Pattern Synthesis for Array Antennas based on Interpolation Gravitational Search Algorithm

Cuizhen Sun [1,2], Jun Ding [1], Chenjiang Guo [1], and Jian Liu [2]

[1] School of Electronics and Information
Northwestern Polytechnical University, Xi'an 710072, China
scz@xust.edu.cn

[2] School of Communication and Information Engineering
Xi'an University of Science and Technology, Xi'an 710054, China

Abstract **─** A new algorithm known as interpolation gravitational search algorithm (IGSA) is proposed in this paper when be used to synthesize pattern for array with complicated side lobe and notch. First, a novel and adjustable coefficient *q* for inertia mass is introduced, which can render the particle large in inertia mass get larger and more attractive to other particles to access to more optimal location, so the convergence can be accelerated through varying the discrepancy of inertia mass $M_i(t)$ of particles in a specific population. Second, a simplified quadratic approximation algorithm (SQA) is interpolated that can make the algorithm perform better in the aspect of optimum seeking, so the computational accuracy can be increased through utilizing the stronger local search ability of SQA. To verify the validation of the algorithm, the proposed IGSA is applied to commit pattern synthesis in terms of different targets. Simulation results show that the IGSA, as a whole, is better than the other algorithms the same kind, mainly because the IGSA can be possessed of faster speed in convergence and perform more accurate in optimization.

Index Terms **─** Inertia mass coefficient, interpolation gravitational search algorithm, notches, side-lobe reduction, simplified quadratic approximation.

I. INTRODUCTION

The pattern synthesis for array antenna is a procedure with deliverables of antenna's relevant parameters upon the required radiation. Since the pattern synthesis can greatly simplify design complexity and reduce design cost, it has increasingly become a hot research topic in the field of antenna design and study. The pattern synthesis belongs to the optimization problems. To solve the optimization problem, various meta-heuristic (M-HS) algorithms such as honey bee mating optimization (HBMO) [1], the sailfish optimizer (SFO) [2], the differential evolution algorithm (DE) [3] and the moth-flame optimization algorithm (MFO) [4] have been proposed recently. The genetic algorithm (GA) [5,6], the particle swarm optimization algorithm (PSO) [7,8] have been verified to be able to satisfy in the requirement of the pattern synthesis of array antennas.

The gravitational search algorithm (GSA) [9] is one of the recent M-HS algorithms inspired by the law of gravity. In GSA, a particle is guided by the sum of gravitational force exerted on it by other particles. To search the optimum efficiently, various improvement versions for the original GSA algorithms have been presented, which can be classified into two categories. One focuses on improving the variable parameters [10]; The other is to combine other state-of-the-art algorithms with GSA to enhance GSA [11].While these efforts ameliorate the performance of GSA, GSA is easier to fall into local optimal and become quite lower in the speed of convergence whenever dealing with the problems in the application of pattern synthesis such as the lower side lobe's level and the notches in specific location. Because the specified objective functions which are usually nonlinear, nondifferentiable and even discontinuous with multiple parameters, how to balance the exploration and exploitation according to evolutionary states is challenging.

In this paper, an interpolation gravitational search algorithm (IGSA) is proposed. First, a novel and adjustable inertia mass coefficient *q* is configured which changes the distribution of inertial masses to improve the searching speed to the optimum; Subsequently, to overcome the drawback that GSA is easier falling into local optimal, the simplified quadratic approximation (SQA) is used as a local search operator and embedded into GSA to enhance the entire ability of optimal seeking. Based on different simulation examples, the IGSA can be verified to be better than the traditional GSA both in the convergence rate and in the solution accuracy, which can be proved that the proposed algorithm is more suitable for solving the issue of the complicated synthesis of array antenna such as ultra side

Submitted On: November 12, 2018
Accepted On: May 4, 2019

lobe and notches.

The remainder of this paper is arranged as follows. Section II briefly describes the pattern synthesis framework. In Section III, a detailed introduction of the proposed IGSA is presented. The IGSA is used to simulate standard benchmark functions and design pattern synthesis in Section IV. Finally, Section V concludes this paper.

