

基于改进整型遗传算法的稀疏矩形平面阵列优化*

国强,王亚妮,袁鼎,戚连刚

(哈尔滨工程大学信息与通信工程学院,黑龙江哈尔滨 150001)

摘要:为了降低固定稀疏率、固定孔径的稀疏矩形阵列的峰值旁瓣电平,提出一种改进整型遗传算法。该算法在整型遗传算法的基础上,提出了等间隔采样的交叉策略、多点变异策略以及优良基因重组的策略。采取等间隔采样的基因交叉方式,可以有效发挥整型编码的优势,从而提高算法的运行效率;为了提高种群的多样性,防止算法陷入局部最优,采用了多点变异策略;采用优良基因重组技术,加快了算法的收敛速度。仿真结果表明,相比传统的二进制和实数编码,整型编码更为直接高效;与用于稀疏矩形阵列优化的相关算法相比,本文所提算法获得了更优的旁瓣电平,证实了算法的有效性和优越性。

关键词:矩形平面阵列;峰值旁瓣电平;整型遗传算法;稀疏阵列优化

中图分类号:TN95 文献标志码:A 文章编号:1001-2486(2023)02-105-07

Optimization of sparse rectangular planar array using modified integer genetic algorithm

GUO Qiang, WANG Yani, YUAN Ding, QI Lian'gang

(College of Information and Communication Engineering, Harbin Engineering University, Harbin 150001, China)

Abstract: In order to reduce the peak sidelobe level of sparse rectangular array with fixed sparse ratio and fixed aperture, a modified integer genetic algorithm was proposed. On the basis of the integer genetic algorithm, the crossover strategy of equal interval sampling, multi-point mutation strategy and excellent gene recombination strategy were proposed. The crossover strategy of equal interval sampling can effectively exert the advantages of integer coding, which improves the operation efficiency of the algorithm. In order to improve the diversity of the population and avoid falling into the local optimum, the multi-point mutation strategy was adopted. The excellent gene recombination technology was used to accelerate the convergence speed of the algorithm. Simulation results show that, compared with the traditional binary and real coding, the integer coding is more direct and efficient; compared with the related algorithms for sparse rectangular array optimization, the proposed algorithm obtains the better sidelobe level, which proves the effectiveness and superiority of the algorithm.

Keywords: rectangular planar array; peak sidelobe level; integer genetic algorithm; sparse array optimization

天线是一种在各种无线系统中实现电磁波传输和接收的设备,随着天线在实际工程中应用的普及,人们逐渐意识到了单一天线具有不可避免的缺陷,例如方向性弱及增益较低等^[1]。天线阵列由几个相同的天线单元定向排列并以某种方式馈电构成。为了方便实用,这些天线单元通常具有相同的结构和尺寸。如果阵元排列在直线或平面上,则称为线阵或平面阵。如果阵元分布与载体表面形状相同,则称为共形阵列。为了获得特定的天线方向图,通常对阵列单元的位置或激励进行加权。阵列综合^[2]主要是通过合理规划阵列单元的位置、激励和相位以得到期望的一些指标,比如峰值旁瓣电平(peak sidelobe level,

PSLL)、主瓣宽度、方向性系数等,从而提高天线性能。当阵元均匀分布在直线、平面或者空间里时,“舍弃”一些阵元,此时得到的天线阵列依然具有良好的方向特性,这样的阵列被称为稀疏阵列^[2]。

平面稀疏阵列通常按栅格形式分布,因此又被称为栅格阵列。由于计算机计算性能的限制,早期对稀疏阵列的研究方法不仅耗时而且十分低效。早在20世纪60年代,基于统计学规律的阵列布阵方法就被提出,通过调整均匀满阵下权系数的幅度分布来优化阵列结构,最终实现了不等间距阵列天线布阵。后来,学者们为了探究全局最优解,采用了穷举法对稀疏阵列进行研究^[3]。

* 收稿日期:2021-03-24

基金项目:国家重点研发计划资助项目(2018YFE0206500);国家自然科学基金资助项目(62071140)

作者简介:国强(1972—),男,黑龙江哈尔滨人,教授,博士,博士生导师,E-mail:guoqiang@hrbeu.edu.cn;

戚连刚(通信作者),男,河北邢台人,讲师,博士,硕士生导师,E-mail:qiliangang@hrbeu.edu.cn

使用穷举法进行稀疏布阵,需要遍历所有情况的阵元分布并计算其方向图,显然这样的方式只能对阵元数极少的稀疏阵列进行稀疏。但随着阵列天线的飞速发展,需要计算的阵元数也越来越多,穷举法不再有效。伴随着计算机性能的提高,傅里叶变换类算法^[4]、智能优化类算法^[5]、矩阵束^[6]、压缩感知^[7]等方法被广泛地应用到阵列优化中。

