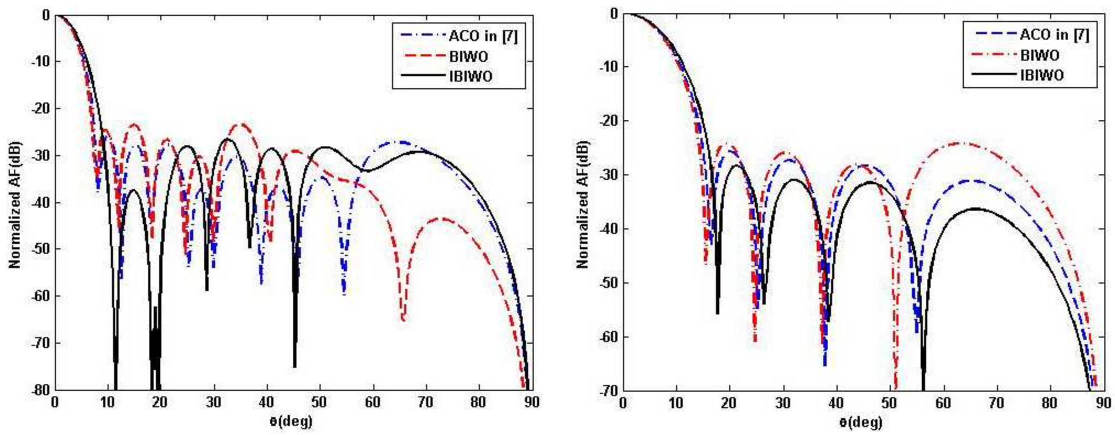
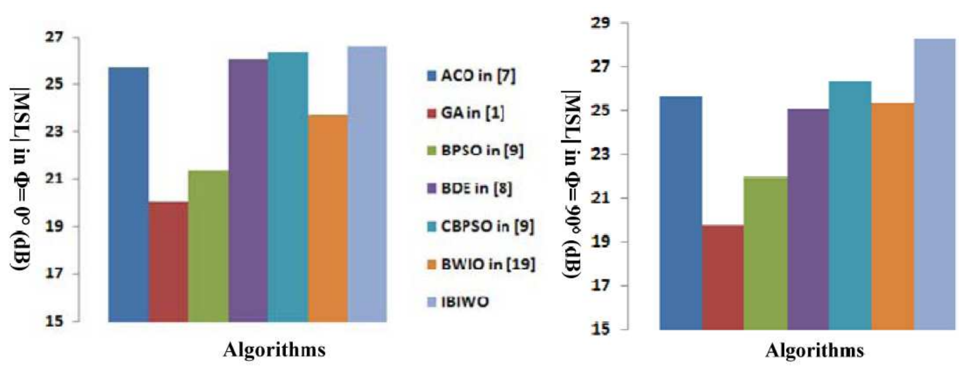


Figure 9. The best element status of 20 × 10 planar array obtained by IBIWO.



(a) (b)

Figure 10. Comparisons of 20 × 10 thinned planar array pattern: (a) $\Phi = 0^\circ$ plane; (b) $\Phi = 90^\circ$ plane.



(a) Φ=0° (b) Φ=90°

Figure 11. Comparisons of |MSL| values obtained by different algorithms: (a) $\Phi = 0^\circ$ plane; (b) $\Phi = 90^\circ$ plane.

know that the convergence speed of IBIWO is faster than the other algorithms in the whole iterations.

From the above results, in the synthesis of thinned linear and planar arrays, it can be clearly observed that the proposed algorithm with adaptive dispersion mechanism can take a good balance between the local search ability and global exploration. Numerical results demonstrate that the IBIWO algorithm is an effective technique for thinned array designs.

Table 3. Comparison of the simulation results.

Simulation results	GA In [1]	ACO In [7]	BDE In [8]	BPSO In [9]	CBPSO In [9]	BIWO In [19]	IBIWO
Average MSLL	20.5	23.1	23.8	19.6	24.3	23.49	24.97
Average convergence iterations	210	184	162	257	206	287	212

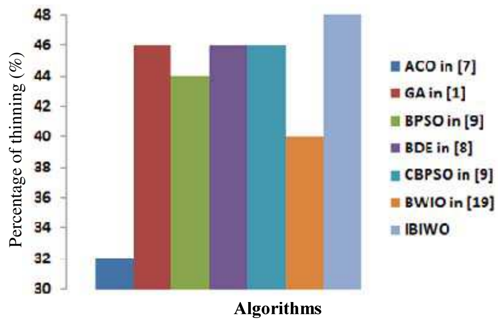


Figure 12. Comparisons of percentage of thinning values obtained by different algorithms.

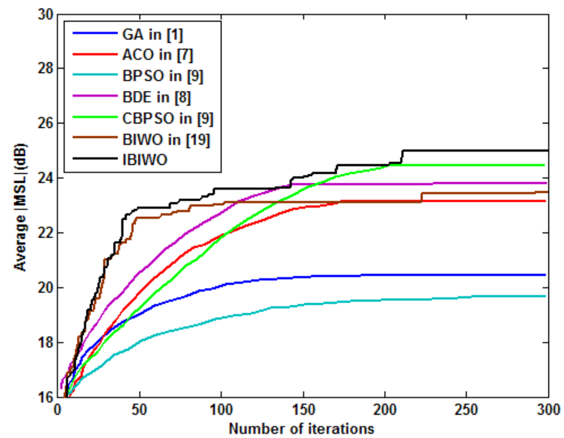


Figure 13. Convergence of the average |MSL| values versus the number of iterations.