II. PATTERN SYNTHESIS FOR LINEAR ARRAY ANTENNAS

In terms of an equispaced linear array with N elements, the principle of pattern multiplication governs that the radiation pattern of this array will be equal to the multiplication of the pattern of element with the array factor, therefore, the pattern function normalized is:

$$F(\theta) = 20\log\left[F_e(\theta)^* \left|\sum_{n=1}^{N} I_n e^{jk(n-1)d\sin\theta}\right| / F_{\max}\right], \quad (1)$$

where N is the number of elements, d is the distance between neighboring elements, $k = 2\pi/\lambda$ is the wave number, I_n is the complex excitation of the *nth* element (amplitude and phase); θ is the included angle between the direction of radiation and the axis of array, $F_{\max}$ is the maximum of the pattern function, $F_e(\theta)$ is the pattern of element that represents the radiation feature of the element own, the current excitation amplitude is described as I_n $n=1,2,...,N$,N is the number of elements.

The increasing traffic in the electromagnetic environment prompts to design the antenna array with a lower side lobe level (SLL). A lower SLL is required to avoid the interference with the other systems operating in the same frequency band. In this article, the design target of antenna synthesis is to make the side lobe level become lower than a specified value and create a deep notch in a designated location. The design of fitness function is determined by the design target. Therefore, the fitness function will be set by taking the following two aspects into consideration:

1) The level of side lobe is lower than the expected target about the peak level of side lobe L_{ESL}.
2) Given m directions $\theta_i (i = 1, 2, ..., m)$ in a region of side-lobe, forming notches which is deep in L_{ENL}.
So the fitness function is designed as:

$$f_{fit} = \alpha|L_{MSL} - L_{ESL}| + \beta|L_{MNL} - L_{ENL}|,$$
$$L_{MSL} = \underset{\theta\in P}{MAX}\left\{F(\theta)\right\}, \quad (2)$$
$$L_{MNL} = \underset{\theta_i\in P, i=1,2,\cdots,m}{MAX}\left\{F(\theta_i)\right\},$$

the optimization objective is described as:

$$\min\ f_{fit} = \alpha|L_{MSL} - L_{ESL}| + \beta|L_{MNL} - L_{ENL}|,$$
$$\text{subject to } I_{\min} < I_n < I_{\max}, \quad (3)$$

where L_{MSL} is the maximum level of the side lobe in a real pattern; L_{MNL} is the one within m direction in the side lobe domain, the level of which is the maximum notch; P is the domain of side lobe; α and β are the weights of error, $\alpha + \beta = 1$. The range of excitation current amplitude is $[I_{\min}, I_{\max}]$, the penalty function method is used to deal with the constraint conditions, when the value of I_n exceeds the range, the corresponding objective value is set to the maximum, so that the individual can't be selected into the next generation. To prevent the happening of the fitness value to be null, the issue about minimum specified by the equation closely above is going to be that of the maximum:

$$f_{fit} = \frac{100}{1 + f_{fit}}. \quad (4)$$

III. DESCRIPTION OF INTERPOLATION GRAVITATIONAL SEARCH ALGORITHM

A. The basic principle of GSA

Different from these clustering algorithms like PSO, the particle in GSA is unnecessary to perceive situation in ambient through the factor of environment. On the contrary, the particle can share information through the gravitation applied each other. Therefore, when be influenced without the factor of environment, the particle can commence a search in terms of the environment perceived by overall situation.

In GSA, the initial location of the particle is randomly produced. Given that there are N particles in a space for D dimensional search, the exact location for the i_{th} particle is $x_i = \left\{x_i^1, x_i^2, \cdots x_i^d, \cdots x_i^D\right\}$ $i = 1, 2, \cdots N$, where, x_i^D representing the location of the i_{th} particle in D dimension.

At the moment of t, F_{ij}^d is the gravitation of the *ith* particle exerted by the *jth* counterpart, that is:

$$F_{ij}^d(t) = G(t)\frac{M_{pi}(t) \times M_{aj}(t)}{R_{ij}(t) + \varepsilon}(x_j^d(t) - x_i^d(t)), \quad (5)$$

where, $M_{pi}(t)$ and $M_{aj}(t)$ are the inertia mass of the *ith* and the *jth* particles, respectively. ε is a quite small constant; $G(t)$ is a gravitation constant at the moment t: $G(t) = G_0 e^{-\alpha_0 \frac{t}{T}}$, $R_{ij}(t)$ is the 2-norm between the *ith* and *jth* particles:

$$R_{ij}(t) = \left\|x_i(t), x_j(t)\right\|_2.$$

According to the Newton's second law, the velocity of the *ith* particle at *t* on the *dth* dimension is:

$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_i^d(t), \qquad (6)$$

where, the acceleration is:

$$a_i^d(t) = \frac{\sum_{j=1, j\neq i}^{N} rand_j F_{ij}^d(t)}{M_i(t)}, \qquad (7)$$