其中智能优化类算法具有容易摆脱局部最优的能力,所以在阵列优化问题中应用最为广泛。常用的智能优化类算法包括遗传算法^[8]、粒子群算法^[9]、布谷鸟算法^[10]等。这些算法基本都是采用实数编码或者二进制编码,实数编码不太适用于稀疏阵列,二进制编码虽然转换成整数编码,但是相比整型编码不够直接。对于稀疏阵列而言,遗传算法由于其灵活的编码特性及多样的变异过程具有天然的优势。

鉴于此,本文针对平面矩形栅格阵列稀疏优化问题提出了一种改进整型遗传算法(modified integer genetic algorithm, MIGA),以抑制 PSLL 为目标,构造了相应的适应度函数。优化结果给出了稀疏矩形阵列在平面内的最优阵元分布。

1 天线阵列模型

设矩形稀疏阵列拥有 $2M \times 2N$ 个阵元,阵列孔径为 $2L \times 2H$,每个阵元在 x 轴方向上的间距为 d_x ,在 y 轴方向上的间距为 d_y 。以阵列中心为坐标原点建立空间直角坐标系时,阵列模型如图 1 所示。当阵元围绕着阵列中心对称分布时,只需对一个象限的阵元位置进行优化,此时需要优化的变量空间约为原来的四分之一,否则在计算天线阵因子时需要计算考虑每个阵元对天线方向图的影响。设每个阵元的坐标为 (x_m, y_n) , $1 \leq m \leq M, 1 \leq n \leq N$,当栅格点上的阵元需要被“舍弃”时,即代表阵元开路或接负载匹配, $I_{mn} = 0$;当阵元需要被“保留”时,即代表阵元需要接激励源, $I_{mn} = 1$ 。设阵元均为理想点源,且阵元的激励幅度相等,同时忽略阵元之间的互耦效应。由于天线阵列孔径的限制,在角落的阵元需要被保留。因此,矩形稀疏阵列的阵因子可以表示如下:

$$F(u, v) = \sum_{m=1}^M \sum_{n=1}^N I_{mn} e^{jk(x_m u + y_n v)} \quad (1)$$

式中: $k = 2\pi/\lambda$ 为波数, λ 为阵列工作波长; $u = \sin\theta\cos\varphi$, $v = \sin\theta\sin\varphi$, θ 、 φ 为俯仰角和方位角; I_{mn} 代表第 (m, n) 个阵元的激励。类似稀疏直线阵,低副瓣的稀疏平面阵也有着内密外疏的性质,越

靠近阵列中心的阵元分布越紧密,远离阵列中心的阵元分布越离散。为了便于优化,可以将矩形阵列上的阵元进行降维,即将二维矩形阵转换成一维向量。

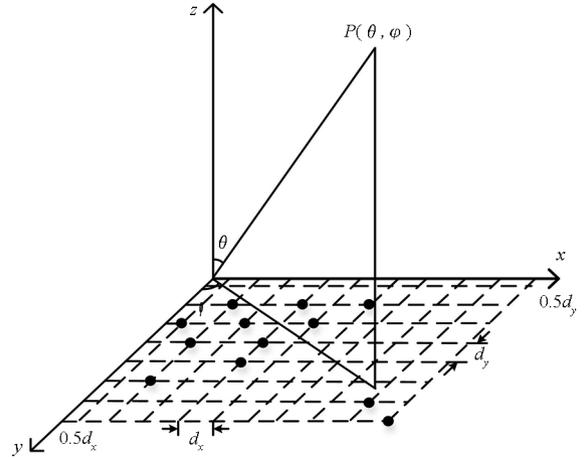


图 1 稀疏矩形阵列模型

Fig. 1 Sparse rectangular array model

为了使天线阵列的方向图在整个平面内的 PSLL 较低,适应度函数 F_1 应设置如下:

$$F_1 = \max_{(u,v) \in S} \left(20 \lg \left| \frac{F(u,v)}{F_{\max}} \right| \right) \quad (2)$$

其中, S 为平面矩形阵列方向图除主瓣外的副瓣区域。

为了使方向图在 $\varphi = 0^\circ$ 和 $\varphi = 90^\circ$ 两个平面内对应的旁瓣电平较低,适应度函数 F_2 应设置为:

$$F_2 = \max \{ SLL(\varphi = 0^\circ) \} + \max \{ SLL(\varphi = 90^\circ) \} \quad (3)$$

