

基于改进风驱动算法的非等间距直线阵综合

任作琳 田雨波 孙菲艳

(江苏科技大学电子信息学院 镇江 212003)

摘要 针对风驱动优化(WDO)算法在解决非等间距直线阵方向图综合问题时收敛精度不高和局部寻优能力不足等缺陷,提出一种小波变异风驱动优化(WDOWM)算法,其中的小波变异算子采用随机化思想丰富了种群多样性。应用该算法综合不同数目阵元到非等间距直线阵方向图实例中,采用二阶多因素多水平的均匀设计方法确定算法参数组合。仿真结果表明,在要求低旁瓣电平和给定方向零陷的情况下,该算法的收敛精度和收敛速度均优于基本风驱动优化算法;与采用粒子群(PSO)算法优化此问题的已有文献相比,所提算法综合的效果更佳。仿真结果说明了所提算法性能良好,适用于天线阵综合问题。

关键词 风驱动优化算法,小波变异,均匀设计,天线阵综合

中图分类号 TP301.6 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.7.049

Unequally Spaced Linear Array Synthesis Using Modified Wind Driven Optimization Algorithm

REN Zuo-lin TIAN Yu-bo SUN Fei-yan

(School of Electronics and Information, Jiangsu University of Science and Technology, Zhenjiang 212003, China)

Abstract Because of some shortcomings of traditional wind driven optimization (WDO) algorithm for the synthesis of unequally spaced linear antenna arrays, such as the bad convergence accuracy and bad local optimal searching capability, the WDO with wavelet mutation (WDOWM) algorithm was proposed. The modified WDO algorithm with a wavelet mutation operator was used to adopt randomization to rich population diversity. Using the modified algorithm to deal with the synthesis problems of multi-elements unequally spaced linear antenna arrays, second-order multi-factor and multi-level uniform design method was used to determine the algorithm parameter combinations. The simulation results show its convergence accuracy and speed are superior to the traditional WDO algorithm in the pattern synthesis of array antennas with low side-lobe level suppression and null control in specified directions. In addition, the performances of the proposed algorithm are superior to the particle swarm optimization (PSO) algorithm used in the cited references. These results suggest that the WDOWM algorithm has good performance, and it is suitable for the antenna synthesis problems.

Keywords Wind driven optimization algorithm, Wavelet mutation, Uniform design, Antenna array synthesis

基本风驱动优化(Wind Driven Optimization, WDO)算法是美国学者 Bayraktar 等人于 2010 年提出的一种基于群体的全局优化算法^[1]。该算法以大气物理学中空气粒子受力运动模型为基础,简化模拟空气粒子在大气中的受力运动情况,结合牛顿第二定律及理想气体状态方程,推导出空气质点在每一次迭代中的速度和位置更新方程。WDO 算法可以解决多维和多模态问题,也可以解决连续和离散优化问题,被广泛应用于电磁优化^[2-4]、图像处理^[5]、云资源调度分配^[6]等多个领域。与其他传统群智能优化算法一样,基本 WDO 算法在解决部分问题时也存在全局“探索”能力与局部“开发”能力之间的矛盾。为了增强基本 WDO 算法局部开发能力,提高算法寻优效率,提出一种带有小波变异策略的 WDO 算法:小波变异风驱动优化(Wind Driven Optimization with Wavelet Mutation, WDOWM)算法。该算法中小波变异算子的引用使

原本丧失活性的群体加入了新鲜血液,很好地避免了基本 WDO 算法在搜索后期寻优能力不足的缺陷。在智能优化算法应用领域中,不同参数的选取对算法性能影响重大,由于缺乏理论指导,算法参数一般采用试探法来确定,使得工作量增大,同时又不能保证算法性能最优。均匀设计是一种可以用最少的信息获取空间最多信息的方法,其设计的科学性和合理性已在遗传算法^[7]、蚁群算法^[8]等算法参数选取中得以验证。本文应用二阶五因素十二水平的均匀设计法确定算法参数组合,避免盲目尝试,从而提高算法的科学性。

阵列天线广泛应用于雷达、无线通信和电子对抗等领域。近年来,非等间距天线阵(也叫非周期天线阵)吸引了许多学者的注意。相对于等间距天线阵,非等间距天线阵具有如下优点:1)不受栅瓣影响;2)在天线孔径不变的情况下可使用更少的阵元;3)当阵元激励均匀等幅时,能获得更低的旁瓣电

到稿日期:2015-04-24 返修日期:2015-07-21 本文受国家自然科学基金项目(61401179)资助。

任作琳(1991-),女,硕士生,主要研究方向为智能优化算法及其应用,E-mail:lebaishi_2014@163.com;田雨波(1971-),男,博士,教授,主要研究方向为智能优化算法在电子学和电磁学中的应用;孙菲艳(1991-),女,硕士生,主要研究方向为智能信息处理与系统。

平。由于上述优点,非等间距线阵设计一直是天线阵列综合领域的研究热点。非等间距线阵综合问题是一个复杂的非线性多维优化问题,不少学者通过群智能优化算法如改进差分进化算法^[9]、改进实数遗传算法^[10]、布谷鸟搜索算法^[11]、粒子群算法^[12]等研究了该问题。本文首先应用基本 WDO 算法实现该问题,得到了优于文献^[12]的结果。为了加快基本 WDO 算法在非线性多维优化问题中的收敛速度并提高其寻优精度,提出了一种 WDOWNM 算法,同时应用该算法实现文献^[12]中的实例。仿真结果表明,改进算法的收敛精度更高,收敛速度更快,更具有优越性。

1 风驱动优化(WDO)算法

WDO 算法是对大气运动的简单模拟,其将空气微团视为单位质量、单位体积的空气粒子来研究,算法模型中并没有完全复制空气粒子在大气运动中所受的力,只是考虑了 4 个主要力的影响:摩擦力、重力、气压梯度力和科里奥利力。将这 4 个力代入牛顿第二定律,并结合理想气体状态方程便能得出 WDO 算法速度更新方程^[2]。算法基本原理可以描述如下。