$M_i(t)$ is the inertia mass of the *ith* particle. The inertia mass and the value of the fitness will be tough related each other. The inertia mass, as a measure used to scale the value of fitness, is involved into the movement of the particle, it can be revised by following:

$$M_i = \frac{m_i(t)}{\sum_{j=1}^{N} m_j(t)}, \qquad (8)$$

where, $M_{ai} = M_{pi} = M_{ii} = M_i, i = 1, 2, \cdots, N$,

$m_i(t) = \dfrac{fit_i(t) - worst(t)}{best(t) - worst(t)}$, $fit_i(t)$ denotes the value of fitness of the *ith* particle at the moment *t*. Upon the issue of maximum, *best(t)* and *worst(t)* represent the fitness's maximum and minimum value at the moment *t*.

In the process of iteration, the fitness value will be calculated through updating the location of particles. Result will be output subsequently (the location of particle) if either the calculation accuracy is satisfied or the maximal iteration time is reached, otherwise, the iteration should be started again:

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1), \qquad (9)$$

B. The adjustable coefficient of inertia mass

In GSA, the particle's inertia mass is closely related to the value of fitness. Inertia mass, as a measure for scaling the value of fitness, involve the movement of location. If inertia mass is updated by the value of the function of fitness, it can be seen that the larger the inertia mass, the easier the attraction to other particles will be, which, as a result, will move to more optimal location. Therefore, in this paper, a novel and adjustable coefficient of inertia mass *q* is designed to change the discrepancy of particle's inertia mass, which can make particle with large inertia mass larger, while particle with small inertia mass smaller. As a result, the velocity for the algorithm to converge to the maximum can be absolutely improved. When (8) is updated to (10):

$$M_{qi}(t) = \begin{cases} (1+q)^{\beta_0} M_i(t) & fit_i(t) > f_{AVG}(t) \\ (1-q)^{\beta_0} M_i(t) & fit_i(t) < f_{AVG}(t) \end{cases}, \qquad (10)$$

$$q = \begin{cases} (\gamma_{max} - \gamma_{min}) \dfrac{fit_i(t) - f_{AVG}(t)}{best(t) - f_{AVG}(t)} \\ \qquad fit_i(t) > f_{AVG}(t) \\ (\gamma_{max} + \gamma_{min}) \dfrac{f_{AVG}(t) - fit_i(t)}{f_{AVG}(t) - worst(t)} \\ \qquad fit_i(t) < f_{AVG}(t) \end{cases}, \qquad (11)$$

where, $M_{qi}(t)$ is the inertia mass of the *ith* particle at the *t* moment after adjustment, *q* is the adjustment factor of the inertia mass, $f_{AVG}(t)$ represents the mean value of the fitness at the moment *t*. $\gamma_{\min}$ and $\gamma_{\max}$ are the zoom factors, controlling the size and the change of the adjustable factor of inertia mass that is usually set by one just between 1 and 0. The value of the exponential weight β_0 can be changed further to adjust the distribution of inertia mass. In this paper, $\gamma_{min} = 0.1$, $\gamma_{max} = 0.7$, $\beta_0 = 3$. When $fit_i(t)$, the value of the inertia mass of the *ith* particle is equal to $f_{AVG}(t)$, the average value of all the particles at the time *t*, the inertia mass is unchanged, $M_{qi}(t) = M_i(t)$. Compared with the basic GSA, the size of *q* can be used to change the inertia mass of particle, making the discrepancy of particle's inertia mass increase, which, as a result, will accelerate the speed to approach the optimal location, the velocity of convergence can thus be increased.

C. The SQA algorithm

As a simplified three point quadratic approximation, SQA is a simple, directive and efficient method for searching. It needn't the message of derivative anymore, three points are enough for model delivery. Since the message of the objective function solved can be effectively utilized, the amount of calculation is quite small to bring more convenient to solve the issue of optimization. A combination method between SQA and DE is reported in terms of the problem of constraint optimization [12].After being optimized with respect to test function, it can be concluded that, compared with the original DE, this algorithm is more superior. This paper attempts to interpolate SQA into GSA for improving the algorithm's overall performance.