5. CONCLUSIONS

This paper introduces the use of a improved binary invasive weed optimization algorithm for thinning periodic linear and planar array to obtain the lowest possible peak side lobe level. An adaptive dispersion mechanism has been adopted to balance between the local search ability and global exploration. A comparison with published results for similar thinned array designs proved that the proposed algorithm achieved the lowest peak side lobe for all considered cases.

REFERENCES

1. Haupt, R. L., "Thinned arrays using genetic algorithms," *IEEE Transactions on Antennas and Propagation*, Vol. 42, No. 7, 993–999, 1994.
2. Wang, X.-K., Y.-C. Jiao, Y. Liu, and Y. Y. Tan, "Synthesis of large planar thinned arrays using IWO-IFT algorithm," *Progress In Electromagnetics Research*, Vol. 136, 29–42, 2013.
3. Bucci, O. M., T. Isernia, and A. F. Morabito, "A deterministic approach to the synthesis of pencil beams through planar thinned arrays," *Progress In Electromagnetics Research*, Vol. 101, 217–230, 2010.
4. Mailloux, R. J. and E. Cohen, "Statistically thinned arrays with quantized element weights," *IEEE Transactions on Antennas and Propagation*, Vol. 39, No. 4, 436–447, 1991.
5. Trucco, A., "Thinning and weighting of large planar arrays by simulated annealing," *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, Vol. 46, No. 2, 347–355, 1999.
6. Jianfeng, Y., et al., "Side lobe reduction in thinned array synthesis using immune algorithm," *Microwave and Millimeter Wave Technology*, 1131–1133, 2008.

7. Quevedo-Teruel, O. S. and E. Rajo-Iglesias, "Ant colony optimization in thinned array synthesis with minimum sidelobe level," *IEEE Antennas and Wireless Propagation Letters*, Vol. 5, No. 1, 349–352, 2006.
8. Zhang, L., et al., "Design of planar thinned arrays using a Boolean differential evolution algorithm," *IET Microwaves, Antennas & Propagation*, Vol. 4, No. 12, 2172–2178, 2010.
9. Wang, W., Q. Feng, and D. Liu, "Synthesis of thinned linear and planar antenna arrays using binary PSO algorithm," *Progress In Electromagnetics Research*, Vol. 127, 371–387, 2012.
10. Oliveri, G., M. Donelli, and A. Massa, "Linear array thinning exploiting almost difference sets," *IEEE Transactions on Antennas and Propagation*, Vol. 57, No. 12, 3800–3812, 2009.
11. Rocca, P., "Large array thinning by means of deterministic binary sequences," *IEEE Antennas and Wireless Propagation Letters*, Vol. 10, 334–337, 2011.
12. Mehrabian, A. R. and C. Lucas, "A novel numerical optimization algorithm inspired from weed colonization," *Ecological Informatics*, Vol. 1, No. 4, 355–366, 2006.
13. Sedighy, S. H., et al., "Optimization of printed Yagi antenna using invasive weed optimization (IWO)," *IEEE Antennas and Wireless Propagation Letters*, Vol. 9, 1275–1278, 2010.
14. Karimkashi, S., A. A. Kishk, and D. Kajfez, "Antenna array optimization using dipole models for MIMO applications," *IEEE Transactions on Antennas and Propagation*, Vol. 59, No. 8, 3112–3116, 2011.
15. Foudazi, A. and A. R. Mallahzadeh, "Pattern synthesis for multi-feed reflector antennas using invasive weed optimisation," *IET Microwaves, Antennas & Propagation*, Vol. 6, No. 14, 1583–1589, 2012.
16. Monavar, F. M., N. Komjani, and P. Mousavi, "Application of invasive weed optimization to design a broadband patch antenna with symmetric radiation pattern," *IEEE Antennas and Wireless Propagation Letters*, Vol. 10, 1369–1372, 2011.
17. Bai, Y.-Y., et al., "A hybrid IWO/PSO algorithm for pattern synthesis of conformal phased arrays," *IEEE Transactions on Antennas and Propagation*, Vol. 61, No. 4, 2328–2332, 2013.
18. Roy, G. G., et al., "Design of non-uniform circular antenna arrays using a modified invasive weed optimization algorithm," *IEEE Transactions on Antennas and Propagation*, Vol. 59, No. 1, 110–118, 2011.
19. Veenhuis, C., "Binary invasive weed optimization," *2010 Second World Congress on Nature and Biologically Inspired Computing (NaBIC)*, 449–454, 2010.
20. Lingnan, Z. S. W. Y., "Invasive weed optimization algorithm of discrete binary version," *Journal of Huazhong University of Science and Technology (Natural Science Edition)*, Vol. 10, 15, 2011.