

# 求解 0-1 背包问题的烟花算法<sup>①</sup>



徐小平<sup>1</sup>, 庞润娟<sup>1</sup>, 王峰<sup>2</sup>, 钱富才<sup>3,4</sup>

<sup>1</sup>(西安理工大学 理学院, 西安 710054)

<sup>2</sup>(西安交通大学 数学与统计学院, 西安 710049)

<sup>3</sup>(西安理工大学 自动化与信息工程学院, 西安 710048)

<sup>4</sup>(西安卫星测控中心 宇航动力学国家重点实验室, 西安 710043)

通讯作者: 庞润娟, E-mail: 2572180564@qq.com

**摘要:** 为了克服现有方法在求解 0-1 背包问题时存在的缺陷, 提出了一种改进的烟花算法. 在给出 0-1 背包问题的数学模型后, 利用 Kent 混沌映射对基本烟花算法的解初始化以使初始位置分布更加均匀, 同时引入 Sigmoid 函数得到渐变的爆炸半径使得算法的求解精度与搜索速度达到某种平衡, 用改进的烟花算法来对其进行求解. 通过对典型测试函数和 0-1 背包问题的求解结果说明了所提出的改进烟花算法求解精度更高, 性能更加稳定.

**关键词:** 0-1 背包问题; 优化; 烟花算法; 混沌映射; 渐变爆炸半径

引用格式: 徐小平, 庞润娟, 王峰, 钱富才. 求解 0-1 背包问题的烟花算法. 计算机系统应用, 2019, 28(2): 164-170. <http://www.c-s-a.org.cn/1003-3254/6765.html>

## Fireworks Algorithm for Solving 0-1 Knapsack Problems

XU Xiao-Ping<sup>1</sup>, PANG Run-Juan<sup>1</sup>, WANG Feng<sup>2</sup>, QIAN Fu-Cai<sup>3,4</sup>

<sup>1</sup>(Faculty of Sciences, Xi'an University of Technology, Xi'an 710054, China)

<sup>2</sup>(School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710049, China)

<sup>3</sup>(Faculty of Automation and Information Engineering, Xi'an University of Technology, Xi'an 710048, China)

<sup>4</sup>(State Key Laboratory of Astronautic Dynamics, Xi'an Satellite Control Center, Xi'an 710043, China)

**Abstract:** To overcome the shortcomings of the existing method in solving the 0-1 knapsack problems, an improved fireworks algorithm is proposed. After a mathematical model of the 0-1 knapsack problem is given, an improved fireworks algorithm is proposed to solve it. The main idea of the improved algorithm is as follows. The initialization of the basic fireworks algorithm is solved by using Kent chaotic map to make the initial position more uniform. The Sigmoid function is also introduced to get the gradual explosion radius to make the solution accurate and the search speed reach a certain balance. The resolving results of the typical test function and the 0-1 knapsack problem show that the improved fireworks algorithm has higher precision and more stable performance.

**Key words:** 0-1 knapsack problem; optimization; fireworks algorithm; chaotic mapping; gradual explosion radius

背包问题又称子集合问题, 其最早由 Merkel 和 Hellman 于 1978 年提出, 属于经典的 NP-hard 难解问题, 无论在理论研究方面还是实际应用方面均有很大的价值, 已被应用于下料问题, 预测控制, 资本投资和

① 基金项目: 国家自然科学基金 (61773016); 陕西省自然科学基金基础研究计划 (2014JM8325); 陕西省教育厅专项科研计划 (14JK1538); 西安理工大学科技创新计划 (2016CX013)

Foundation item: National Natural Science Foundation of China (61773016); Fundamental Research Program of Natural Science of Shaaxi Province (2014JM8325); Scientific Research Project of Education Bureau of Shaaxi Province (14JK1538); Science and Technology Innovation Program of Xi'an University of Technology (2016CX013)