In terms of the issue of maximization, three optimal individuals x_a, x_b, x_c are provided. In which,

$$x_a = [x_a^1, x_a^2, \cdots, x_a^D]^T, \; x_b = [x_b^1, x_b^2, \cdots, x_b^D]^T,$$
$$x_c = [x_c^1, x_c^2, \cdots, x_c^D]^T.$$

The fitness function are $f^a = f_{fit}(x_a)$, $f^b = f_{fit}(x_b)$, $f^c = f_{fit}(x_c)$, where $f^a > f^b > f^c$. The approximation points achieved by SQA are:

$$x_w = [x_w^1, x_w^2, \cdots, x_w^D]^T,$$
$$x_w^i = \frac{1}{2}\frac{M_i}{N_i} \quad i = 1, 2, ..., D, \tag{12}$$
$$M_i = [(x_a^i)^2 - (x_c^i)^2]f^b + [(x_c^i)^2 - (x_b^i)^2]f^a + [(x_b^i)^2 - (x_a^i)^2]f^c,$$
$$N_i = (x_a^i - x_c^i)f^b + (x_c^i - x_b^i)f^a + (x_b^i - x_a^i)f^c.$$

The interpolation of SQA into basic GSA can improve the problem that the algorithm is easy to fall into local optimal and can increase the velocity for particle to approximate the best.

D. The procedure of IGSA

The IGSA can be realized by following procedures:

Step 1: Population Initiation

Step 2: Gravitation calculation

(1) Particle's fitness value is calculated by equation (4), updated by *G(t)*, $f_{AVG}(t)$, *best(t)*, *worst(t)*, simultaneously.

(2) Updating particle's inertia mass by utilizing well-designed inertia mass coefficient *q* according to equation (11). Updating the total gravitation from all directions by equation (5).

Step 3: Updating particle's velocity by equation (6).

Step 4: Updating particle's location, the location is calculated by equation (9), updating the value of fitness. The result will be output (the location of particle) if either the calculation accuracy is satisfied or the maximal iteration times are reached, otherwise, turn to Step 2.

Step 5: Interpolation into SQA

(1) Calculating f^a, f^b, f^c, selecting three optimal particles x_a, x_b, x_c to calculate fitness from Step 4.

(2) Approximation point determination. If $N_i = 0$, the Step 5 is omitted, otherwise the approximating point x_w is calculated according to (12). In the meanwhile, fitness $f_{fit}(x_w)$ is calculated.

(3) Substitution. If $f_{fit}(x_w) > f^a$, x_w is used to replace the optimal x_a in old population. If $f^a > f_{fit}(x_w) > f^b$, x_w is used to replace the worst case there; otherwise Step 5 is omitted.

Step 6: Stop Judgment. If meeting stop criteria, the algorithm is terminated, the optimal result is output, otherwise turn to Step 2.

IV. RESULT ANALYSIS

A. Standard benchmark functions

In this section, IGSA is compared with the MFO [4], PSOGSA [11], GSA, PSO, GA algorithms. The parameters of GSA and IGSA are set by the same as these in [9]. In benchmark functions, dimension is 30(n=30) and maximal iteration is 1000(T=1000). The minimum value (f_{opt}) of the benchmark functions are zero, except for F_4 which has a minimum value of -12569.487 (-418.9829*n, n=30).

The results are averaged over 30 runs and the average best-so-far solution and median of the best solution in the last iteration are given in Table 1:

(1) $F_1(X) = \sum_{i=1}^{n} x_i^2 \qquad x_i \in [-100, 100]^n$,

(2) $F_2(X) = \max_i\{|x_i|, 1 \le i \le n\} \qquad x_i \in [-100, 100]^n$,

(3) $F_3(X) = \sum_{i=1}^{n}([x_i + 0.5])^2 \qquad x_i \in [-100, 100]^n$,

(4) $F_4(X) = \sum_{i=1}^{n} -x_i \sin(\sqrt{|x_i|}) \qquad x_i \in [-500, 500]^n$,

(5) $F_5(X) = -20\exp(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}) - \exp(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)) + 20 + e \qquad x_i \in [-32, 32]^n$,

(6) $F_6(X) = \frac{\pi}{n}\{10\sin(\pi y_1) + \sum_{i=1}^{n-1}(y_i - 1)^2[1 + 10\sin^2(\pi y_{i+1})] + (y_n - 1)^2\} + \sum_{i=1}^{n} u(x_i, 10, 100, 4), \quad y_i = 1 + \frac{x_i + 1}{4}$.

$$u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases} \qquad x_i \in [-50, 50]^n$$

Table 1: Minimization results of benchmark functions with the average best-so-far solution and median of the best solution