一个由 m 个空气粒子组成的群体在 D 维搜索空间中以一定速度运动搜索,速度矩阵可以表示为 $V=[V_1^k, \dots, V_m^k, \dots, V_m^k]^T$,其中 $1 \leq i \leq m, k$ 为迭代次数。由于每个空气粒子都在 D 维空间内进行搜索,将第 i 个粒子的每一维度都赋予一个速度,那么第 i 个空气粒子在第 k 次迭代中的速度表示为 $V_i^k=(v_{i1}^k, v_{i2}^k, \dots, v_{id}^k, \dots, v_{iD}^k)$,其中 $1 \leq d \leq D$ 。不难看出,对于 D 维问题,空气粒子速度可以被定义为一个 $m \times D$ 矩阵,即 $V=$

$$\begin{bmatrix} v_{11}^k & \dots & v_{1D}^k \\ \vdots & \ddots & \vdots \\ v_{m1}^k & \dots & v_{mD}^k \end{bmatrix}, \text{其中 } 1 \leq k \leq T, T \text{ 为最大迭代次数。在空气}$$

粒子运动搜索过程中,其速度的不断变化导致对应位置不断移动,最终探索出的最优位置即为最优解。那么,第 i 个空气粒子在第 k 次迭代中的位置矢量可以表示为 $X_i^k=(x_{i1}^k, x_{i2}^k, x_{i3}^k, \dots, x_{id}^k, \dots, x_{iD}^k)$,则种群位置矩阵为 $X=$

$$\begin{bmatrix} x_{11}^k & \dots & x_{1D}^k \\ \vdots & \ddots & \vdots \\ x_{m1}^k & \dots & x_{mD}^k \end{bmatrix}, \text{其中 } 1 \leq i \leq m, 1 \leq d \leq D, 1 \leq k \leq T. \text{与其}$$

他群智能优化算法相似,WDO 算法也需要根据问题的目标函数构造压力值函数(适值函数),根据压力值(适应值)的优劣不断更新种群信息。

空气粒子速度和位置根据如下方程进行变化:

$$v_{id}^{k+1}=(1-\alpha)v_{id}^k - gx_{id}^k + [RT] \frac{1}{j} - 1 | (x_{id}^{k_{best}} - x_{id}^k) | + \left(\frac{c v_{i0}^{k_{therdim}}}{j} \right) \quad (1)$$

$$x_{id}^{k+1}=x_{id}^k + (v_{id}^{k+1} \Delta t) \quad (2)$$

式(1)中, α 为摩擦系数, g 为重力加速度常数, RT 为气压梯度力影响系数, c 为科里奥利力影响系数,4个常数都是由相对应的力简化得来,均为正常数。 j 为压力值的一个升序排列, $j=1$ 表示压力值最小,可以认为是空气粒子迄今为止找到的最优解(个体最优解)。 $x_{id}^{k_{best}}$ 为全局最优解(全局最优空气粒子位置,为 D 维行向量), $v_{i0}^{k_{therdim}}$ 为第 i 个粒子在第 k 次迭代中除 d 维以外的其他任意一维度的速度,此速度由

科里奥利力引入。该速度的加入增强了各个个体之间的社会合作,使得空气粒子在搜索过程中结合同伴的经验进行适当调整,不仅增强了空气粒子之间的相互合作和信息共享,同时也避免了算法陷入局部最优,使算法鲁棒性增强。为了简化方程,一般令式(2)中 $\Delta t=1$ 。

2 小波变异风驱动优化(WDOWNM)算法

2.1 WDO 算法的改进

基本 WDO 算法中空气粒子速度的更新方程采用式(1),式中等号右边共分为 4 个部分:1)摩擦力对空气粒子的影响,是空气粒子对先前速度的“继承”,属于认知部分;2)重力影响部分,重力项的存在可以有效防止空气粒子在搜索过程中困于边界,这增加了算法整体寻优能力;3)为气压梯度力影响部分,空气粒子追随自身经过的最好位置,它是自我认知学习部分,这部分关系到算法局部搜索精度;4)科里奥利力影响部分,其他维度速度的加入促使各个空气粒子信息互享,增强了算法探索新区域的能力。4个部分相互合作、相互制约,使得空气粒子向最为理想状态发展,然而当某些空气粒子的第一部分和第三部分占优势,算法就极易出现收敛停滞现象,导致种群逐渐丧失多样性,最终收敛到局部最优值。

为了克服这种现象,在基本 WDO 算法中引入小波变异算子,采用随机化思想引入精英个体丰富了种群多样性,减小了算法陷入局部极值的概率。根据文献^[13],本文选取 Morlet 小波函数进行变异,算法实现过程中,采用变异概率 $pm \in [0,1]$ 的方式决定是否进行变异操作。空气粒子位置更新过后,系统产生一个随机数 $rand \in [0,1]$,若 $rand < pm$,则认定该空气粒子需要进行小波变异, pm 取值的大小根据粒子的维数决定。一般来说,当粒子的维数很低时(粒子的维数小于 5), pm 概率可以设为 0.5~0.8;当粒子的维数为 5~10 时, pm 概率可以设为 0.3~0.4;当粒子的维数为 11~100 时, pm 概率可以设为 0.1~0.2;当粒子的维数为 101~1000 时, pm 概率可以设为 0.05~0.1,当粒子维数很大(粒子维数大于 1000)时, pm 概率小于 0.05。原则上,一般当粒子维数越大时, pm 的值越小。这是因为若粒子维数很大,而变异概率值也很大,那么将对更多的粒子维数进行变异操作就会增加搜索时间。更重要的是,算法在每次迭代时,变异操作会破坏掉当前粒子所包含的有用信息,从而使所有的粒子元素都重新随机分配。

小波变异公式如下:

$$(x_{id}^k)' = \begin{cases} x_{id}^k + \sigma \times (x_{id}^{k_{max,i}} - x_{id}^k), & \sigma > 0 \\ x_{id}^k + \sigma \times (x_{id}^k - x_{id}^{k_{min,i}}), & \sigma \leq 0 \end{cases} \quad (3)$$

