

基于改进多目标飞蛾扑火算法的干扰资源优化方法

马铭希¹, 陈旭祎², 王绍祺², 刘成奎², 王超³

(1. 96716 部队, 江西赣州, 341005; 2. 空军指挥学院研究生大队, 北京, 100080; 3. 93787 部队, 北京, 102600)

摘要 干扰资源优化是当前电子战任务规划的重要环节, 针对多目标优化算法容易陷入局部最优及在三目标优化时的收敛问题, 提出一种基于改进多目标飞蛾扑火算法(TLWP-NSMFO)的多机干扰资源优化方法。首先在多目标飞蛾扑火算法的基础上利用 Tent 混沌映射完成种群初始化, 增加解的多样性和均匀性, 提高算法的搜索能力; 而后引入判定因子和 Lévy 飞行, 使得算法既能够以一定的概率接受当前解, 也能根据产生的扰动跳出当前解, 进行重新搜索, 增强了算法的搜索能力; 最后利用广泛分布参考点解决多目标飞蛾扑火算法在三目标函数的收敛性问题。仿真实验表明该算法比 MOEA/D 算法、NSMFO 算法具有更好的收敛性和种群多样性, 且该方法收敛结果稳定。

关键词 干扰资源优化; 多目标优化; 多目标飞蛾扑火算法; 混沌映射; Lévy 飞行; 广泛分布参考点

DOI 10.3969/j.issn.2097-1915.2025.04.012

中图分类号 V247.1⁺5; TN974 **文献标志码** A **文章编号** 2097-1915(2025)04-0100-10

A Jamming Resource Optimization Method Based on Improved Multi-Objective Moth-to-Flame Algorithm

MA Mingxi¹, CHEN Xuyi², WANG Shaoqi², LIU Chengkui², WANG Chao³

(1. Unit 96716, Ganzhou 341005, Jiangxi China; 2. Graduate Student Brigade, Air Force Command College, Beijing 100080, China; 3. Unit 93787, Beijing 102600, China)

Abstract Jamming resource optimization is an important link in current EW mission planning. Aimed at the problems that multi-objective optimization algorithm is easy to fall into local optimization and there is too much trouble in converges in three-objective optimization, a multi-aircraft jamming resource optimization method is proposed based on improved multi-objective moth-to-flame algorithm. Firstly, based on the multi-objective moth-to-flame algorithm, Tent chaotic map is utilized for initializing the population, increasing the diversity and uniformity of the solution and improving the search ability of the algorithm. And then, the induction of decision factor and Lévy flight is to make the algorithm accept not only the current solution with a certain probability, but also jump out of the current solution according to the disturbance and search again, enhancing the search ability of the algorithm. Finally, the widely distributed reference points are used to solve the convergence problem of the multi-objective moth-to-flame algorithm in the three-objective function. The simulation results show that this algorithm is better in convergence and population diversity than the MOEA/D algorithm and the NSMFO algorithm, and the convergence result of

收稿日期: 2024-10-09

作者简介: 马铭希(1998—), 男, 内蒙古包头人, 硕士, 研究方向为作战筹划与任务规划。E-mail: 18192289597@163.com

通信作者: 陈旭祎(1989—), 女, 北京人, 博士, 研究方向为指挥控制、联合进攻作战。E-mail: 312353233@qq.com

引用格式: 马铭希, 陈旭祎, 王绍祺, 等. 基于改进多目标飞蛾扑火算法的干扰资源优化方法[J]. 空军工程大学学报, 2025, 26(4): 100-109.
MA Mingxi, CHEN Xuyi, WANG Shaoqi, et al. A Jamming Resource Optimization Method Based on Improved Multi-Objective Moth-to-Flame Algorithm[J]. Journal of Air Force Engineering University, 2025, 26(4): 100-109.

this method is stable, achieving the purpose of assisting combat decision.

Key words jamming resource optimization; multi-objective optimization; multi-objective moth-to-flame algorithm; chaotic mapping; Lévy flight; widely distributed reference points

目前雷达的发展日新月异,其作用距离、方位角、仰角、分辨率及机动性等不断提高,加大了作战飞机执行任务的难度。必须采用多架干扰机协同作战,才能共同完成压制任务。如何对干扰资源进行分配,提升作战效能,使得作战设计更加合理,就形成了干扰资源优化这一问题。在现实作战场景中,问题的解决往往不仅仅需要满足一个约束条件,而是要进行综合考量。那么多目标优化模型应运而生。为解决多目标问题,智能优化算法开始蓬勃发展。目前,常见的多目标智能优化算法有 NSGA-II (non-dominated sorting genetic algorithm-II) 算法^[1]、MOPSO (multiple objective particle swarm optimization) 算法^[2]、MOEA/D (multi-objective evolutionary algorithm based on decomposition) 算法^[3]以及近年来提出的 NSMFO (non-dominated sorting moth flame optimization) 算法^[4]等。

针对这一问题,相关学者运用智能优化算法展开了相应研究。邢怀玺等^[5]针对传统灰狼算法无法解决多目标优化问题,提出多目标灰狼算法,从干扰功率、干扰效能及作战损失方面对干扰资源进行优化,解决了 GWO 算法探索能力弱、容易陷入局部最优的问题。高杨等^[6]将改进的离散布谷鸟算法应用到干扰资源分配问题中,解决了传统离散布谷鸟算法搜索速度慢精度低的问题。张养瑞等^[7]提出以“融合中心对航线各点定位精度的加权积分”为目标函数,并用遗传算法进行求解,对提高干扰机编队干扰效果有一定的可行性。文献[8]针对多干扰机协同干扰问题,结合干扰消隐区的形成原理,以干扰功率为目标函数,建立干扰资源分配模型,提出了一种基于自适应改进的遗传算法,并通过仿真实验证明了算法的有效性。文献[9]针对在特定作战态势下的无人机群协同干扰资源优化问题,以干扰效益、毁伤成本和射程成本为目标函数激进型建模,提出一种改进的多目标离散粒子群优化算法,通过对粒子编码及状态更新机制的调整,得到该问题的 pareto 解,证明了算法的有效性。文献[10]为了提高协同干扰机编队对雷达网的干扰效果,基于组网雷达融合中心的检测概率,建立了干扰资源优化分配模型。其次,提出了一种混合量子行为粒子群优化和自调整遗传算法,在多约束条件下对干扰资源进行优化,最后通过仿真实验验证了算法的有效性。

但上述文献中,对干扰资源优化问题的研究仍

旧存在一些不足。首先,文献[6~8]以及文献[10]对干扰资源优化问题的考虑仍旧以单目标函数或者其线性组合为评价依据,忽视了现实作战中部分目标存在互斥的现象;其次,文献[5]、文献[9]的算法采用非支配排序及拥挤距离排序的方式来解决多目标优化问题,其在处理三目标函数的问题中,由于计算量增加,会出现收敛性差、甚至无法收敛的问题。最后,上述文献采用的优化算法普遍存在收敛速度慢、精度低、操作复杂的问题。