	IGSA	PSOGSA	MFO	RGA	PSO	GSA
F_1	1.69×10^{-18}	5.31×10^{-16}	1.49×10^{-15}	23.13	$1.8*10^{-3}$	7.3×10^{-11}
	1.68×10^{-18}	5.04×10^{-16}	1.32×10^{-15}	21.87	$1.2*10^{-3}$	7.1×10^{-11}
F_2	7.38×10^{-10}	3.65×10^{-8}	2.05×10^{-11}	11.78	8.1	3.7×10^{-6}
	9.30×10^{-10}	3.57×10^{-8}	1.97×10^{-11}	11.94	7.4	3.7×10^{-6}
F_3	2.78×10^{-18}	4.39×10^{-15}	6.23×10^{-12}	24.01	$1.0*10^{-3}$	8.3×10^{-11}
	4.18×10^{-18}	3.86×10^{-15}	5.72×10^{-12}	24.55	6.6×10^{-3}	$7.7*10^{-11}$
F_4	$-1.04\times10^{+4}$	$-6.83\times10^{+4}$	$-7.68\times10^{+4}$	$-1.2\times10^{+4}$	$-9.8*10^{+3}$	$-2.8\times10^{+3}$
	$-1.45\times10^{+4}$	$-6.95\times10^{+4}$	$-7.41\times10^{+4}$	$-1.2\times10^{+4}$	$-9.8*10^{+3}$	$-2.6\times10^{+3}$
F_5	1.023×10^{-9}	2.145×10^{-9}	7.1780×10^{-7}	2.13	9.0×10^{-3}	6.9×10^{-6}
	1.023×10^{-9}	2.150×10^{-9}	7.1780×10^{-7}	2.16	6.0×10^{-3}	6.9×10^{-6}
F_6	2.72×10^{-19}	6.59×10^{-9}	1.49×10^{-7}	0.051	0.29	0.01
	1.89×10^{-19}	4.98×10^{-9}	1.36×10^{-7}	0.039	0.11	$4.2*10^{-13}$

Statistically speaking, for the average value of 30 runs, IGSA reaches best in all of the benchmark functions except F_2 for which the average best-so-far solution and median of the best solution based on MFO is better. The number of functions which the MFO performs better is inferior to PSOGSA and superior to GSA. The MFO and PSOGSA can balance the exploration and exploration well than the GA and PSO algorithms. From the Table 1, it can be concluded that IGSA performs best on the test functions.

Table 2: Results comparison of ultra low side lobe synthesis for the six algorithms

Results of Ultra Low Side Lobe Synthesis			
Algorithms	Directivity	Peak Side-Lobe Level (dB)	Width of Main-Lobe (°)
GA	15.1579	-37.8756	0.8
PSO	15.2763	-39.1364	0.8
GSA	15.3204	-40.5069	0.6
MFO	15.2987	-40.7952	0.6
PSOGSA	15.3462	-41.0551	0.6
IGSA	15.4543	-42.0317	0.6

The linear antenna arrays (LAA) are widely used in many high-performance radio systems like radar, sonar, air and space navigation, underground propagation etc. In this paper, we consider a central symmetry linear array with N=20, $d = \lambda/2$. The current excitation amplitudes are optimized for the synthesis of LAA keeping the excitation phase as zero. So the optimization parameters are 10 amplitudes, the amplitude range of I_n is [0.1,1], $n=1,2,\ldots,N$, N is the number of elements.

B. The pattern synthesis for ultra low side lobe

In this section, the design objective is that beam width of the main lobe is 20°, and the side lobe levels are lower than -42dB. So, in equations (1), (3), $F_e(\theta) = \sin\theta$, L_{ESL}=-42dB, $\alpha = 1, \beta = 0$.

The IGSA is used compared with the GA, PSO, standardized GSA, PSOGSA, MFO to conduct a specific optimization, the iterative times for them are 1000; the number of population for GSA and IGSA are 100, $G_0 = 10$ $\alpha_0 = 5$. Simulation results averaged over 30 runs are depicted in Figs. 1-4 and Table 2.

Figure 1 shows the patterns of ultra low side lobe for the six algorithms. It is clear that the pattern obtained by IGSA meets the desired objective very well. Figure 2 shows the average evolution curves of these six algorithms when they run 30 times. It can be seen that the IGSA performs better in convergence speed and computer accuracy. The best convergence profile, worst convergence profile, average profile for IGSA over 30 runs are depicted in Fig. 3. Figure 4 is the optimized current amplitude value. Table 2 is the detailed comparisons of the above algorithms. PSOGSA and MFO perform similar and only inferior to IGSA. The GA and PSO have no advantages. It can be noted that under the same width of main lobe, GSA is better than MFO dealing with directivity, while it is weak in peak side lobe level. When the width of main lobe is almost unchanged, the IGSA not only performs most low peak side lobe (-42.0317), but also has the best directivity value (15.4543). It can conclude that the IGSA outperforms other algorithms according to the directivity, the peak side lobe level and width of main lobe.