其中, $(x_{id}^k)'$ 为进行小波变异后空气粒子的位置, $x_{id}^{k_{min,i}}$ 和 $x_{id}^{k_{max,i}}$ 分别表示该空气粒子此次迭代搜索的位置的下界和上界,即 $x_{id}^k \in [x_{id}^{k_{min,i}}, x_{id}^{k_{max,i}}]$ 。 σ 为 Morlet 小波函数值,其计算公式如下:

$$\sigma = \frac{1}{\sqrt{a}} e^{-\frac{\varphi^2}{2}} \cos\left(5\left(\frac{\varphi}{a}\right)\right) \quad (4)$$

由文献^[13]可知, $a=1$ 时 Morlet 小波函数的 99% 的能量都包含在区间 $[-2.5, 2.5]$ 中,所以式(4)中 φ 取值范围区间为 $[-2.5a, 2.5a]$ 中的伪随机数。 a 为尺度参数,其计算公式如下:

$$a = e^{-\ln(g) \times (1 - \frac{1}{j})^{\epsilon_{gen}} + \ln(g)} \quad (5)$$

其中, ξ_{sm} 是单调递增函数的形状参数, g 为 a 的上限值, t 为当前迭代次数, T 为最大迭代次数。通过调节小波函数的尺度参数 a 来调整小波函数 σ 的幅度值。随着迭代次数的增加, 通过减少小波函数的幅度值来限制算法的搜索空间, 从而使算法的能力得到更好的调整。由式(5)不难看出, 当 t/T 和形状参数 ξ_{sm} 相等时, g 越大, 尺度参数 a 的值越大, 变异幅度 σ 越小; 当 t/T 和 g 相等时, 形状参数 ξ_{sm} 越大, 尺度参数 a 的值越大, 变异幅度 σ 下降的速率就越快。根据实验结果, 建议参数 g 为 1000 或 100, 通过调整参数 ξ_{sm} 来满足不同实际应用问题的需要。由式(3)不难发现: 如果 $\sigma(\sigma > 0)$ 越接近 1, 则空气质点变异后的值越接近 $x_{\max i}$; 反之, $\sigma(\sigma < 0)$ 越接近 -1, 空气质点变异后的值越接近 $x_{\min i}$ 。变异操作引入精英个体维持种群多样性, 有助于扩大算法搜索区域。

2.2 改进算法流程

WDOWM 算法在基本 WDO 算法位置更新后选择性地执行小波变异操作, 将进行过变异的空气粒子回归到种群中去, 继续进行迭代更新。改进算法具体流程如图 1 所示。

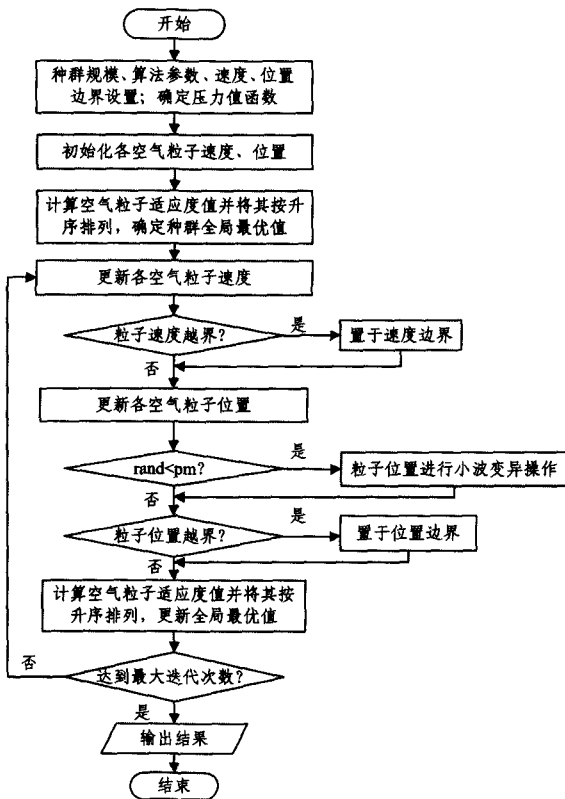


图 1 小波变异风驱动优化算法流程

2.3 改进算法性能分析

为了验证 WDOWM 算法的有效性, 选用两个典型的不同特征的测试函数对其进行测试。

① Schwefel 函数

$$f_{Schwefel}(x) = \sum_{i=1}^N |x_i| + \prod_{i=1}^N |x_i| \quad (6)$$

Schwefel 函数搜索范围为 $x_i \in [-10, 10]$, 该函数为单峰函数, 可以用来检测算法的执行性能。在自变量取值 $(0, 0, \dots, 0)$ 处取得极小值 0。

② Ackley 函数

$$f_{Ackley}(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2}\right) - \exp\left(\frac{1}{N} \sum_{i=1}^N \cos(2\pi x_i)\right) + 20 + e \quad (7)$$

Ackley 函数的搜索范围为 $x_i \in [-32, 32]$, 是经典多峰函数, 通过增加余弦波来调整指数函数存在很多局部最优值, 用来测试算法跳出局部最优值的能力; 在自变量取值 $(0, 0, \dots, 0)$ 处取得极小值 0。

用上述两个测试函数分别对基本 WDO 算法、WDOWM 算法和标准粒子群优化 (PSO) 算法进行性能优化测试。具体参数设置如下: 种群规模 $m=20$, 种群维度 $D=15$, 最大迭代次数 $T=200$, 算法参数 $RT=4.09$, $g=0.356$, $\alpha=0.161$, $c=0.155$, 小波变异概率 $pm=0.2$, 小波变异算子参数 $\xi_{sm}=0.5$, $g_{sm}=1000$, 最大速度 $v_{\max}=0.172$, 标准 PSO 算法中学习因子 $c_1=c_2=2.05$, 惯性因子 $\omega=1$ 。每次优化实验中, 3 种算法的初始种群、初始速度、初始位置、初始全局最优值均相同, 算法按照 Monte Carlo 规则独立运行 50 次。图 2 和图 3 为 50 次优化的平均最优适应值绘制成的迭代曲线, 用该曲线来分析各算法的收敛速度。表 1 和表 2 为仿真结果, 其中标准差用来评估算法稳定性, 最优适应值和平均最优适应值用来分析算法精度。