NSMFO 算法在 2017 年提出,其保留了传统飞蛾扑火算法参数调节少,收敛速度快等优点,同时引入了 NSGA-II 算法的精英非支配排序方法和保持种群多样性的拥挤度排序方式,在三目标函数上仍旧存在着收敛问题。因此,本文提出基于混沌 Lévy 飞行及广泛分布参考点改进的多目标飞蛾扑火算法 (tent chaotic mapping, Lévy flight and widely distributed reference points improved non-dominated sorting moth-flame optimization algorithm, TLWP-NSMFO), 主要改进以下 3 个方面内容:

1) 首先通过 Tent 混沌映射,增加初始种群的多样性。

2) 引入判定算子,允许算法以一定概率接受当前解或通过 Lévy 飞行在飞蛾位置更新时进行扰动,产生新解,提高算法的寻优能力。

3) 运用广泛分布参考点代替拥挤距离排序,解决算法在三目标函数上由于计算量复杂而无法收敛的问题。

而后运用 TLWP-NSMFO 算法、MOEA/D 算法和 NSMFO 算法在 DTLZ 系列测试函数和仿真环境上进行对比实验,通过结果显示,TLWP-NSMFO 算法在收敛性和多样性上均优于其他算法,得出的解决方法能够起到辅助指挥员决策的作用。

1 干扰评估指标

1.1 决策变量设置与说明

针对基本作战场景,假设我方具有 M 架干扰机组成的任务编队,敌方有 N 部雷达组成的雷达网。我方采用“一对一”干扰策略,即每个雷达只受到 1 架干扰机的干扰,并且 1 架干扰机不能同时对多个雷达进行干扰,得到“0-1”分配矩阵。其中 $x_{ij} = 1$

为雷达受到干扰, $x_{ij}=0$ 为雷达没有受到干扰。

1.2 目标函数

1.2.1 干扰功率

本文以节约干扰资源为依据,以能达到压制目的为前提条件下最小化干扰功率为目标对干扰资源进行优化。

当雷达接收机在有干扰的条件下,接收的干扰功率表达式为:

$$P_{\text{jammer_radar}} = \frac{P_j G_j G(\phi) r_j B_r}{(4\pi)^2 R_j^2 L_j B_j} X_{ij} \quad (1)$$

其中: $P_{\text{jammer_radar}}$ 为雷达接收机接收到的干扰功率; P_j 为干扰机的干扰功率; G_j 为干扰机在干扰方向上的天线增益; $G(\phi)$ 为雷达天线在干扰机干扰方向上的增益; ϕ 为雷达天线主瓣方向与干扰机之间的夹角; r_j 为干扰机的极化损失; B_r 为雷达接收机的信号带宽; R_j 为雷达与干扰机之间的距离; L_j 为干扰机产生干扰信号的综合损耗; B_j 为干扰机信号带宽; X_{ij} 为决策变量。其中雷达天线在干扰机干扰方向上的增益 $G(\phi)$ 表达式为:

$$G(\phi) = \begin{cases} G, & 0 \leq |\phi| < \phi_{0.5} \\ K(2(\phi_{0.5}/\phi))^2 G, & \phi_{0.5} \leq |\phi| < 90^\circ \\ K(2(\phi_{0.5}/90^\circ))^2 G, & 90^\circ \leq |\phi| < 180^\circ \end{cases} \quad (2)$$

式中: K 为常数,通常取(0.04, 0.1); G 为敌方雷达天线增益; $\phi_{0.5}$ 为敌方雷达天线在半功率点处的波瓣宽度。

干扰机编队对雷达网总的干扰功率计算表达式为:

$$f_{\text{power}}(x) = \min \sum_{i=1}^N \sum_{j=1}^N P_{i, \text{jammer_radar}} \quad (3)$$

式中: N 为雷达的数量,其他符号均与上文表述一致。

1.2.2 干扰效能

干扰效能需要从多个角度进行评估,本文参考张翔等人的建模方式确定干扰效能矩阵^[11],引导时间因子为描述干扰机从接收到雷达信号到输出干扰信号的时间;干扰功率因子为描述干扰机可释放最大干扰功率的评价因子,其数值为干扰机最大释放功率的比值;干扰频带因子为描述干扰机工作带宽的评价因子,其数值为干扰机的工作带宽与敌方所有雷达工作总带宽的比值;干扰样式因子为描述干扰机干扰方式的评价因子,其数值为干扰机能够采取的干扰方式的值;雷达抗干扰措施因子为描述雷达抗干扰措施的评价因子,其数值是雷达能够采取的抗干扰方式的值。

记第 i 架干扰机对第 j 部雷达的干扰效能为:

$$e_{ij} = \beta_1 I_{iij} + \beta_2 I_{fij} + \beta_3 I_{pij} + \beta_4 I_{mij} + \beta_5 I_{aij} \quad (4)$$

式中: $\beta_1 I_{iij}$ 、 $\beta_2 I_{fij}$ 、 $\beta_3 I_{pij}$ 、 $\beta_4 I_{mij}$ 和 $\beta_5 I_{aij}$ 分别为引导时间因子、干扰功率因子、干扰频带因子、干扰样式因子、雷达抗干扰措施因子,由雷达和干扰机自身的性能决定,均为常数; β_i 为各因子所占权重, $\sum_{i=1}^5 \beta_i = 1$ 。同时干扰效能也受到敌方雷达威胁程度的影响,记敌方雷达的威胁因子为 ω_i ,则干扰机编队的总干扰效能表达式为:

$$f_{\text{efficiency}}(x) = \max \sum_{j=1}^N \omega_j [1 - \prod_{i=1}^M (1 - e_{ij}) x_{ij}] \quad (5)$$

式中: x_{ij} 为决策变量的分配矩阵。

1.2.3 探测半径

雷达的探测面积是在描述雷达性能时最为直观的表现,而雷达的探测面积直接由雷达探测半径所决定,通常,我们使用雷达方程对雷达探测模型进行表征,雷达的探测半径表征为:

$$R = \left(\frac{P_i G^2 \sigma \lambda^2}{(4\pi)^3 (\text{SNR}) F_n k T_0 B_n L} \right)^{1/4} \quad (6)$$

式中: R 为雷达的探测半径; P_i 为雷达发射功率的最大值; G 为雷达天线增益; σ 为目标雷达截面积; λ 为雷达发射波波长; SNR 为雷达信噪比; F_n 为噪声系数; k 为玻尔兹曼常数, $k = 1.38 \times 10^{-23} \text{ J/K}$; T 为雷达工作的工作环境开尔文温度; B_n 为噪声带宽; L 为系统损耗。