收稿时间: 2018-08-08; 修改时间: 2018-08-30; 采用时间: 2018-09-05; csa 在线出版时间: 2019-01-28

贷款组合优化决策问题等许多领域<sup>[1]</sup>. 背包问题描述简单, 但实际求解过程复杂, 其求解方法可分为两大类: 传统求解和群智能算法. 早期传统方法如穷举法、分支界定法、回溯法和动态规划法等枚举了所有可能解<sup>[2-5]</sup>, 因此其解的数量相当大, 可能达到指数爆炸, 而群智能算法适于求解大规模问题, 为传统方法无法解决的优化问题提供了技术保障, 如差分进化算法<sup>[6]</sup>, 粒子群算法<sup>[7]</sup>和遗传算法<sup>[8]</sup>等. 但随着问题规模的增大, 逃逸局部极小值, 提高求解精度成为人们普遍关注的问题.

烟花算法是2010年由Tan和Zhu提出的一种新型群体智能算法<sup>[9]</sup>, 其属于随机搜索算法, 具有较强的并行性和鲁棒性, 近年来逐渐受到研究者的广泛关注. 目前, 烟花算法及其改进算法已被应用于多个领域, 例如, 文献<sup>[10]</sup>利用烟花算法在图像识别中进行了优化的距离度量; 文献<sup>[11]</sup>使用烟花算法进行了配电网优化方案设计; 文献<sup>[12]</sup>将烟花算法应用于非负矩阵分解问题; 文献<sup>[13]</sup>提出多目标烟花算法, 并用其进行了施肥比例的设计; 文献<sup>[14]</sup>提出动态变化半径的烟花算法, 并将其应用于负荷在线优化分配中; 文献<sup>[15]</sup>提出自适应烟花算法, 并应用于重型装备装载问题. 但是, 随着问题复杂性的增加, 该算法还会出现陷入局部极值的弊端. 因此, 对该算法进一步研究是很必要的, 并对解决实际问题具有重要的现实意义.

为了提高烟花算法的寻优性能, 本文将Kent混沌映射和Sigmoid函数引入到基本烟花算法中, 从而给出了一种改进烟花算法, 并将其应用到求解0-1背包问题. 通过实验仿真验证了所给算法是有效的.

## 1 烟花算法

### 1.1 基本烟花算法

烟花算法是由烟花的爆炸过程得到启示而提出的一种新型群体智能算法<sup>[9]</sup>. 该算法在每一代都选择出性能较好的烟花作为父代, 来产生子代火花用以搜索最优解. 每一个烟花 $x_i$ , 其产生火花的数量 $S_i$ 以及爆炸半径 $A_i$ 分别定义如下:

$$S_i = Mii \times \frac{f_{\max} - f(x_i) + \varepsilon}{\sum_{j=1}^M (f_{\max} - f(x_j)) + \varepsilon} \quad (1)$$

$$A_i = Aii \times \frac{f(x_i) - f_{\min} + \varepsilon}{\sum_{j=1}^M (f(x_j) - f_{\min}) + \varepsilon} \quad (2)$$

其中,  $Mii$ 和 $Aii$ 是常数, 分别用来调整产生的爆炸火花和爆炸半径大小;  $M$ 为种群规模;  $f_{\max}$ 和 $f_{\min}$ 分别表示目标函数的最大值和最小值;  $\varepsilon$ 是用以避免结果为零的极小量.

为了使产生的火花粒子不过多或过少, 需对 $S_i$ 的大小范围进行如下限制:

$$S_i = \begin{cases} S_{\min}, & S_i < S_{\min} \\ S_{\max}, & S_i > S_{\max} \\ S_i, & \text{otherwise} \end{cases} \quad (3)$$