从图 2 和图 3 以及表 1 和表 2 可以得出以下结论。

(1) 算法收敛速度分析

从图 2 和图 3 清晰看出, 无论是优化单峰 Schwefel 函数还是多峰 Ackley 函数, WDOWM 算法的收敛速度都是最快的, 标准 PSO 算法收敛得最慢, 说明增加小波变异操作对 WDO 算法的收敛速度有了明显提高。

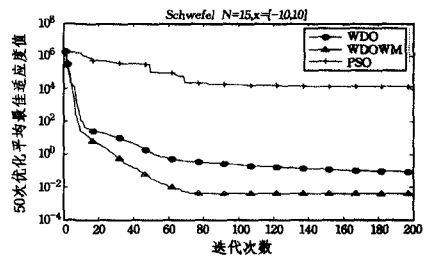


图 2 Schwefel 函数迭代曲线图

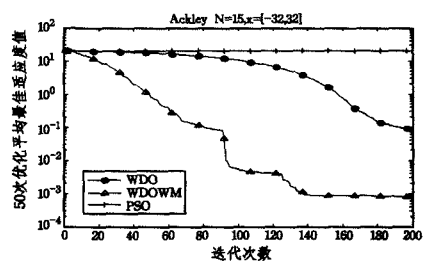


图 3 Ackley 函数迭代曲线图

(2) 算法稳定性分析

标准差反映了算法的稳定性, 标准差越小表明算法稳定性越好。由表 1 和表 2 可以看出, 优化两种函数所得标准差由小到大为: WDOWM 算法 < WDO 算法 < PSO 算法, 这说明 3 种算法中 WDOWM 算法的稳定性优于 WDO 算法, 标准 PSO 算法的稳定性最差。

表 1 Schwefel 函数仿真结果

算法	标准差	平均最优适应度值	最优适应度值
WDO	0.010436	0.087403	0.004436
WDOWM	4.76E-05	0.003893	0.000775
PSO	1826.222	12819.55	29.2392

表2 Ackley 函数仿真结果

算法	标准差	平均最优适应度值	最优适应度值
WDO	0.003176	0.087077	0.002059
WDOWM	5.63E-05	0.000784	7.99E-05
PSO	0.03175	20.28387	19.62084

(3)算法收敛精度分析

分析表1和表2中的最优适应度值,不难看出WDOWM算法的收敛精度明显高于WDO算法和PSO算法。平均最优适应度值中WDO算法和WDOWM算法均能寻到满足要求的结果,但WDO算法的搜索结果的精度没有WDOWM算法搜索结果的精度高。

综合上述分析,WDOWM算法无论是在收敛速度、收敛精度还是算法稳定性方面均优于基本WDO算法,同时也优于标准PSO算法。

3 非等间距直线阵综合

天线阵通常包括单元总数、各单元的空间位置分布、各单元的激励幅度及激励相位,这4个参数是可变的。若给定上述4个参数,根据这些参数确定阵列天线辐射特性,包括方向图、方向性系数、增益和阻抗等,则称为阵列天线的分析问题;反之,根据需求的辐射特性去确定上述4个参数的过程则称为阵列天线的综合问题。这两个问题是该领域的核心问题,也是阵列天线工程应用中必须解决的基础问题^[14]。

3.1 直线天线阵列模型

考虑阵元数为2N的非等间距直线阵列,且阵元为各向同性单元,其阵列模型如图4所示,其中W₁,W₂,...,W_{2N}为阵元,以原点为中心沿着Z轴对称分布。θ为阵轴线与入射波射线的夹角,θ∈[0,π]。由于阵元关于原点呈对称分布,因此可以对其简化研究,仅考虑第1~N个天线阵元。该阵列阵因子表达式为:

$$AF(\theta) = 2 \sum_{n=1}^N I_n \exp[j(kd_n \cos\theta + \varphi_n)] \quad (8)$$

应用欧拉定理可将式(8)改写为:

$$AF(\theta) = 2 \sum_{n=1}^N I_n \cos(kd_n \cos\theta + \varphi_n) \quad (9)$$

其中,I_n和φ_n为第n个天线阵元的激励幅度和激励相位,φ_n∈[0,2π],k=2π/λ为波数,d_n为第n个天线阵元到原点的距离。

本文研究非等间距直线阵综合问题,优化对象为各单元的空间位置,即d_n(1≤n≤N)。假设天线阵元的激励均匀等幅同相,则I_n=1且φ_n=0,那么阵因子表达式(9)可以简化为:

$$AF(\theta) = 2 \sum_{n=1}^N \cos(kd_n \cos\theta) \quad (10)$$

天线阵的方向图函数F(θ)用归一化的阵因子来表示:

$$F(\theta) = 20 \log \left| \frac{AF(\theta)}{AF_{\max}} \right| \quad (11)$$

其中,AF_{max}=max|AF(θ)|,单位为分贝(dB)。

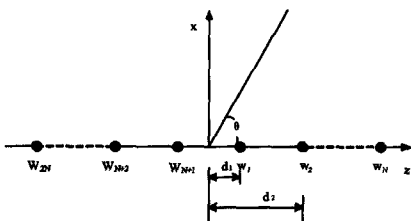


图4 非等间距直线天线阵列模型

3.2 阵列综合目标函数及算法位置边界条件的设定

基于WDOWM算法的非等间距直线阵综合的具体实施是根据不同设计要求将算法中的压力值函数设为对应的目标函数。若方向图综合中只对旁瓣电平有要求,根据文献[12]定义目标函数为:

$$fitness_1 = \sum_i \frac{1}{\Delta\theta_i} \int_{\theta_{li}}^{\theta_{ui}} |AF(\theta)|^2 d\theta \quad (12)$$

其中,[θ_{li},θ_{ui}]为旁瓣电平抑制区间,Δθ_i=θ_{ui}-θ_{li},AF(θ)为阵因子。若方向图综合中,不仅对旁瓣电平有要求,而且要求在指定方向上生成零陷,则根据文献[12],此时目标函数定义为:

$$fitness_2 = \sum_i \frac{1}{\Delta\theta_i} \int_{\theta_{li}}^{\theta_{ui}} |AF(\theta)|^2 d\theta + \sum_k |AF(\theta_k)|^2 \quad (13)$$