2 改进整型遗传算法

对于稀疏天线阵列,整型遗传算法采用整型编码的方式对阵元位置进行编码,直接高效。由于稀疏天线阵的阵元是在栅格点上随机分布的,采用整型编码不仅可以有效减少优化时的变量空间,避免了采用实数编码或者二进制编码而导致出现不可行解的问题,并且可以有效控制天线阵列的稀疏率。

2.1 算法步骤

2.1.1 种群初始化

当矩形阵列的稀疏率 f 大于 50% 时,应该将保留阵元的位置作为优化目标,从而减少优化变量个数,反之,则需要将需要扣除的阵元位置作为优化目标。当稀疏率小于 50% 时,优化问题的维度 T 可以由式(4)求得。

$$T = \begin{cases} M \times N \times f & \text{对称阵列} \\ 2M \times 2N \times f & \text{非对称阵列} \end{cases} \quad (4)$$

设定种群数为 N_p , 初始化的种群矩阵 \mathbf{P}_{init} 如下:

$$\mathbf{P}_{init} = \begin{bmatrix} \alpha_{1,1} & \alpha_{1,2} & \cdots & \alpha_{1,T} \\ \alpha_{2,1} & \alpha_{2,2} & \cdots & \alpha_{2,T} \\ \vdots & \vdots & & \vdots \\ \alpha_{N_p,1} & \alpha_{N_p,2} & \cdots & \alpha_{N_p,T} \end{bmatrix} \quad (5)$$

其中, $\alpha_{i,j} \in \{1, 2, \dots, T\}$ ($1 \leq i \leq N_p, 1 \leq j \leq T$), 每一行是不重复的小于等于 T 的正整型。同时将 \mathbf{P}_{init} 按行从小到大、从左到右排序, 方便后续的计算。阵元位置编号与实际位置的对应关系并不是唯一的, 给定时需要优先考虑计算复杂度。阵元的位置坐标可以采用复数矩阵, 横坐标 x_m 作为实部, 纵坐标 y_n 作为虚部, 即 $P_{mn} = x_m + iy_n$, 因此对应的坐标矩阵 \mathbf{P} 应该设置为三维矩阵, 矩阵规模为 $2M \times 2N \times N_p$ 。

2.1.2 基因交叉

计算每个个体的适应度, 然后将个体按照适应度值从大到小排序, 保留一半较好的个体, 得到的种群为 \mathbf{P}_{old} 。将旧种群中所有元素打乱, 并从小到大排序, 便会得到一个行向量 $\mathbf{g} = [\gamma_1, \gamma_2, \dots, \gamma_{N_p \times T/2}]$, 然后按照一定间隔 $\Delta z = N_p/2$ 进行采样, 形成新的子代个体, 从而生成新的子代种群如下:

$$\mathbf{P}_{new} = \begin{bmatrix} \gamma_{1,1} & \gamma_{1,2} & \cdots & \gamma_{1,T} \\ \gamma_{2,1} & \gamma_{2,2} & \cdots & \gamma_{2,T} \\ \vdots & \vdots & & \vdots \\ \gamma_{\Delta z,1} & \gamma_{\Delta z,2} & \cdots & \gamma_{\Delta z,T} \end{bmatrix} \quad (6)$$

2.1.3 多点变异

定义个体的相似度 s 为:

$$s = \sum_{j=1}^T \sum_{i=1}^{\Delta z} \left| \gamma_{ij} - 2a \sum_{i=1}^{\Delta z} \gamma_{ij} / N_p \right| \quad (7)$$

其中, a 是一个常数, 可以调节相似度的大小, 本文设定 $a = 2$ 。设定算法迭代次数为 K , 通过相似度得到一种新的变异概率, 会随着代数变化, 如式(8)所示。

$$P_i = \begin{cases} 1 - \frac{i}{K} \cdot \frac{s_i}{s_1} & s_i < s_1 \\ \frac{i}{K} \cdot \frac{s_1}{s_i} & s_i \geq s_1 \end{cases} \quad (8)$$

其中, s_1 是初始子代的相似度值, s_i 是第 i 次迭代的相似度值。

通过比较每个子代的变异概率是否大于 s_1 (s_1 为 $[0, 1]$ 中的随机数), 决定是否变异。这里采取了一种新的变异方式, 考虑到自然界中基因变异的不确定性, 单点变异的方式过于单一, 因此