因此,本文选取干扰后雷达的探测半径为指标,将受到压制干扰后的雷达半径的最大值求和,作为评价干扰效果的目标函数。

综上可得敌方雷达在单个干扰机的干扰下,探测半径的表达式为:

$$R = \left(\frac{P_i G^2 \sigma \lambda^2}{(4\pi)^3 L F_n k T_0 B_n K_j + \frac{4\pi L P_j G_j G(\phi) r_j B_r K_j}{B_j L_j R_j^2}} \right)^{1/4} \quad (7)$$

式中: K_j 为干扰机对敌方雷达的压制系数。干扰机编队对雷达网总的干扰效果计算表达式为:

$$f_{\text{radius}}(x) = \min \sum_{j=1}^N \max R_{j, \text{compression}} x_{ij} \quad (8)$$

1.3 多目标干扰资源优化模型

干扰资源优化这一问题归根结底也是多目标优化问题,旨在通过对干扰资源进行分配,使得既保证干扰机的自身安全,同时也达到作战需求。

因此,构建多目标优化模型如下:

$$Z = F(X) = \begin{pmatrix} f_1(x) = f_{\text{power}}(x) \\ f_2(x) = f_{\text{efficiency}}(x) \\ f_3(x) = f_{\text{radius}}(x) \end{pmatrix} \quad (9)$$

$$\text{s. t. } \phi(X) = \left\{ \begin{array}{l} x_{ij} = 0, 1, \dots, N_{\text{jammer}} \\ x_{i,j} \neq x_{i,j+1} \neq x_{i,j+2} \neq \dots \neq x_{i,j+N_{\text{jammer}}} \\ i = 1, 2, \dots, N_{\text{radar}}; j = 1, 2, \dots, N_{\text{jammer}} \end{array} \right\} \quad (10)$$

以最小化干扰功率 $\min(f_{\text{power}}(x))$, 最大化干扰效能 $\max(f_{\text{efficiency}}(x))$, 最小化干扰半径 $\min(f_{\text{radius}}(x))$ 为目标函数, 建立多目标干扰资源优化模型, 为了便于求解, 将“最大化”目标函数转换为“最小化”目标函数, 即 $\max(f_{\text{efficiency}}(x))$ 转换为 $\min(-f_{\text{efficiency}}(x))$ 。

2 多目标干扰资源优化方法

2.1 基本飞蛾扑火算法

飞蛾扑火算法 (moth-flame optimization algorithm, MFO) 是 Seyedali Mirjalili 等人在 2015 年提出的一种新型智能优化算法^[12], 主要用于解决单目标优化问题。初始火焰的集合与飞蛾种群集合大小相同, 但随着后续的迭代逐渐递减, 其自适应更新机制为:

$$\text{flameno} = \text{round}\left(N - \text{iteration} \times \frac{N - 1}{\text{max_iteration}}\right) \quad (11)$$

式中: N 为最大火焰数量; iteration 为当前迭代次数; max_iteration 为算法最大迭代次数。火焰数量随算法迭代次数的变化如图 1 所示。

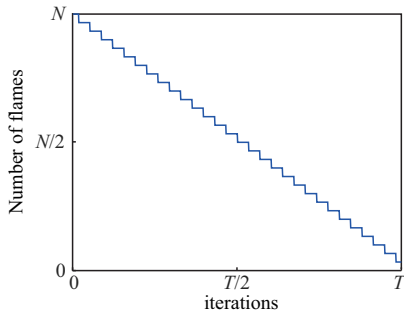


图 1 火焰自适应更新策略

Fig. 1 Flame adaptive update strategy

MFO 算法采用了飞蛾在人造光源下的飞行原理。根据飞蛾的趋光性, 将飞蛾的飞行方式用对数螺旋曲线进行拟合, 飞蛾 M_i 围绕其对应的火焰 F_i 进行对数螺旋运动, 定义飞蛾位置的更新公式为:

$$S(M_i, F_j) = D_i \cdot e^{bt} \cdot \cos(2\pi t) + F_j \quad (12)$$

式中: $S(M_i, F_j)$ 为飞蛾 M_i 更新后的位置; b 为常数参数, 其值的大小决定对数等角螺旋线的形状, 通常取 1; 参数 t 为 $[-1, 1]$ 范围内的随机数, 其取值的大小控制着飞蛾与火焰的距离, t 值越小, 飞蛾离火焰距离越近, 通过改变参数 t 的取值, 调整飞蛾最终

到达火焰周围的位置, 反映了算法的局部寻优能力; D_i 为第 j 个火焰与第 i 只飞蛾的欧式距离; $D_i = |M_i - F_j|$, 飞蛾的运动轨迹^[13] 如图 2 所示。

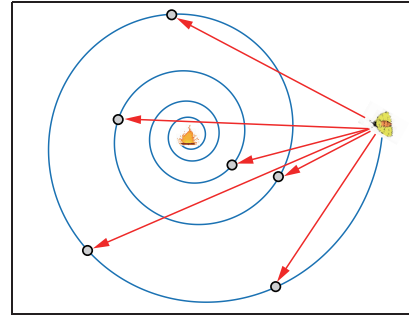


图 2 飞蛾位置更新示意图

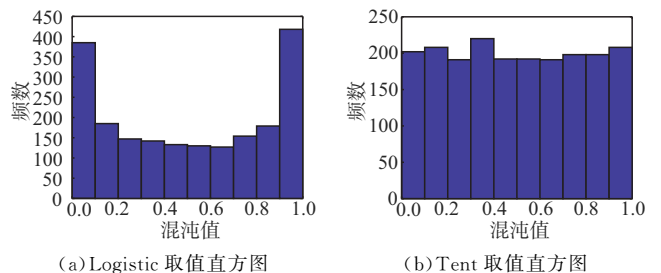
Fig. 2 The trajectories of moths

2.2 多目标飞蛾扑火算法

多目标飞蛾扑火算法 (non-dominated sorting moth flame optimization, NSMFO) 是以多个目标函数为优化对象, 同时对这些目标函数进行优化以期同时达到最优状态。其引入了 NSGA-II 算法中的非支配快速排序和拥挤距离进行优化, 使得算法具有处理多目标问题的能力。但是, 由于其种群初始化, 更新机制也没有发生变化, 仍存在着容易陷入局部最优等问题, 同时, 由于运用拥挤距离的筛选方式, 算法在边界上容易发生重复搜索, 过度保留的现象, 使得算法在处理三目标问题时出现收敛速度慢、甚至无法收敛的问题。因此提出下列优化方式, 对 NSMFO 算法进行优化。