式(13)为在式(12)的基础上考虑零陷所得的目标函数的表达式,其中θ_k为指定零陷方向,k为零陷个数。

对于非等间距直线阵综合问题,设置算法的最大搜索速度为v_{max},则经过速度更新方程(1)更新过的速度v_{d^{k+1}}需要进行越界判断操作,即(v_{d^{k+1}})' = { v_{max}, if v_{d^{k+1}} > v_{max}; -v_{max}, if v_{d^{k+1}} < -v_{max} }, (v_{d^{k+1}})'为进行越界判断后的第k+1次迭代后速度。由于第一个阵元位置在λ/2×[0,1]的范围内,第二个阵元位置在λ/2×[1,2]的范围内,第三个阵元位置在λ/2×[2,3]的范围内(λ为波长),...,因此设置空气粒子位置搜索范围为:

$$(x_d^{k+1})' = \begin{cases} x_d^{k+1}, & \text{if } x_d^{k+1} \in [d-1, d] \\ d - rand, & \text{otherwise} \end{cases} \quad (14)$$

其中,1≤d≤D,rand为[0,1]之间产生的均匀分布随机数,(x_{d^{k+1}})'为进行越界判断后的第k+1次迭代后的位置。

3.3 基于二阶均匀设计的算法参数选取

对于不同的优化问题,算法参数选取有所不同。即使对于同一类优化问题,由于问题规模不一样,算法参数选取也不尽相同。对于WDO算法,需要确定v_{max},RT,g,α,c5个参数,它们在算法运行时不仅各尽其责,而且相互影响、相互耦合并联系紧密,与算法性能之间有着复杂的关系。由于缺乏理论指导,算法参数一般采用试探法来确定。若每个参数取5个不同水平值,要在这些取值组合中找到最好的一组,则共需要进行5⁵=3125次实验,这使得工作量增大,且又不能同时保证所选水平值范围能使算法性能发挥到最优。

本文采用均匀设计法确定WDO算法参数。均匀设计是由中国数学家王元和方开泰于1978年首次提出的,基本思想是使试验点在因子空间中具有较好的均匀分散性,使某种均匀性度量最小化。均匀设计表用代号U_n(q^s)表示,U表示均匀设计表,s表示因素个数(参数个数),q为试验水平数,n表示试验次数^[15]。所选最大速度范围v_{max}∈(0.1,0.21),各参数范围分别为:RT∈(1.6,4.35),g∈(0.1,0.65),α∈(0.05,0.27),c∈(0.05,0.38)。将每个参数范围进行11等分后,得到从小到大排列的12个参数值,采用文献[16]附录中得到的均匀设计表U₁₂(12⁵),以相应均匀设计表中的数值为序号,编制实验方案如表3所列。表3中括号前的数值为序号,括号内数值为对应的参数值。

表3 均匀设计方案(一阶)

组	v_{max}	RT	g	α	c
1	1(0.10)	3(2.1)	4(0.25)	9(0.21)	12(0.38)
2	2(0.11)	6(2.85)	8(0.45)	5(0.13)	11(0.35)
3	3(0.12)	9(3.6)	12(0.65)	1(0.05)	10(0.32)
4	4(0.13)	12(4.35)	3(0.2)	10(0.23)	9(0.29)
5	5(0.14)	2(1.85)	7(0.4)	6(0.15)	8(0.26)
6	6(0.15)	5(2.6)	11(0.6)	2(0.07)	7(0.23)
7	7(0.16)	8(3.35)	2(0.15)	11(0.25)	6(0.20)
8	8(0.17)	11(4.1)	6(0.35)	7(0.17)	5(0.17)
9	9(0.18)	1(1.6)	10(0.55)	3(0.09)	4(0.14)
10	10(0.19)	4(2.35)	1(0.1)	12(0.27)	3(0.11)
11	11(0.20)	7(3.1)	5(0.3)	8(0.19)	2(0.08)
12	12(0.21)	10(3.85)	9(0.5)	4(0.11)	1(0.05)

考虑经典测试函数集中 $f_{Rastrigrin}(x) = \sum_{i=1}^N [x_i^2 - 10\cos(2\pi x_i) + 10]$ 函数与天线阵综合问题目标函数(12)相似,采用该函数对这12组参数组合进行测试。均匀设计实验中WDO算法使用20个空气粒子,迭代次数为200,维度为15,且 $x_i \in [-5.12, 5.12]$,每次迭代对初始种群、速度、位置均相同,实验独立运行20次。所得迭代曲线如图5所示。由图5不难看出第八组(见表3粗体)参数取值使得算法寻优精度达到最高,这表明试验点中最优参数组合应该在此组合附近,故在该组附近再进行二阶均匀实验。5个参数范围缩小: $v_{max} \in (0.161, 0.172)$, $RT \in (4.00, 4.11)$, $g \in (0.348, 0.359)$, $\alpha \in (0.155, 0.177)$, $c \in (0.155, 0.188)$,继续进行均匀设计实验。通过仿真结果分析,最大速度取值为 $v_{max} = 0.172$,算法中 $RT=4.09$, $g=0.356$, $\alpha=0.161$, $c=0.155$ 时性能最优。

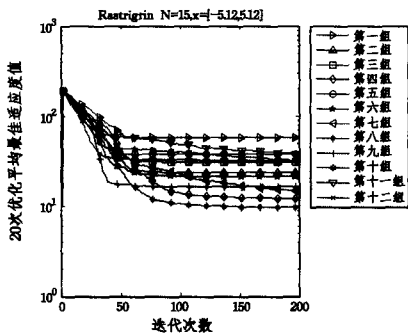


图5 仿真结果

4 实例仿真与结果分析

4.1 仿真实例

将WDOWM算法和WDO算法应用到文献[12]中的非等间距直线阵位置综合实例中以验证算法的可用性;同时进行仿真实例验证,将仿真结果与文献中PSO算法结果进行对比,说明了基于WDOWM算法的非等间距直线阵综合方法的有效性。两个算法中空气粒子种群个数 $m=40$,由于线阵模型中阵元对称分布,算法维度 $D=N$,算法参数采用均匀设计结果,即 $RT=4.09$, $g=0.356$, $\alpha=0.161$, $c=0.155$,小波变异概率 $pm=0.2$,空气粒子最大速度 $v_{max}=0.172$ 。空气粒子位置搜索范围如式(14)所示。在仿真实例中,WDO算法和WDOWM算法各自独立实验50次,每次实验的初始种群不同,实验结果为50次实验中的最优结果。