采用多点变异, 增强变异的随机性。变异的基因个数一般很少, 所以限定变异的基因个数不超过整个基因组的 2%。由于传统的变异方式无法保证变异之后编码位置不重复, 本文采用与互补空间交换的变异方式, 既可以保证变异之后个体的不重复性, 也可以加快运算速度。定义全集 $H = \{\gamma \mid 1 \leq \gamma \leq T, \gamma \in \mathbf{Z}^+\}$, \mathbf{P}_{new} 中的个体 $B = \{\gamma_1, \gamma_2, \dots, \gamma_T\}$ 对应的补集为 $B_{mut} = \{\gamma \mid \gamma \in H, \gamma \neq B\}$ 。随机产生变异的基因数和变异的基因位置, 然后和补空间中随机位置的元素相互交换, 便得到了变异之后的基因, 同时对种群 \mathbf{P}_{new} 进行更新。

2.1.4 基因重组

采用优良基因重组技术^[11], 将 \mathbf{P}_{old} 和 \mathbf{P}_{new} 合并成一个 $N_p \times T$ 的种群, 即新的初始种群 \mathbf{P}_{init} , 重复上述步骤。

2.2 算法伪代码

改进整型遗传算法的步骤如算法 1 所示。

算法 1 改进整型遗传算法

Alg. 1 Modified integer genetic algorithm

1. 参数初始化: 种群数 N_p , 满阵阵元数 $2M \times 2N$, 迭代次数 K
2. 由式(4)计算优化变量空间的维度 T , 生成初始种群矩阵 \mathbf{P}_{init}
3. **while** $i_k < K$ **do**
4. **for** $i = 1$ to N_p **do**
5. 将个体转换成坐标, 并根据式(2)或式(3)计算初始种群中每个个体的适应度值, 并进行排序
6. **end for**
7. 保留一半适应度更高的个体生成旧种群 \mathbf{P}_{old}
8. **for** $iter = 1$ to $N_p/2$ **do**
9. 从旧个体中通过间隔采样的方式生成新种群矩阵 \mathbf{P}_{new}
10. 根据式(7)计算 \mathbf{P}_{new} 个体之间的相似度, 根据式(8)得到变异概率 P_i
11. **if** $P_i \geq \text{rand}(0, 1)$ **then**
12. 进行多点变异, 从互补空间中产生变异基因, 得到新种群矩阵 \mathbf{P}_{new}
13. **end if**
14. **end for**
15. 将 \mathbf{P}_{old} 和 \mathbf{P}_{new} 合并成新的种群, 更新 \mathbf{P}_{init}
16. 更新迭代次数
17. **end while**
18. 输出全局最优个体, 转换成阵元位置坐标

3 实验结果及分析

为了验证 MIGA 在稀疏平面天线阵列优化中

的有效性和可靠性,本文进行了两种仿真实验。以 $PSLL$ 为优化目标,寻找最优的阵元分布,并将仿真结果与一些传统算法进行对比。实验的硬件环境为 Intel Core i7 - 7700HQ、2.8 GHz、8 GB,操作系统为 Windows 10。假定所有阵元均为理想的各向同性点源,阵元之间的间隔为 $d_x = d_y = 0.5\lambda$,并且忽略阵元之间的耦合作用。

3.1 全局旁瓣抑制

文献 [12] 采用迭代快速傅里叶变换算法 (iterative fast Fourier transform algorithm, IFFT) 对阵列规模为 20×10 、稀疏率为 46% 的矩形平面阵列进行稀疏。以抑制三维方向图在整个平面的 $PSLL$ 为优化目标,分别对阵元对称分布和非对称分布的矩形阵列进行优化,所得 $PSLL$ 分别为 -18.68 dB 和 -19.69 dB。

为了方便进行对比,本文采用 MIGA 对相同的阵列模型进行了优化,优化算法的适应度函数设置为 F_1 ,种群大小 N_p 设置为 60,采样点数为 256,迭代次数为 300。当阵元对称分布时,优化得到 $PSLL = -20.50$ dB;当阵元非对称分布时,得到的 $PSLL = -21.08$ dB。表 1 中记录了 IFFT^[12]、改进差分进化算法^[13] (improve differential evolution, IDE) 和 MIGA 优化得到的全局 $PSLL$, E, H 面的 3 dB 带宽 $\Theta_{3dB E}$ 和 $\Theta_{3dB H}$,方向性系数 D 等参数。从表 1 中可以看出, MIGA 优化得到的方向图在 E 面的 3 dB 带宽相比 IFFT 和 IDE 算法分别减少了 0.42° 和 0.64° , H 面的 3 dB 带宽分别减少了 0.70° 和 0.03° 。 $PSLL$ 相比上述算法分别下降了 1.82 dB 和 0.62 dB,方向性系数提高了 0.60 dB 和 0.02 dB。为验证算法稳定性,一共进行了 10 次蒙特卡罗实验,其中最好的 $PSLL = -20.50$ dB,最差的 $PSLL = -19.55$ dB,方差为 0.075,仿真实验的单次运行时间为 210 s。当阵元非对称分布时,采用 MIGA 算法优化得到 $PSLL = -21.06$ dB。