2.2.1 混沌映射

在优化领域, 混沌映射因其有遍历性、随机性以及非线性等特点被广泛应用到智能优化算法当中^[14]。目前常见的几种混沌映射有 Logistic 混沌映射、Tent 混沌映射、Fuch 混沌映射、Chebyshev 混沌映射及 Sine 映射, 其映射图如图 3 所示^[15-18]。在迭代次数为 2 000 时, Tent 混沌映射在取值区间的分布更加均匀, 具有更好的遍历性和随机性, 因此本文使用 Tent 混沌映射对 NSMFO 算法的种群初始化进行改进, 使其初始种群分布更加广泛, 提高算法的全局搜索能力。



(a) Logistic 取值直方图

(b) Tent 取值直方图

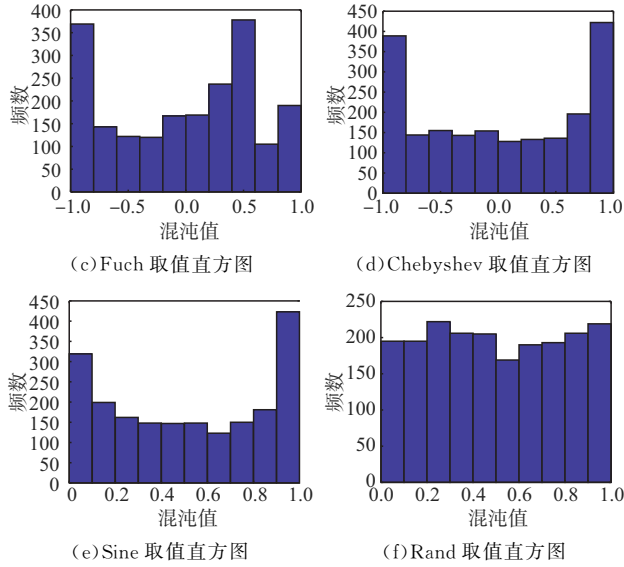


图 3 混沌映射取值直方图

Fig. 3 Histogram of chaotic mapping values

2.2.2 Lévy 飞行

由于传统 NSMFO 算法在搜索过程中采用等角对数螺线的搜索方式,虽然搜索速度快,但是在一定程度上容易使算法陷入局部最优。而 Paul Pierre Lévy 最早发现 Lévy 飞行,指出其在随机游走的过程中具有相对较高的概率出现大跨步^[19]。相比于布朗运动,Lévy 飞行具有更好的发散性,Mantegan 在 1994 年提出生成服从莱维分布的随机步长的方法为^[20]:

$$s = \frac{\mu}{|v|^{\frac{1}{\beta}}} \quad (13)$$

式中: s 为获取的随机步长; $\mu \in N(0, \sigma^2)$; $v \in N(0, \sigma^2)$;

$$\sigma = \left\{ \frac{\Gamma(1-\beta) \sin\left(\frac{\pi\beta}{2}\right)}{\beta \Gamma\left(\frac{1+\beta}{2}\right) 2^{\frac{\beta-1}{2}}} \right\}^{\frac{1}{\beta}}。 \text{Lévy 飞行示意图如图 4 所示。}$$

图 4 所示。

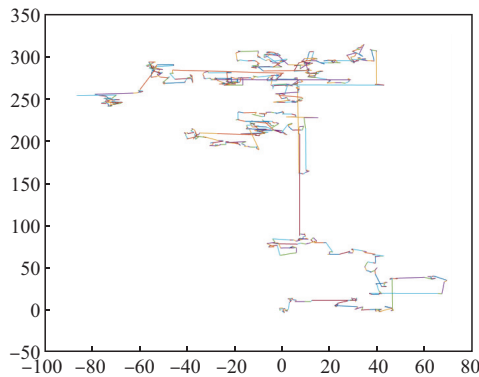


图 4 Lévy 飞行示意图

Fig. 4 Diagram of Lévy's flight

因此,本文选用 Lévy 飞行在飞蛾更新位置后对其进行扰动,位置更新为:

$$X(t+1) = X(t) \cdot \text{Lévy} \quad (14)$$

同时,引入判定算子 $\zeta = 0.5$,允许算法以一定的概率接受当前的劣解,使得算法能够跳出局部最优,搜寻到全局最优解。

2.2.3 广泛分布参考点

广泛分布参考点是在决策空间中均匀分布的一组点,其与理想点连接构成的参考向量将决策空间均匀的分割,通过计算种群中的个体与各参考向量之间的距离以保证算法的多样性^[21]。

1)建立参考点。首先参考点建立在一个 $(M-1)$ 维的超平面上, M 为目标空间的维度, H 为需要进行划分的份数,由这两者决定参考点的数目 P :

$$P = C_{M+H-1}^H \quad (15)$$

$$\text{参考点集合为 } S_j = \left\{ \frac{0}{H}, \frac{1}{H}, \dots, \frac{H}{H} \right\}, \sum_{j=1}^M S_j = 1。$$

2)确定参考点的位置坐标。在确定位置参考点坐标时首先构建 $(M-1)$ 维的 x 的集合, $\left\{ \frac{0}{H}, \frac{1}{H}, \dots, \frac{H+M-2}{H} \right\}$, 然后按照 $x_{ij} = x_{ij} - \frac{j-1}{H}$ 对 x 进行更新,最后获得参考点位置坐标。

3)进行种群个体的自适应归一化。首先构建种群的理想点 $z = (z_1^{\min}, z_2^{\min}, \dots, z_n^{\min})$, 之后按照 $f'_i(x) = f_i(x) - z_i^{\min}$ 对目标函数进行转换;然后求解得出每个坐标轴对应的额外点 $\text{ASF}(x, \omega)$, 进而构造出超平面,公式为:

$$\text{ASF}(x, \omega) = \max_{i=1}^M f'_i(x) / \omega_i \quad (16)$$

式中: ω_i 为目标函数进行标量化的权向量。

最后利用超平面与坐标轴的截距,对目标函数值进行自适应归一化如下:

$$f_i^n(x) = (f'_i(x) - z_i^{\min}) / (a_i - z_i^{\min}) \quad (17)$$

式中: a_i 为第 i 个目标轴在超平面上的截距。

这里归一化平面与坐标轴交点的函数值为 1,

此时归一化超平面上的点满足 $\sum_{i=1}^m f_i^n = 1$ 。

4)构建参考点与种群个体的距离关系。我们连接参考点与原点,构建射线,种群个体到射线的距离作为两者之间的距离关系,距离越小,种群越能被选中。

5)在考虑前 $(l-1)$ 前沿等级的情况下,根据参考点被选择的个数,选择参考点被选择较少的作为最后一个前沿种群个体选择的依据,进而实现选择。

2.3 改进的多目标飞蛾扑火算法

综上,为提升算法对复杂多目标问题的求解能力,基于传统飞蛾扑火算法的基本原理,结合多目标

优化特性进行改进。图 5 为改进后的多目标飞蛾扑火算法流程,其通过引入混沌初始化、帕累托前沿,适应度函数优化等机制,清晰呈现算法从初始化到迭代寻优的完整逻辑框架,为多目标优化问题提供了更有效的解决路径。