实例一 对一个 $2N=10$ 阵元直线阵进行位置综合,阵元各向同性。设计指标要求主瓣对准 90° 方向,旁瓣电平抑制区间为 $[0^\circ, 76^\circ]$ 和 $[104^\circ, 180^\circ]$,最大旁瓣电平(SLL)在

-15dB 以下。目标函数选择式(12),最大迭代次数 $T=200$,维度 $D=5$,小波变异算子参数 $\xi_{um}=0.2$, $g_{um}=1000$ 。算法得到的阵元位置分布如表4所列,归一化波束方向图如图6所示,WDO算法、WDOWM算法的适应函数收敛曲线如图7所示,为使读者更好理解各阵元位置的搜索范围关系,WDOWM算法中种群第一个空气粒子的位置矢量变化如图8所示。

表4 10阵元位置分布表

阵元序号	PSO($\ast \lambda/2$)	WDO($\ast \lambda/2$)	WDOWM($\ast \lambda/2$)
1,6	± 0.503	± 0.480	± 0.482
2,7	± 1.11	± 1.079	± 1.100
3,8	± 2.13	± 2.052	± 2.051
4,9	± 3.00	± 3.012	± 3.000
5,10	± 4.22	± 4.240	± 4.268
SLL(dB)	-17.4173	-19.1133	-19.2322

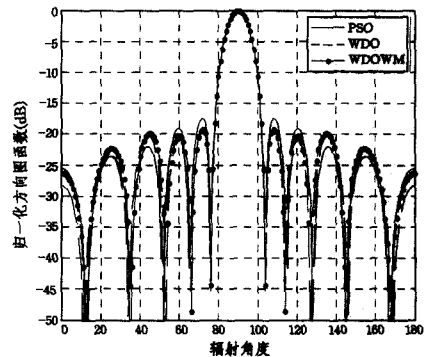


图6 10阵元线阵归一化波束方向图

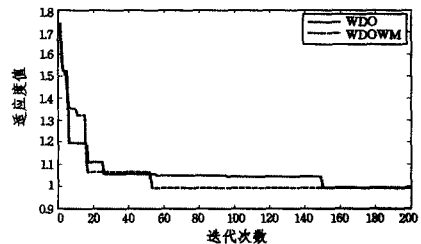


图7 WDO、WDOWM算法综合10阵元阵列适应函数收敛曲线

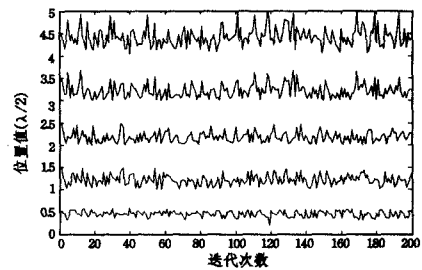


图8 WDOWM算法中第一个空气粒子位置矢量变化图

实例二 对一个 $2N=28$ 阵元直线阵进行位置综合,阵元各向同性。设计指标要求主瓣对准 90° 方向,旁瓣电平抑制区间为 $[0^\circ, 86^\circ]$ 和 $[94^\circ, 180^\circ]$,最大旁瓣电平(SLL)在 -10dB 以下,在 $55^\circ, 57.5^\circ, 60^\circ, 120^\circ, 122.5^\circ, 125^\circ$ 产生低于 -50dB 零陷。目标函数选择式(13),最大迭代次数 $T=2000$,维度 $D=14$,小波变异算子参数 $\xi_{um}=1.5$, $g_{um}=100$ 。算法得到的阵元位置分布如表5所列,图9为辐射角 $\theta \in [90^\circ, 180^\circ]$ 范围内的归一化波束方向图,WDO算法、WDOWM算法适应函数收敛曲线如图10所示。表6列出了WDO算法和WDOWM算

法在两个实例中的耗时情况。

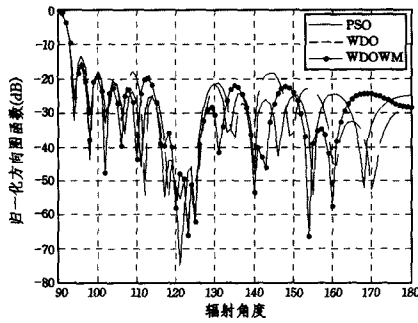


图9 28阵元线阵归一化波束方向图

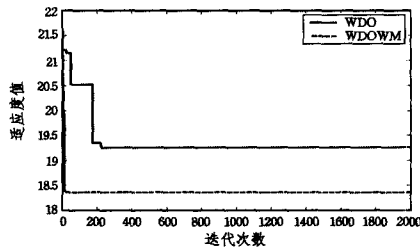


图10 WDO、WDOWM算法综合28阵元阵列适应函数收敛曲线

表5 28阵元位置分布表

阵元序号	PSO(*λ/2)	WDO(*λ/2)	WDOWM(*λ/2)
1,15	±0.56	±0.381	±0.454
2,16	±1.56	±1.457	±1.459
3,17	±2.44	±2.001	±2.358
4,18	±3.50	±3.308	±3.038
5,19	±4.54	±4.201	±4.134
6,20	±5.48	±5.263	±5.159
7,21	±6.53	±6.088	±6.237
8,22	±7.54	±7.076	±7.245
9,23	±8.39	±8.019	±8.155
10,24	±9.50	±9.242	±9.139
11,25	±10.63	±10.664	±10.540
12,26	±11.42	±11.291	±11.688
13,27	±12.32	±12.316	±12.623
14,28	±13.68	±13.511	±13.981
SLL(dB)	-13.2671	-15.0769	-15.9968
55°零陷(dB)	-61.4622	-50.9922	-62.0002
57.5°零陷(dB)	-51.6551	-53.0154	-55.8512
60°零陷(dB)	-52.7399	-51.4069	-57.8162