表 1 IFFT、IDE 和 MIGA 仿真结果对比

Tab. 1 Comparison of simulation results of IFFT, IDE and MIGA

| 算法 | $PSLL$ /dB | $\Theta_{3dB E}/(^{\circ})$ | $\Theta_{3dB H}/(^{\circ})$ | D /dB |
|----------------------|------------|-----------------------------|-----------------------------|---------|
| IFFT ^[12] | -18.68 | 6.72 | 12.78 | 24.74 |
| IDE ^[13] | -19.88 | 6.94 | 12.11 | 25.32 |
| MIGA | -20.50 | 6.30 | 12.08 | 25.34 |

阵元对称分布优化结果对应的阵元位置如图 2 所示,阵元非对称分布优化结果对应的阵元

位置如图 3 所示,图中实心圆代表被“保留”下来的阵元位置,空心圆代表开路阵元位置。图 3 显得无序,随机性大,这是由非对称阵列的优化空间陡增导致的。图 4(a) 和图 5(a) 分别绘制了阵元对称分布和非对称分布所对应的三维方向图,从中可以看出方向图的旁瓣整体都非常低,证实了算法抑制旁瓣的有效性。为了进一步观察方向图的旁瓣电平值,图 4(b) 和图 5(b) 绘制了空间方向图在 $u=0, v=0$ 两个主平面的切面方向图,同时绘制了全局最低的 $PSLL$ 所对应的切面方向图。从图中可以看出,非对称阵元分布对应的全局 $PSLL$ 更低。

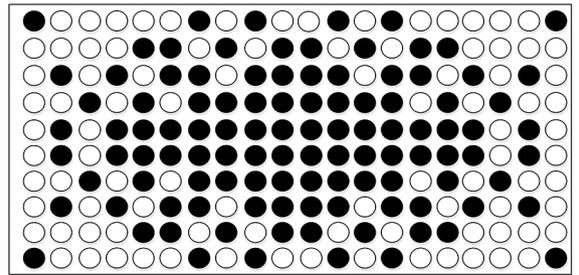


图 2 适应度函数为式(2)时阵元对称分布优化结果
Fig. 2 Optimization results of symmetrical distribution of array elements when the fitness function is Eq. 2

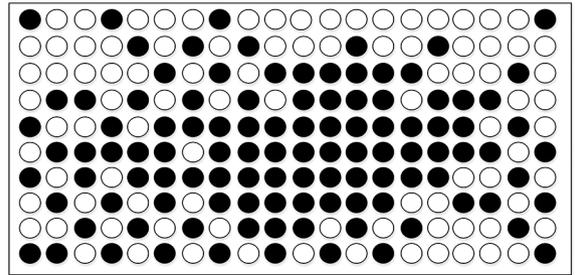
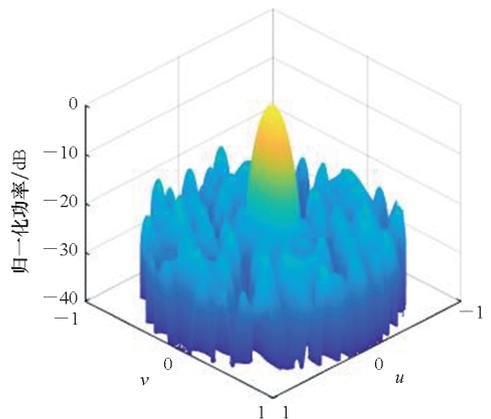


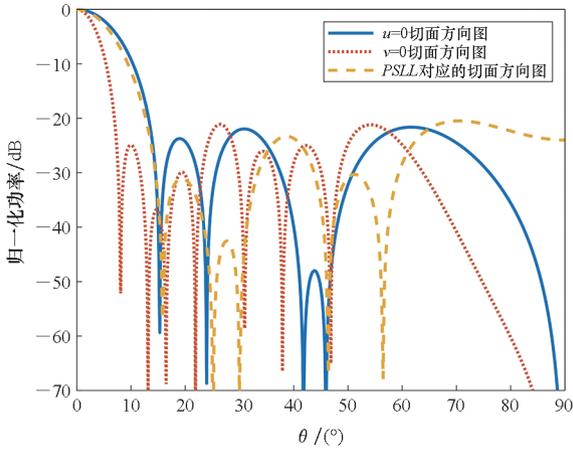
图 3 适应度函数为式(2)时阵元非对称分布优化结果

Fig. 3 Optimization results of asymmetrical distribution of array elements when the fitness function is Eq. 2