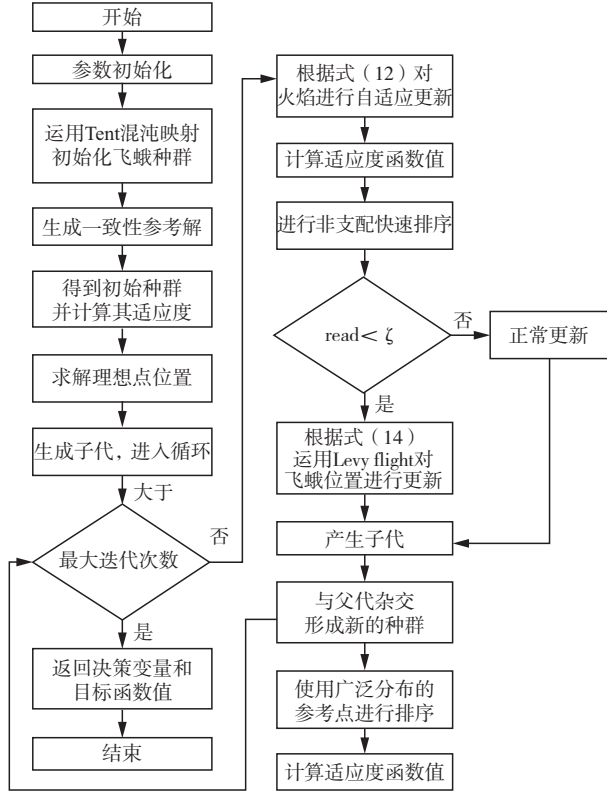


图 5 TLWP-NSMFO 算法流程

Fig. 5 Process of the TLWP-NSMFO Algorithm

3 仿真分析

3.1 DTLZ 系列测试函数及评价指标

DTLZ 系列测试函数如表 1 所示。

表 1 DTLZ 系列测试函数

Tab. 1 DTLZ series of test functions

测试函数	表达式
	$f_1(x) = \frac{1}{2}(1 + g(x))x_1x_2$
	$f_2(x) = \frac{1}{2}(1 + g(x))x_1(1 - x_2)$
DTLZ1	$f_3(x) = \frac{1}{2}(1 + g(x))(1 - x_1)$
	$g(x) = 100 \times \left\{ 5 + \sum_{i=3}^{\text{Dim}} [(x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5))] \right\}$
维度	Dim=7
区间	$x = (x_1, x_2, \dots, x_{\text{Dim}})^T \in [0, 1]$

续表

测试函数	表达式
	$f_1(x) = (1 + g(x))\cos\left(\frac{x_1\pi}{2}\right)\cos\left(\frac{x_2\pi}{2}\right)$
	$f_2(x) = (1 + g(x))\cos\left(\frac{x_1\pi}{2}\right)\sin\left(\frac{x_2\pi}{2}\right)$
DTLZ2	$f_3(x) = (1 + g(x))\sin\left(\frac{x_1\pi}{2}\right)$
	$g(x) = 100 \times \left\{ 10 + \sum_{i=3}^{\text{Dim}} [(x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5))] \right\}$
维度	Dim=12
区间	$x = (x_1, x_2, \dots, x_{\text{Dim}})^T \in [0, 1]$
	$f_1(x) = (1 + g(x))\cos\left(\frac{x_1\pi}{2}\right)\cos\left(\frac{x_2\pi}{2}\right)$
	$f_2(x) = (1 + g(x))\cos\left(\frac{x_1\pi}{2}\right)\sin\left(\frac{x_2\pi}{2}\right)$
DTLZ3	$f_3(x) = (1 + g(x))\sin\left(\frac{x_1\pi}{2}\right)$
	$g(x) = 100 * \left\{ 10 + \sum_{i=3}^{\text{Dim}} [(x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5))] \right\}$
维度	Dim=12
区间	$x = (x_1, x_2, \dots, x_{\text{Dim}})^T \in [0, 1]$

为了评价不同算法的性能,采用反世代距离 (inverted generational distance, IGD) 作为评价指标。IGD 是通过计算算法求解得出的点与真实 pareto 解集中点的欧式距离的平均值作为衡量算法性能的标准,IGD 的值越小,则算法的性能更好。

$$\text{IGD}(P, P^*) = \frac{\sum_{x \in P^*} \min_{y \in P} \text{dis}(x, y)}{|P^*|} \quad (18)$$

3.2 多目标优化算法在测试函数上的结果

由图 6~图 8 和测试结果表 2 可知, TLWP-NSMFO 算法在三目标的 DTLZ 系列函数上能够比其他 2 种算法更少的迭代次数获得更好的结果, 其 IGD 的值相比于其他 2 种算法更小, 因此其综合性能更好, 证明 TLWP-NSMFO 算法具有更好的收敛性和多样性。MOEA/D 算法采用的基于分解的方式、NSMFO 算法采用拥挤距离的方式、TLWP-NSMFO 算法采用广泛分布参考点的方式分别对进行非支配排序后的解进一步排序, 通过对比实验可以看出, TLWP-NSMFO 算法的排序方式更优, 而且能够有效解决 NSMFO 算法在边界上反复搜索、无法收敛的情况。证明了本文提出的 TLWP-NSMFO 的有效性。