表6 算法运行耗时

算法	综合10阵元阵列时间(s)	综合28阵元阵列时间(s)
WDO	2.68	34.61
WDOWM	2.72	34.06

4.2 结果分析

由图6、图9归一化波束方向图可以看出,3种算法的综合结果均能满足设计要求,且主瓣对准方向均为90°。由图6清晰可见,WDOWM算法的最大旁瓣电平是3种优化结果中最低的,图9中3种算法均能在120°、122.5°、125°处产生低于-50dB零陷。由于对称性,该线阵在55°、57.5°、60°方向也可生成低于-50dB零陷。图6和图9的结论说明WDO算法和WDOWM算法可以解决如阵列天线综合问题这一类的复杂非线性多维优化问题。

由图7和图10可以看出,WDOWM算法的收敛精度均高于WDO算法,图10中WDOWM算法在不到200次迭代时就可以找到最优值,而WDO算法在搜索后期均出现不同程度的搜索停滞现象,这表明WDOWM算法寻优效率高,可

以有效跳出局部最优解,探索更优的解空间。

WDO算法加入小波变异算子后形成WDOWM算法,增强了种群多样性,维持种群多样性可以扩大算法搜索区域。然而,过分强调多样性的算法容易陷入盲目搜索状态,从而导致算法搜索效率低下,时间冗长。但从表6可以清晰看出,小波变异算子的加入并没有使算法的寻优时间增加很多,甚至在迭代次数为2000次的实例二中,WDOWM算法耗时小于WDO算法,这说明WDOWM算法的收敛速度快。

由表4可以看出,PSO算法最大旁瓣电平为-17.4173dB,而WDO算法最大旁瓣电平为-19.1133dB,相比PSO算法降低了1.696dB,WDOWM算法最大旁瓣电平为-19.2322dB,与WDO算法相比效果更优;由表5可以看出,PSO算法最大旁瓣电平为-13.2671dB,而WDO算法最大旁瓣电平为-15.0769dB,与PSO算法相比降低了1.8098dB,WDOWM算法最大旁瓣电平为-15.9968dB,与PSO算法相比降低了2.7297dB,效果更优。在指定方向生成零陷方面,3种算法均能在指定辐射度数上产生低于-50dB的零陷;然而WDOWM算法与PSO算法相比,55°方向零陷降低了0.538dB,57.5°方向零陷降低了4.1961dB,60°方向零陷降低了5.0763dB。综合分析表4和表5可以得出:WDOWM算法的性能优于WDO算法,同时优于PSO算法,且基于WDOWM算法的非等间距直线阵综合方法是有效可用的,适合在今后线阵综合问题中推广。

结束语 本文将文献[2]中提出的基本WDO算法进行了改进,引入了小波变异算子,加快了算法收敛速度,同时保持了种群多样性,使算法可以有效跳出局部最优,算法收敛精度得到了显著提高。针对算法参数设置易陷入因盲目选取而加大工作量的问题,文中采用二阶多因素多水平均匀设计法确定参数组合,从而提高了算法的科学性。将改进算法用于两个非等间距直线阵综合实例并进行了计算机仿真。仿真结果表明,WDO算法和WDOWM算法均适合用于线阵方向图综合问题,在收敛精度、收敛速度方面,WDOWM算法均优于WDO算法;在综合效果方面,WDOWM算法更具有应用性。本文算法可以很容易地推广到其他优化设计问题中。

参考文献

- [1] Bayraktar Z, Komurcu M, Werner D H. Wind Driven Optimization (WDO): A novel nature-inspired optimization algorithm and its application to electromagnetics [C] // 2010 IEEE Antennas and Propagation Society International Symposium (APSURSI). IEEE, 2010: 1-4
- [2] Bayraktar Z, Komurcu M, Bossard J A, et al. The wind driven optimization technique and its application in electromagnetics [J]. IEEE Transactions on Antennas and Propagation, 2013, 61 (5): 2745-2757
- [3] Bayraktar Z, Komurcu M, Jiang Z H, et al. Stub-loaded inverted-F antenna synthesis via wind driven optimization [C] // 2011 IEEE International Symposium on Antennas and Propagation (APSURSI). IEEE, 2011: 2920-2923
- [4] Bayraktar Z, Turpin J P, Werner D H. Nature-Inspired Optimization of High-Impedance Metasurfaces With Ultrasmall Interwoven Unit Cells [J]. Antennas and Wireless Propagation Letters, IEEE, 2011, 10: 1563-1566