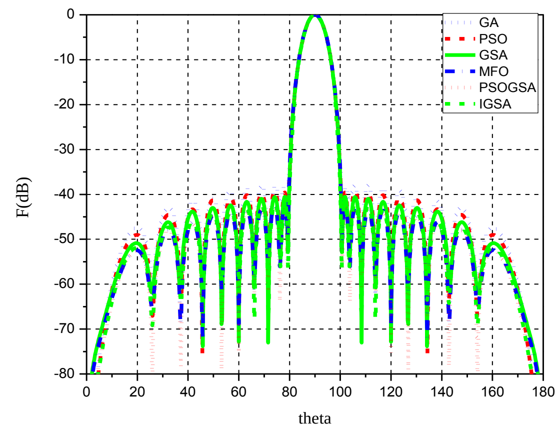

Fig. 1. Patterns of ultra low side lobe for the six algorithms.

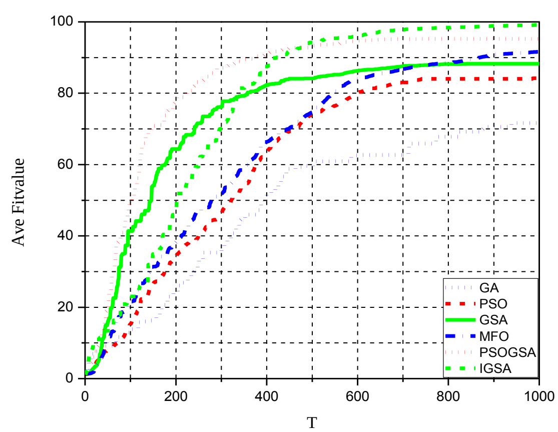

Fig. 2. Average convergence based on six algorithms for the pattern synthesis of ultra low side lobe.

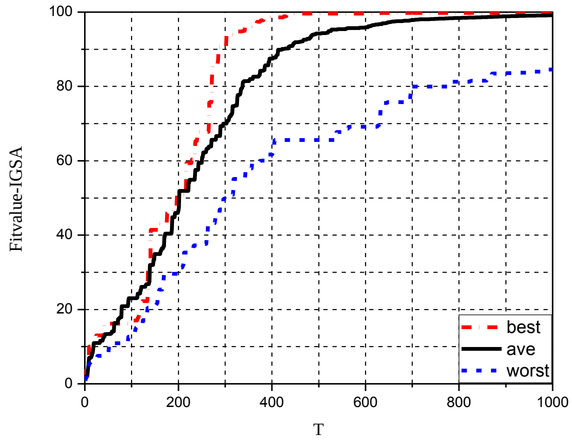

Fig. 3. Comparison of different convergence curves based on IGSA.

C. The pattern synthesis for low side lobe with notches

In this section, the optimizing objective is that the side lobe level is required not exceeding -29dB and between 50°and 60°, a notch with -70dB is required to be created. In equation (1), $F_e(\theta)=1$. In equation (3), $L_{ESL}=-29dB$, $L_{ENL}=-70dB$, $\alpha=0.65$, $\beta=0.35$. The parameters are same as that in Section B. The optimal results averaged over 30 runs are shown in Figs. 5-8 and Table 3.

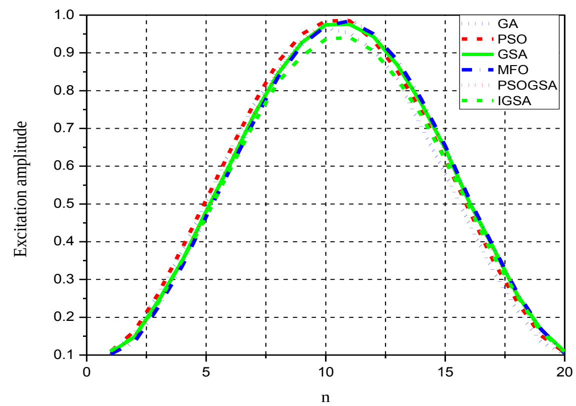

Fig. 4. The optimized current excitation amplitude for the pattern synthesis of ultra low side lobe.