(a) 三维方向图

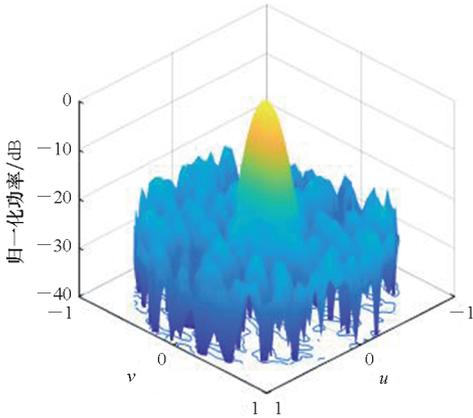
(a) Three-dimensional pattern



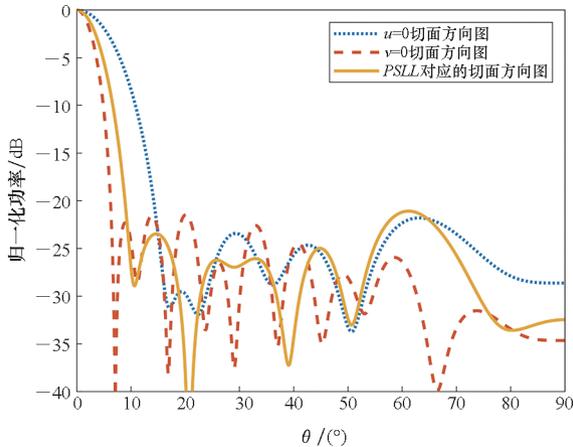
(b) 切面方向图
(b) Section pattern

图 4 适应度函数为式(2)时对称稀疏矩形阵列方向图

Fig. 4 Symmetrical sparse rectangular array pattern when the fitness function is Eq. 2



(a) 三维方向图
(a) Three-dimensional pattern



(b) 切面方向图
(b) Section pattern

图 5 适应度函数为式(2)时非对称稀疏矩形阵列方向图

Fig. 5 Asymmetrical sparse rectangular array pattern when the fitness function is Eq. 2

3.2 $\varphi = 0^\circ$ 和 $\varphi = 90^\circ$ 旁瓣抑制

文献[14]为了降低规模为 20×10 、稀疏率为 46%的矩形平面阵 $\varphi = 0^\circ$ 和 $\varphi = 90^\circ$ 两个平面的 $PSLL$ 值,采用混沌二进制粒子群优化 (chaotic binary particle swarm optimization, CBPSO)算法进行了优化,得到了 $PSLL = -52.72$ dB。

为了方便对比,本文采用 MIGA 算法对相同的阵列模型进行优化,适应度函数设置为 F_2 ,种群大小 N_p 设置为 60,采样点数为 256,迭代次数为 300。

最优的峰值旁瓣电平和为 -56.97 dB (当 $\varphi = 0^\circ$ 时, $SLL = -26.06$ dB;当 $\varphi = 90^\circ$ 时, $SLL = -30.91$ dB),最差的结果为 -52.22 dB,方差为 2.965 8。其对应的最好的阵元分布如图 6 所示。综合文献[14]和文献[15]的优化结果,从图 6 中可以看出,靠近阵列中心的阵元分布集中有利于降低 $\varphi = 0^\circ$ 和 $\varphi = 90^\circ$ 两个平面的旁瓣电平值。

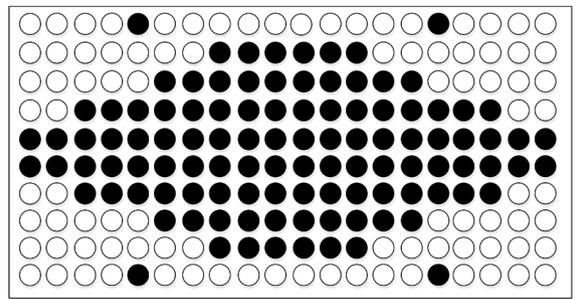


图 6 适应度函数为式(3)时阵元对称分布优化结果
Fig. 6 Optimization results of symmetrical distribution of array elements when the fitness function is Eq. 3

表 2 中记录了采用 CBPSO^[14]、二进制差分进化 (binary differential evolution, BDE) 算法^[15]和 MIGA 的优化结果,实验结果表明 MIGA 算法对旁瓣电平的抑制作用明显。在相同稀疏率的情况下, MIGA 算法优化得到的 $PSLL$ 相比 BDE 和 CBPSO 分别降低了 5.79 dB 和 4.25 dB。仿真实验结果证实了算法对旁瓣电平抑制的有效性。