- [5] Bhandari A K, Singh V K, Kumar A, et al. Cuckoo search algorithm and wind driven optimization based study of satellite image segmentation for multilevel thresholding using Kapur's entropy[J]. *Expert Systems with Applications*, 2014, 41(7): 3538-3560
- [6] Sun J, Wang X, Huang M, et al. A Cloud Resource Allocation Scheme Based on Microeconomics and Wind Driven Optimization [C] // 2013 8th ChinaGrid Annual Conference (ChinaGrid). IEEE, 2013: 34-39
- [7] He Da-kuo, Wang Fu-li, Jia Ming-xing. Uniform design genetic algorithm initial population and operational parameters [J]. *Journal of Northeastern University (Natural Science)*, 2005, 26(9): 828-831 (in Chinese)
何大阔, 王福利, 贾明兴. 遗传算法初始种群与操作参数的均匀设计[J]. *东北大学学报(自然科学版)*, 2005, 26(9): 828-831
- [8] Zhang Ke, Ling Hai-feng. Parameter Turning of Ant Colony Algorithm Based on Uniform Design and Chaos Theory[J]. *Computer Engineering*, 2012, 38(14): 141-143 (in Chinese)
张可, 凌海峰. 基于均匀设计和混沌理论的蚁群算法参数调整[J]. *计算机工程*, 2012, 38(14): 141-143
- [9] Zhang Xu-zhen, Jia Pin-gui. Unequally Spaced Linear Array Synthesis Using Modified DE[J]. *Computer Simulation*, 2013, 30(6): 226-229 (in Chinese)
张旭珍, 贾品贵. 基于改进 DE 的非等间距线阵综合[J]. *计算机仿真*, 2013, 30(6): 226-229
- [10] Chen K, He Z, Han C. A modified real GA for the sparse linear array synthesis with multiple constraints[J]. *IEEE Transactions on Antennas & Propagation*, 2006, 54(7): 2169-2173
- [11] Rani K N A, Malek F. Symmetric linear antenna array geometry synthesis using cuckoo search metaheuristic algorithm [C] // Conference on Communications. IEEE, 2011: 374-379
- [12] Khodier M M, Christodoulou C G. Linear Array Geometry Synthesis With Minimum Sidelobe Level and Null Control Using Particle Swarm Optimization[J]. *IEEE Transactions on Antennas & Propagation*, 2005, 53(8): 2674-2679
- [13] Ling S H, Iu H H C, Chan K Y, et al. Hybrid Particle Swarm Optimization With Wavelet Mutation and Its Industrial Applications[J]. *IEEE Transactions on Systems Man & Cybernetics Part B Cybernetics*, 2008, 38(3): 743-763
- [14] Xue Zheng-hui, Li Wei-ming, Ren Wu. Antenna array analysis and synthesis[M]. Beijing: Beihang University Press, 2011 (in Chinese)
薛正辉, 李伟明, 任武. 阵列天线分析与综合[M]. 北京: 北京航空航天大学出版社, 2011
- [15] Hou Wei, Dong Hong-bin, Yin Gui-sheng. Enhanced Multi-objective Evolutionary Algorithm Based on Decomposition [J]. *Computer Science*, 2014, 41(2): 114-118 (in Chinese)
侯薇, 董红斌, 印桂生. 一种改进的基于分解的多目标进化算法[J]. *计算机科学*, 2014, 41(2): 114-118
- [16] Fang Kai-tai. Uniform design and uniform design table[M]. Beijing: Science Press, 1994 (in Chinese)
方开泰. 均匀设计与均匀设计表[M]. 北京: 科学出版社, 1994

(上接第 254 页)

- [5] Li C H, Sun Z H. GridOF: An Efficient Outlier Detection Algorithm for very Large Datasets[J]. *Journal of Computer Research and Development*, 2003, 40(11): 1586-1592 (in Chinese)
李存华, 孙志辉. GridOF: 面向大规模数据集的高效离群点检测算法[J]. *计算机研究与发展*, 2003, 40(11): 1586-1592
- [6] Yu Dan-tong, Gholambosein S, Zhang Ai-dong. FindOut: Finding Outliers in Very Large Datasets[J]. *Knowledge and Information Systems*, 2002, 4(4): 387-412
- [7] Xue A R, Ju S G. Outlier Mining Based on Spatial Constraint [J]. *Computer Science*, 2007, 34(6): 207-210 (in Chinese)
薛安荣, 鞠时光. 基于空间约束的离群点挖掘算法[J]. *计算机科学*, 2007, 34(6): 207-210
- [8] Wang J C, Zhang J C, Jiang X Y. An Effective and Efficient Approach to Detect and Predict Outliers visually[J]. *Computer Science*, 2007, 34(6): 200-203 (in Chinese)
汪加才, 张金城, 江效尧. 一种有效的可视化孤立点发现与预测新途径[J]. *计算机科学*, 2007, 34(6): 200-203
- [9] Liu H, Wu J J, Su J Q. Differentiation Distance-based Outliers Detection Algorithm[J]. *Application Research of Computers*, 2010, 27(9): 3316-3318 (in Chinese)
刘欢, 吴介军, 苏锦旗. 基于分化距离的离群点检测算法[J]. *计算机应用研究*, 2010, 27(9): 3316-3318
- [10] Jiang F, Du J W, et al. Outlier Detection Based on Boundary and Distance[J]. *Acta Electronica Sinica*, 2010, 38(3): 700-705 (in Chinese)
江峰, 杜军威, 等. 基于边界和距离的离群点检测[J]. *电子学报*, 2010, 38(3): 700-705
- [11] Hu Y, Shi J, Wang C J, et al. Outlier Detection Algorithm based on Global Nearest Neighborhood[J]. *Journal of Computer Applications*, 2011, 31(10): 2778-2781 (in Chinese)
胡云, 施珺, 王崇骏, 等. 基于全局最近邻的离群点检测算法[J]. *计算机应用*, 2011, 31(10): 2778-2781
- [12] Pan Y, Zhang J B. Parallel Programming on Cloud Computing Platforms: Challenges and Solution [J]. *KITCS/FTRA Journal of Convergence*, 2012, 3(4): 23-28
- [13] Dean J, Ghemawat S. MapReduce: simplified data processing on large clusters [J]. *Communications of the ACM*, 2008, 51(1): 107-113
- [14] Yi X W, Li T R, et al. Performance Testing and Analysis among Different MapReduce Runtime Systems[J]. *Computer Science*, 2015, 42(5): 24-27 (in Chinese)
易修文, 李天瑞, 等. 不同 MapReduce 运行系统的性能测试与分析[J]. *计算机科学*, 2015, 42(5): 24-27
- [15] Subramanyam R B V, Sonam G. Map-Reduce Algorithm for Mining Outliers in the Large Data Sets using Twister Programming Model[J]. *International Journal of Computer Science and Electronics Engineering*, 2015, 3(1): 81-86
- [16] Liu X Y, Li J W, Yu H, et al. Adaptive Spectral Clustering Based on Shared Nearest Neighbors[J]. *Journal of Chinese Computer Systems*, 2011, 32(9): 1876-1880 (in Chinese)
刘馨月, 李静伟, 于红, 等. 基于共享近邻的自适应谱聚类[J]. *小型微型计算机系统*, 2011, 32(9): 1876-1880
- [17] Lu W, Shen Y, Chen S, et al. Efficient processing of k nearest neighbor joins using mapreduce[J]. *PVLDB*, 2012, 5(10): 1016-1027
- [18] Zhang C, Li F, Jests J. Efficient parallel kNN joins for large data in MapReduce[C] // Proc. of the 15th Int'l Conf. on Extending Database Technology. ACM Press, 2012: 38-49