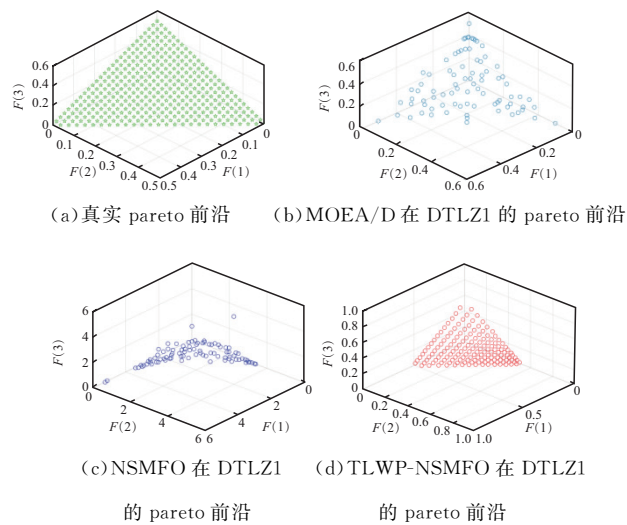


图 6 不同算法在 DTLZ1 函数上的测试结果

Fig. 6 Test results of different algorithms on DTLZ1 function

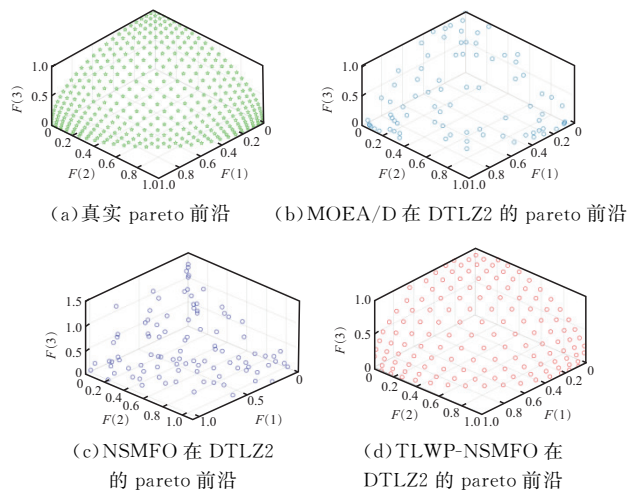


图 7 不同算法在 DTLZ2 函数上的测试结果

Fig. 7 Test results of different algorithms on DTLZ2 function

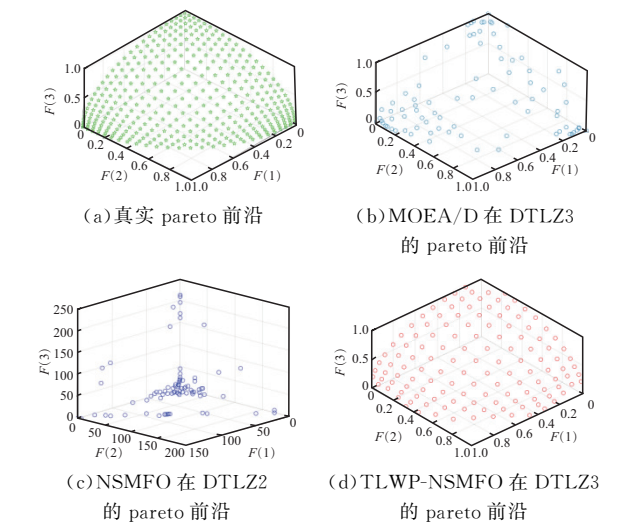


图 8 不同算法在 DTLZ3 函数上的测试结果

Fig. 8 Test results of different algorithms on DTLZ3 function

表 2 不同测试函数在 DTLZ 系列函数上的测试结果
Tab. 2 Test results of different test functions on DTLZ series functions

测试函数名称	算法名称	迭代次数	IGD
DTLZ1	MOEA/D	3 000	$3.433\ 9\times10^{-2}$
	NSMFO	3 000	$7.323\ 9\times10^{-1}$
	TLWP-NSMFO	2 000	$2.234\ 9\times10^{-2}$
DTLZ2	MOEA/D	1 000	$6.791\ 6\times10^{-2}$
	NSMFO	3 000	$7.575\ 9\times10^{-2}$
	TLWP-NSMFO	2 000	$3.865\ 1\times10^{-2}$
DTLZ3	MOEA/D	3 000	$8.505\ 2\times10^{-2}$
	NSMFO	3 000	6.987 1
	TLWP-NSMFO	2 000	$4.793\ 3\times10^{-2}$

3.3 多目标优化算法的干扰资源优化分析

假设某干扰编队承担对敌方雷达干扰任务,由上文中提出,采用“一对一”干扰策略,为验证 TLWP-NSMFO 算法的可行性,进行以下仿真实验。仿真计算机的 CPU 为 Intel Core i7-9750H CPU 2.60 GHz,操作系统为 Windows 10,仿真软件为 MATLAB R2021a。雷达性能参数见表 3,干扰机性能参数见表 4。

表 3 雷达性能参数
Tab. 3 Radar performance parameter

雷达编号	P_t	L	λ	B_n
1	1.58×10^{-6}	5.0	0.50	1.80×10^{-5}
2	1.80×10^{-6}	4.5	0.65	3.00×10^{-5}
3	1.50×10^{-6}	3.8	0.65	2.90×10^{-5}
4	2.30×10^{-6}	3.5	0.60	3.50×10^{-5}

表 4 干扰机性能参数
Tab. 4 Jammer performance parameter

干扰机编号	P_j	G_j	r_j	L_j
1	1.00×10^{-6}	2.0	1.0	20
2	1.80×10^{-6}	3.0	1.5	30
3	2.00×10^{-6}	2.4	2.0	25
4	1.00×10^{-6}	4.0	3.0	35

此外,雷达天线增益 $G=40$,干扰信号带宽 $B_j=2\times10^6$,雷达接收机信号带宽 $B_r=2\times10^6$,噪声系数 $F_n=4$,温度 $T=291\text{ K}$,雷达和干扰机的距离 $d=35\text{ km}$ 。

实验 1 少雷达目标下的干扰资源优化及对比
仿真参数设置:种群规模 $N_{\text{moth}}=100$,迭代次数 $\text{max_iteration}=100$,雷达数量 $N_{\text{radar}}=4$,干扰机数量 $M_{\text{jammer}}=4$,等角对数螺线参数 $b=1$ 。

由图 9、图 10、表 5、表 6、表 7 及表 8 可知,虽然

NSMFO 算法能够求解多目标干扰资源优化问题,但是从实验结果看,其方案的数量及结果劣于 TLWP-NSMFO 算法;反观 TLWP-NSMFO 算法能够较好的求解多目标干扰资源优化问题。从图 10 中可以看出非劣解的分布较为分散,证明算法具有较好的多样性;从表中可以看出各个解互不支配,针对每个方案干扰功率、干扰效能、干扰效果的值均不相同,在实际应用中需要决策者根据作战需求,从表 8 中选择表 7 结果对应的分配方案。

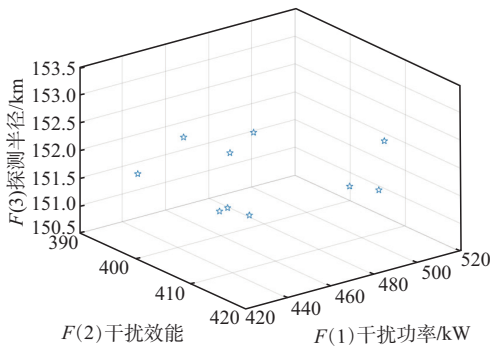


图 9 TLWP-NSMFO 算法非劣解分布图

Fig. 9 TLWP-NSMFO algorithm non-inferior solution distribution

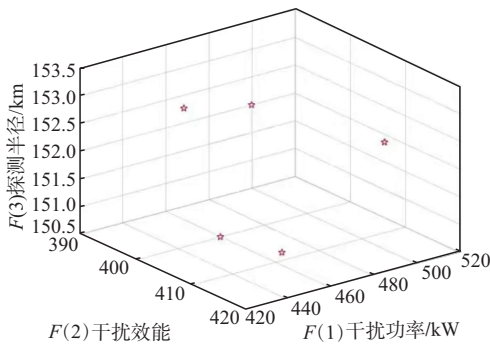


图 10 NSMFO 算法非劣解分布图

Fig. 10 NSMFO algorithm non-inferior solution distribution

表 5 TLWP-NSMFO 算法非劣解对应的目标函数值

Tab. 5 Objective function values corresponding to pareto solutions of TLWP-NSMFO algorithm