表 2 CBPSO、BDE 和 MIGA 仿真结果对比
Tab. 2 Comparison of simulation results of CBPSO, BDE and MIGA

| 优化算法 | $SLL(\varphi = 0^\circ)$ / dB | $SLL(\varphi = 90^\circ)$ / dB | $PSLL$ / dB |
|-----------------------|----------------------------------|-----------------------------------|----------------|
| CBPSO ^[14] | -26.39 | -26.33 | -52.72 |
| BDE ^[15] | -26.09 | -25.09 | -51.18 |
| MIGA | -26.06 | -30.91 | -56.97 |

图 7 绘制了三维方向图,图 8 绘制了方向图在 $\varphi = 0^\circ$ 和 $\varphi = 90^\circ$ 两个平面的切面方向图,图 9 绘制了 10 次蒙特卡罗实验中最好的、最差的和平均的适应度迭代曲线,所需时间为 1 678.08 s。由图 9 可以看出,平均适应度值在 140 次时基本

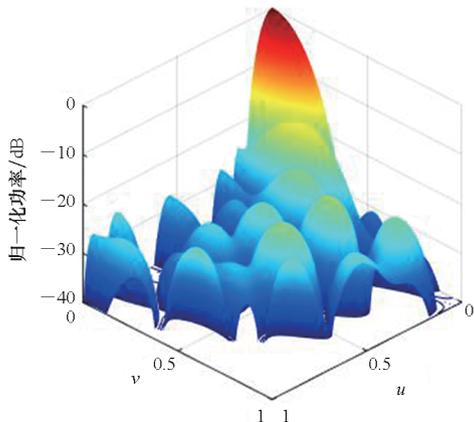


图 7 适应度函数为式(3)时三维方向图
Fig. 7 Three-dimensional pattern when the fitness function is Eq. 3

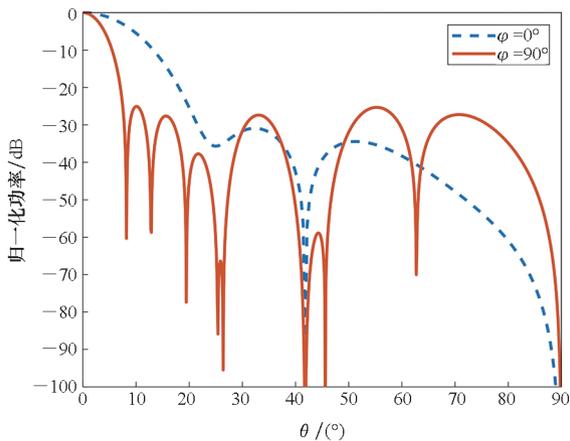


图 8 适应度函数为式(3)时切面方向图

Fig. 8 Section pattern when the fitness function is Eq. 3

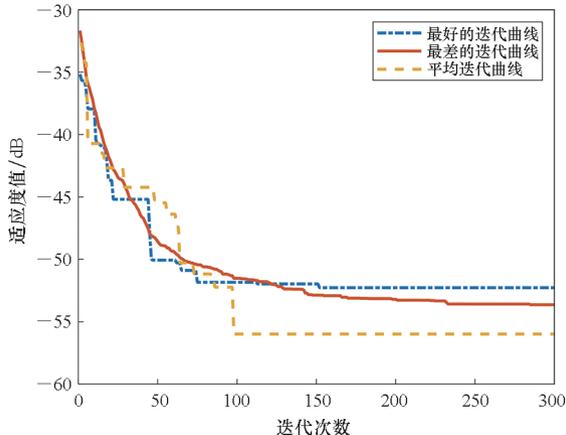


图 9 适应度函数为式(3)时适应度迭代曲线
Fig. 9 Fitness iteration curve when the fitness function is Eq. 3

收敛,可以得到平均计算量(计算方向图的次数)约为 $140 \times 60 = 8\ 400$ 。

4 结论

针对固定稀疏率、固定孔径的稀疏矩形平面阵列优化时容易陷入局部最优和优化困难的问题,本文提出了一种改进整型遗传算法。该算法在整型遗传算法的基础上,引入了等间隔采样、多点变异和优良基因重组策略,用于增强算法跳出局部最优的能力,加快算法的收敛速度。仿真结果表明,与其他算法相比,本文所提算法在固定稀疏率、固定孔径的条件下有效降低了稀疏矩形平面阵列的旁瓣电平值。本文所提出的算法对于稀疏直线阵以及各种边界条件的稀疏平面阵都具有良好的适用性。

参考文献 (References)