Table 3: Results comparison of low side lobe and notches for the six algorithms

Results of Low Sidelobe and Notches			
Algorithms	Peak Side Lobe Level (dB)	Peak Level of Notch (dB)	Width of Main Lobe (°)
GA	-27.7897	-65.2736	1.4
PSO	-28.4271	-67.4101	1.6
GSA	-29.0359	-70.3720	1.4
PSOGSA	-29.1577	-70.3410	1.2
MFO	-29.6085	-71.0416	1.4
IGSA	-29.6170	-71.3873	1.2

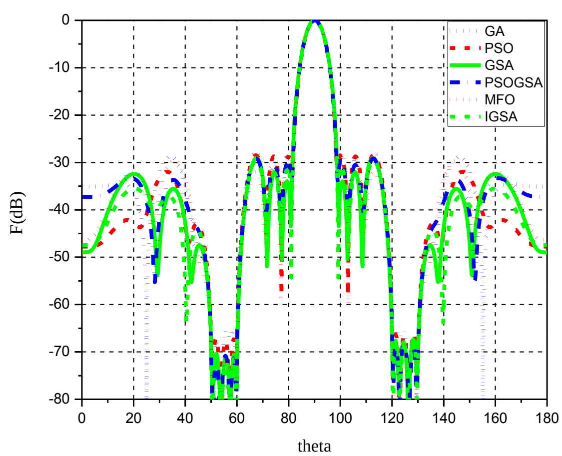

Fig. 5. Patterns of low side lobe with notches based on six algorithms.

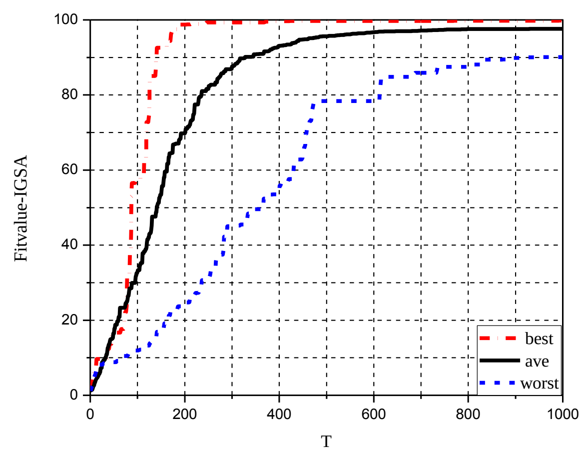

Fig. 6. Comparison of different convergence curves based on the IGSA algorithm.

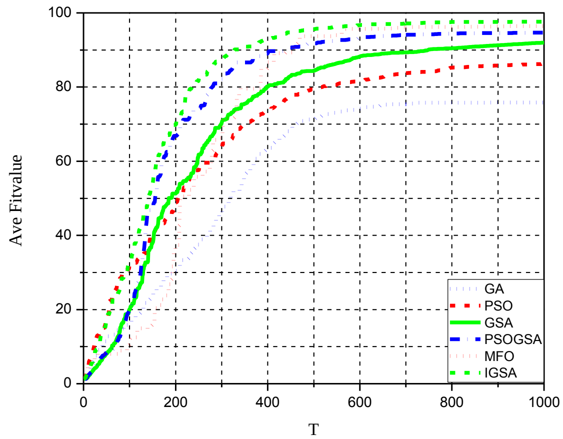

Fig. 7. Average convergence based on six algorithms for the pattern synthesis of low side lobe with notches.

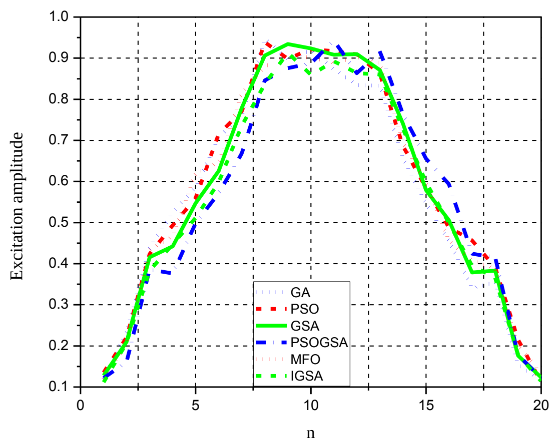

Fig. 8. The optimized current excitation amplitude for the pattern synthesis of low side-lobe with notches.

The patterns obtained by these six algorithms after 1000 iterations are shown in Fig. 5. The best convergence profile, worst convergence profile, average profile for IGSA over 30 runs are depicted in Fig. 6. It can be seen that the peak side lobe level and notches value obtained by IGSA is lower than the result obtained by GSA and other algorithms. Figure 7 shows the average evolution curves after 30 runs. It can be seen that IGSA outperforms other algorithms in evolutionary speed and accuracy. It is worth noting that the problem of trapping in local optimum of GSA is improved efficiently by using IGSA. The further comparison of IGSA with GSA and other four algorithms are shown in Table 3.