方案	干扰功率	干扰效能	干扰效果
1	425.059 4	406.955	152.957 3
2	427.867 0	414.258	152.980 8
3	432.100 0	416.865	153.423 3
4	432.110 4	410.700	151.723 0
5	435.393 1	394.622	151.625 4
6	466.528 8	398.711	150.872 3
7	466.982 3	402.480	150.906 6
8	476.582 9	416.872	151.985 9
9	487.546 4	418.866	152.778 2
10	511.454 9	408.528	151.169 5

表 6 TLWP-NSMFO 算法干扰分配方案

Tab. 6 Jamming resources optimization allocation of TLWP-NSMFO algorithm

方案	Jammer 1	Jammer 2	Jammer 3	Jammer 4
1	Radar 4	Radar 2	Radar 1	Radar 3
2	Radar 2	Radar 3	Radar 1	Radar 2
3	Radar 3	Radar 4	Radar 1	Radar 2
4	Radar 4	Radar 1	Radar 2	Radar 3
5	Radar 4	Radar 3	Radar 2	Radar 1
6	Radar 1	Radar 3	Radar 2	Radar 4
7	Radar 1	Radar 4	Radar 2	Radar 3
8	Radar 4	Radar 1	Radar 3	Radar 2
9	Radar 3	Radar 1	Radar 4	Radar 2
10	Radar 1	Radar 4	Radar 3	Radar 2

表 7 NSMFO 算法非劣解对应的目标函数值

Tab. 7 Objective function values corresponding to pareto solutions of NSMFO algorithm

方案	干扰功率	干扰效能	干扰效果
1	448.258 3	410.833 0	152.468 9
2	500.067 5	395.689 0	152.178 2
3	510.689 5	405.562 5	151.895 6
4	459.921 1	401.600 0	151.963 0
5	463.462 1	409.221 0	153.278 5

表 8 NSMFO 算法干扰分配方案

Tab. 8 Jamming resources optimization allocation of NSMFO algorithm

方案	Jammer 1	Jammer 2	Jammer 3	Jammer 4
1	Radar 4	Radar 2	Radar 3	Radar 3
2	Radar 2	Radar 4	Radar 3	Radar 1
3	Radar 1	Radar 3	Radar 4	Radar 2
4	Radar 3	Radar 4	Radar 1	Radar 2
5	Radar 4	Radar 1	Radar 2	Radar 3

实验 2 多雷达目标下的干扰资源优化及对比

仿真参数设置:种群规模 $N_{\text{moth}}=100$,迭代次数 $\text{max_iteration}=100$,雷达数量 $N_{\text{radar}}=10$,干扰机数量 $M_{\text{jammer}}=10$,等角对数螺线参数 $b=1$ 。因为在实验 1 中仅有 4 部雷达,4 架干扰机,在“一对一”干扰策略下,仅存在 24 种干扰方案;而在本次实验中设置 10 部雷达,10 架干扰机,在“一对一”干扰策略下,存在 3 628 800 种方案,更能很好的证明算法的性能。同时为了验证改进措施的有效性,与 NSMFO 算法形成对比实验,结果如图 11 所示。

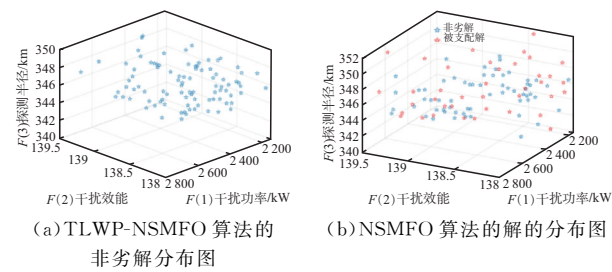


图 11 2 种算法的非劣解分布图

Fig. 11 Distribution of pareto solutions of the two algorithms

表 9 不同算法在 3 种情况下的目标函数值

Tab. 9 Objective function values of different algorithms under three conditions

方案	干扰功率		干扰效能		干扰效果	
	TLWP-NSMFO	NSMFO	TLWP-NSMFO	NSMFO	TLWP-NSMFO	NSMFO
1	2 145.138	2 148.959	138.209	138.063	346.769 5	346.342 9
2	2 568.793	2 579.808	139.479	139.436	347.158 8	343.578 3
3	2 345.409	2 440.005	138.821	138.217	341.228	341.468

表 10 不同算法在 3 种情况下的分配方案

Tab. 10 Allocation schemes of different algorithms in three cases

方案		Jammer1	Jammer2	Jammer3	Jammer4	Jammer5	Jammer6	Jammer7	Jammer8	Jammer9	Jammer10
TLWP-NSMFO 算法	1	R4	R3	R10	R6	R9	R7	R8	R2	R1	R5
	2	R2	R4	R5	R10	R1	R7	R3	R6	R8	R9
	3	R6	R1	R3	R5	R4	R7	R9	R10	R8	R2
NSMFO 算法	1	R4	R9	R7	R6	R3	R10	R2	R5	R1	R8
	2	R6	R5	R9	R10	R1	R9	R8	R7	R2	R5
	3	R1	R4	R2	R6	R5	R3	R7	R9	R8	R10

由图 11、表 9、表 10 可看出虽然 NSMFO 算法能够对干扰资源优化问题求解,但是由于其拥挤距离排序的机制,在同样的迭代次数下,仍旧存在被支配的解的情况。同时,算法求解的结果劣于本文提出的 TLWP-NSMFO 算法,证明了本文算法改进的有效性。

4 结论

本文针对多机干扰资源优化问题展开研究,提出基于 TLWP-NSMFO 算法的干扰资源优化方法。首先对决策变量进行描述,而后确定多机干扰资源优化的 3 个目标函数,一是干扰功率,二是干扰效能,三是探测半径,建立了多目标干扰资源优化模型。同时,针对 NSMFO 算法在搜索过程中收敛速度较慢、在处理三目标问题时甚至无法收敛进行了改进:

1) Tent 混沌映射初始化使得算法初期搜索的广泛性更强,增强了算法的全局搜索能力,加快了算法的收敛速度。

由图 11 可知 TLWP-NSMFO 算法在问题维度增加后,仍旧能够较好地求解多目标干扰资源优化问题,得出互不支配的解集。针对每个方案干扰功率、干扰效能、干扰效果的值均不相同,在实际应用中需要决策者根据作战需求选择分配方案。由于解集较多,不能一一列举,此处分别将干扰功率、干扰效能、干扰效果各自达到需求时的方案及目标函数值进行列举,如表 9、表 10 所示。

2)引入判定因子和 Lévy 飞行在飞蛾位置更新时产生扰动,使得算法既能够以一定概率接受当前的劣解,同时也能跳出局部最优,增强算法的求解能力。

3)采用广泛分布参考点代替拥挤距离排序,使得算法在处理三目标问题时能够有效收敛。

最后,通过测试函数证明,TLWP-NSMFO 算法性能均优于 MOEA/D、NSMFO 算法,并且在多目标干扰资源优化中能够较好的解决问题。下一步研究中,将提出算法集成到任务规划软件中去,从而实现工程化应用。

参考文献

[1] DEB K, AGRAWAL S, PRATAP A, et al. A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II[C]// Parallel Problem Solving from Nature PPSN VI. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000: 849-858.