- [1] HAUPT R L. Antenna arrays: a computational approach [M]. New Jersey: Wiley-IEEE Press, 2010.
- [2] 陈志坤, 杜康, 彭冬亮, 等. 基于混合三角变异差分进化算法的平面稀疏阵列约束优化 [J]. 电子与信息学报, 2020, 42(4): 895 - 901.
CHEN Z K, DU K, PENG D L, et al. Planar sparse array constraint optimization based on hybrid trigonometric mutation differential evolution algorithm [J]. Journal of Electronics & Information Technology, 2020, 42(4): 895 - 901. (in Chinese)
- [3] HAUPT R L. Thinned arrays using genetic algorithms [J]. IEEE Transactions on Antennas and Propagation, 1994, 42(7): 993 - 999.
- [4] YADAV K, RAJAK A K, SINGH H. Array failure correction with placement of wide nulls in the radiation pattern of a linear array antenna using iterative fast Fourier transform [C]// Proceedings of IEEE International Conference on Computational Intelligence & Communication Technology, 2015: 471 - 474.
- [5] 王新宽, 王桂宝, 黄朝军, 等. 基于多目标粒子群优化的同心圆环阵列方向图综合算法 [J]. 重庆邮电大学学报 (自然科学版), 2022, 34(5): 877 - 885.
WANG X K, WANG G B, HUANG C J, et al. Approach for pattern synthesis of concentric ring antenna arrays based on multiple objective particle swarm optimization [J]. Journal of Chongqing University of Posts and Telecommunications (Natural Science Edition), 2022, 34(5): 877 - 885. (in Chinese)
- [6] 徐娟, 张月, 赵建平. 酉矩阵束方法的阵列稀疏优化研究 [J]. 聊城大学学报 (自然科学版), 2020, 33(6): 47 - 54.
XU J, ZHANG Y, ZHAO J P. Sparse array optimization based on unitary matrix beam method [J]. Journal of Liaocheng University (Natural Science Edition), 2020, 33(6): 47 - 54. (in Chinese)
- [7] OLIVERI G, CARLIN M, MASSA A. Complex-weight sparse linear array synthesis by Bayesian compressive sampling [J]. IEEE Transactions on Antennas and Propagation, 2012,

- 60(5): 2309 – 2326.
- [8] 曹孟华, 李龙, 谢红卫. 改进遗传算法在传声器阵列优化中的应用[J]. 国防科技大学学报, 2019, 41(6): 126 – 134.
CAO M H, LI L, XIE H W. Application of improved genetic algorithm in microphone array optimization [J]. Journal of National University of Defense Technology, 2019, 41(6): 126 – 134. (in Chinese)
- [9] 丛雯珊, 余岚, 杜鹏飞, 等. 基于粒子群算法的宽带真延时平面阵列方向图综合[J]. 国防科技大学学报, 2020, 42(5): 31 – 36.
CONG W S, YU L, DU P F, et al. Planar array pattern synthesis of wideband real time delay based on particle swarm optimization algorithm [J]. Journal of National University of Defense Technology, 2020, 42(5): 31 – 36. (in Chinese)
- [10] 梁爽, 孙庚, 刘衍珩. 改进布谷鸟算法用于阵列天线方向图优化[J]. 西安电子科技大学学报, 2019, 46(1): 174 – 180.
LIANG S, SUN G, LIU Y H. Improved cuckoo search algorithm for optimizing the beam patterns of linear antenna arrays [J]. Journal of Xidian University, 2019, 46(1): 174 – 180. (in Chinese)
- [11] JIANG Y, ZHANG S, GUO Q, et al. Synthesis of uniformly excited concentric ring arrays using the improved integer GA [J]. IEEE Antennas and Wireless Propagation Letters, 2016, 15: 1124 – 1127.
- [12] 曾伟一, 梁颖, 黄伟. 基于迭代FFT算法的平面稀疏阵列优化方法[J]. 电讯技术, 2011, 51(11): 99 – 102.
ZENG W Y, LIANG Y, HUANG W. An optimum method for thinned planar array based on iterative FFT algorithm [J]. Telecommunication Engineering, 2011, 51(11): 99 – 102. (in Chinese)
- [13] 刘恒, 赵宏伟, 李维梅, 等. 平面稀疏阵列天线的约束优化设计[J]. 电讯技术, 2016, 56(2): 166 – 170.
LIU H, ZHAO H W, LI W M, et al. Constraint optimization of planar thinned array antenna [J]. Telecommunication Engineering, 2016, 56(2): 166 – 170. (in Chinese)
- [14] WANG W B, FENG Q Y, LIU D. Synthesis of thinned linear and planar antenna arrays using binary PSO algorithm [J]. Progress in Electromagnetics Research, 2012, 127: 371 – 387.
- [15] ZHANG L, JIAO Y C, WENG Z B, et al. Design of planar thinned arrays using a Boolean differential evolution algorithm [J]. IET Microwaves, Antennas & Propagation, 2010, 4(12): 2172.