The notch depth of PSOGSA is relatively poor than that of GSA and MFO, but the peak side lobe level is better than GSA. It is worth noting that MFO performs better than GSA and PSOGSA, when dealing with the peak side lobe level, MFO is even comparable to that of IGSA. When the width of main lobe is narrower than other algorithms, the IGSA also performs better in peak side lobe and notch value. It can conclude that IGSA outperforms other algorithms. Figure 8 is the optimized current value. The optimization results meet the current amplitude requirements in the constraint conditions.

V. CONCLUSION

In terms of the drawbacks of the basic GSA that is slow in convergence and fall easily into local optimal when synthesizing patterns to the array with low side lobe and notch, IGSA is proposed in this paper. This algorithm can realize the balance between the overall convergence and the local convergence to improve convergence speed through introducing new inertia mass adjustable factor q. This algorithm can also be enhanced to foster local search through interpolating the SQA. In IGSA, the SQA is used as an algorithm with high power in local searching, and the GSA is used to commit search overall. The simulation results in the final show that the proposed IGSA, compared with the same kind of algorithm under the same situation, can be endorsed with faster convergence speed, while keeping the smaller width of the main lobe and the lower level of side lobe. As a conclusion, the proposed algorithm is really more suitable for the pattern synthesis of the complicated array antenna, IGSA can balance between the exploration and the exploitation to reach the optimum quickly and accurately.

REFERENCES

[1] F Günes, S. Demirel, and P. Mahouti, "A simple and efficient honey bee mating optimization approach to performance characterization of a microwave transistor for the maximum power delivery and required noise," *International Journal of Numerical Modelling-Electronic Networks Devices and Fields: Electronic Networks, Devices and Fields*, vol. 1, pp. 4-20, 2015.

[2] S. Shadravan, H. R. Naji, and V. K. Bardsiri, "The sailfish optimizer: A novel nature-inspired metaheuristic algorithm for solving constrained engineering optimization problems," *Engineering Applications of Artificial Intelligence*, vol. 80, pp. 20-34, 2019.

[3] F. Güne, M. A. Belen, and P. Mahouti. "Competitive evolutionary algorithms for building performance database of a microwave transistor," *International Journal of Circuit Theory and Applications*, vol. 4, pp. 1-15, 2017.

[4] A. Das, D. Mandal, S. P. Ghoshal, and R. Kar, "Moth flame optimization based design of linear and circular antenna array for side lobe reduction," *Int. J. Numer. Model.*, 32:e2486, 2019.

[5] O.-B. Pinar, "GA optimization of the optical directional coupler," *ACES Journal*, vol. 32, pp. 1136-1139, 2017.

[6] G. Oliveri and A. Massa, "Genetic algorithm

(GA)-enhanced almost difference set (ADS)-based approach for array thinning," *IET Microwaves Antennas & Propagation*, vol. 5, no. 3, pp. 305-315, 2011.

[7] W. Wang and Q. Feng, "Application of PSO algorithm in pattern synthesis," *Journal of Xidian University*, vol. 38, no. 3, pp. 175-180, 2011.

[8] O. Kilic and Q. M. Nguyen. "Enhanced artificial immune system algorithm and its comparison to bio-inspired optimization techniques for electro-magnetics applications," *ACES Journal*, vol. 33, no. 2, pp. 132-135, 2018.

[9] R. Esmat, N. Hossein, and S. Saeid, "GSA: A gravitational search algorithm," *Information Sciences*, vol. 179, no. 13, pp. 2232-2248, 2009.

[10] A. Z. Zhang, G. Y. Sun, J. C. Ren, X. D. Li, Z. J. Wang, and X. P. Jia, "A dynamic neighborhood learning based gravitational search algorithm," *IEEE Trans. on Cyb.*, vol. 48, no. 1, pp. 436-447, 2018.

[11] S. Mirjalili and S. Z. M. Hashim, "A new hybrid PSOGSA algorithm for function optimization," *International Conference on Computer and Information Application*, pp. 374-377, 2010.

[12] H. Li, Y. C. Jiao, and L. Zhang, "Hybrid differential evolution with a simplified quadratic approximation for constrained optimization problems," *Engineering Optimization*, vol. 43, no. 2, pp. 115-134, 2011.

Cuizhen Sun was born in 1981. She is a Ph.D. candidate in the Northwestern Polytechnical University. She is also a Teacher in Xi'an University of Science and Technology. Her current research interests are antenna arrays and optimization algorithms in antenna design.