[2] COELLO C A, LECHUGA M S. MOPSO: A Proposal for Multiple Objective Particle Swarm Optimization [C]// Proceedings of the 2002 Congress on Evolution-

- ary Computation. Honolulu, HI: IEEE, 2002: 1051-1056.
- [3] ZHANG Q F, LI H. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition [J]. IEEE Transactions on Evolutionary Computation, 2007, 11(6): 712-731.
- [4] SAVSANI V, TAWHID M A. Non-Dominated Sorting Moth Flame Optimization (NS-MFO) for Multi-Objective Problems [J]. Engineering Applications of Artificial Intelligence, 2017, 63: 20-32.
- [5] 邢怀玺, 吴华, 陈游, 等. 基于多目标灰狼算法的干扰资源多效能优化方法 [J]. 北京航空航天大学学报, 2020, 46(10): 1990-1998.
- XING H X, WU H, CHEN Y, et al. Multi-Efficiency Optimization Method of Jamming Resource Based on Multi-Objective Grey Wolf Optimizer [J]. Journal of Beijing University of Aeronautics and Astronautics, 2020, 46(10): 1990-1998. (in Chinese)
- [6] 李东生, 高杨, 雍爱霞. 基于改进离散布谷鸟算法的干扰资源分配研究 [J]. 电子与信息学报, 2016, 38(4): 899-905.
- LI D S, GAO Y, YONG A X. Jamming Resource Allocation via Improved Discrete Cuckoo Search Algorithm [J]. Journal of Electronics & Information Technology, 2016, 38(4): 899-905. (in Chinese)
- [7] 张养瑞, 李云杰, 高梅国. 协同干扰资源优化分配模型及算法 [J]. 系统工程与电子技术, 2014, 36(9): 1744-1749.
- ZHANG Y R, LI Y J, GAO M G. Optimal Assignment Model and Solution of Cooperative Jamming Resources [J]. Systems Engineering and Electronics, 2014, 36(9): 1744-1749. (in Chinese)
- [8] ZHANG L, SHI G Q, GENG X T. Blanket Jamming Targets Assignment Based on Adaptive Genetic Algorithm [C]//2019 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM). Bangkok: IEEE, 2019: 171-175.
- [9] QIN Q W, DONG W F, LIN M Q, et al. Cooperative Jamming Resource Allocation of Uav Swarm Based on Multi-Objective DPSO [C]//2018 Chinese Control and Decision Conference (CCDC). Shenyang, IEEE, 2018: 5319-5325.
- [10] JIANG H Q, ZHANG Y R, XU H Y. Optimal Allocation of Cooperative Jamming Resource Based on Hybrid Quantum-Behaved Particle Swarm Optimisation and Genetic Algorithm [J]. IET Radar, Sonar & Navigation, 2017, 11(1): 185-192.
- [11] 张翔, 吴华, 陈游. 基于人工蜂群的协同干扰资源优化方法 [C]//2018 全国电子战大会论文集, 北京: 中国电子学会, 2018: 57-61.
- ZHANG X, WU H, CHEN Y. Cooperative Jamming Resource Optimization Method Based On Artificial Bee Colony [C]//Proceedings of the 2018 National Electronic Warfare Conference. Beijing: Chinese Institute of Electronics, 2018: 57-61. (in Chinese)
- [12] MIRJALILI S. Moth-Flame Optimization Algorithm: A Novel Nature-Inspired Heuristic Paradigm [J]. Knowledge-Based Systems, 2015, 89: 228-249.
- [13] 黄鹤, 吴琨, 王会峰, 等. 基于改进飞蛾扑火算法的无人机低空突防路径规划 [J]. 中国惯性技术学报, 2021, 29(2): 256-263.
- HUANG H, WU K, WANG H F, et al. Path Planning of UAV Low Altitude Penetration Based on Improved Moth-Flame Optimization [J]. Journal of Chinese Inertial Technology, 2021, 29(2): 256-263. (in Chinese)
- [14] VAROL ALTAY E, ALATAS B. Bird Swarm Algorithms with Chaotic Mapping [J]. Artificial Intelligence Review, 2020, 53(2): 1373-1414.
- [15] BINGOL H, ALATAS B. Chaotic League Championship Algorithms [J]. Arabian Journal for Science and Engineering, 2016, 41(12): 5123-5147.
- [16] ALATAS B. Chaotic Bee Colony Algorithms for Global Numerical Optimization [J]. Expert Systems with Applications, 2010, 37(8): 5682-5687.
- [17] ALATAS B. Chaotic Harmony Search Algorithms [J]. Applied Mathematics & Computation, 2010, 216(9): 2687-2699.
- [18] 岳龙飞, 杨仁农, 张一杰, 等. Tent 混沌和模拟退火改进的飞蛾扑火优化算法 [J]. 哈尔滨工业大学学报, 2019, 51(5): 49-55.
- YUE L F, YANG R N, ZHANG Y J, et al. Tent Chaos and Simulated Annealing Improved Moth Fire Suppression Optimization Algorithm [J]. Journal of Harbin Institute of Technology, 2019, 51(5): 49-55.
- [19] LEVY P. Théorie de l'addition des variables aléatoires [M]. Paris: Gauthier-Villars, 1937.
- [20] YANG X S, TERRAZAS. A New Metaheuristic Bat-Inspired Algorithm [J]. Nature Inspired Cooperative Strategies for Optimization (nicso 2010), 2010, 284: 65-74.
- [21] 毕晓君, 王朝. 一种基于参考点约束支配的 NSGA-III 算法 [J]. 控制与决策, 2019, 34(2): 369-376.
- BI X J, WANG C. A Reference Point Constrained Dominance-Based NSGA-III Algorithm [J]. Control and Decision, 2019, 34(2): 369-376. (in Chinese)

(编辑: 刘勇)