假设搜索空间的维数为 $D$ , 随机选择烟花 $x_i$ 的 $z(z < D)$ 个维度, 对每一个维度的 $x_i^k$ 进行位置偏移, 构成 $x_j^k (1 \leq j \leq S_i, 1 \leq k \leq z)$ . 生成火花 $x_j^k$ 的方法有两种, 大部分火花是通过对 $x_i^k$ 加上一个位移 $h_k = A_i \cdot \text{rand}(-1, 1)$ 而得到的, 即:

$$x_j^k = x_i^k + h_k \quad (4)$$

为了提高种群的多样性, 少数的烟花将需进行高斯变异, 即将一个高斯偏移乘到 $x_i^k$ 上, 即:

$$x_j^k = x_i^k \cdot \text{Gaussian}(1, 1) \quad (5)$$

在上述过程中, 产生的火花可能会超出可行域的边界范围, 从而当火花 $x_i$ 在 $k$ 维上超出边界时, 可由(6)式映射到其它位置.

$$x_j^k = x_{\min}^k + |x_j^k| \bmod (x_{\max}^k - x_{\min}^k) \quad (6)$$

在每一次迭代中, 所有的个体(包括烟花, 爆炸火花和高斯变异火花)中的最优个体, 总是被选为烟花进入下一代, 并根据轮盘赌的方法选择其他 $M-1$ 个个体作为下一代迭代计算的烟花.

### 1.2 改进的烟花算法

在基本烟花算法中, 初始化种群由随机函数产生, 而利用随机初始化方法给出的一组解可能会出现与最优值较为接近, 也可能相差甚远, 即给出的解往往无法覆盖整个解空间, 解得不到充分挖掘, 从而致使算法全局搜索能力较弱. 另外, 基本烟花算法的爆炸半径采用式(2)进行计算, 使得对于最优烟花带入式(2)后得到的火花半径趋于0, 即原样复制了最优烟花, 没有进行任何搜索, 从而降低了搜索速度, 降低了在其周围找到最优解的可能性. 针对基本烟花算法自身所存在的这

些缺陷,本文在算法初始化过程中,采用具有遍历性和随机性的 Kent 混沌映射来替代随机初始化过程,同时引入 Sigmoid 函数对烟花半径进行改进,使得爆炸半径能及时地对搜索方向进行引导和限制.改进方法具体如下:

### 1.2.1 解初始化的改进

种群初始解的质量在一定程度上影响最终解的质量,初始解在解空间分布越均匀,覆盖面就越广泛,从而在最优解邻域搜索的可能性就越大.而混沌的无规则性可使初始解在解空间分布更加均匀,所以,这里采用混沌初始化代替原来的随机初始化过程,使得初始化过程可以搜索到整个解空间,从而可提高最优解被搜索到的概率.目前,在改进群智能算法时,一般引入的是 Logistic 映射策略进行混沌初始化,但是,该映射的混沌特性并不十分优良,其分岔图如图 1 所示.而 Kent 映射克服了 Logistic 映射的缺点<sup>[16]</sup>,图 2 给出了 Kent 映射的分岔图.

从图 1 可看出,当 $3.42 < \mu \leq 3.52$ 时,发生了第二周期分岔,从周期二轨道变成周期四轨道,当 $3.52 < \mu \leq 3.58$ 时,发生了第三周期分岔,从周期四轨道变成周期八轨道,而当 $3.58 < \mu \leq 4$ 时,轨道周期达到无穷长,系统才开始工作于混沌状态.而从图 2 的 Kent 映射分岔图中可以看出, Kent 映射始终处于混沌状态.从而可看出 Kent 映射的混沌特性更明显.

另外, Lyapunov 指数通常是衡量系统动力学特性的一个重要定量指标,它表征了系统在相空间中相邻轨道间收敛或发散的平均指数率,指数越大,说明混沌特性越明显,混沌程度越高,其分布越均匀<sup>[16]</sup>. Kent 映射的 Lyapunov 指数是 0.696,而 Logistic 映射的 Lyapunov 指数是 0.691,因此,这里采用 Kent 混沌映射对烟花粒子进行初始化,具体由如下函数来产生:

$$h_{j+1} = \begin{cases} \frac{h_j}{b}, & 0 < h_j \leq b \\ \frac{1-h_j}{1-b}, & b < h_j \leq 1 \end{cases} \quad (7)$$

其中,  $h_j$  为混沌变量,  $h_j \in [0, 1]$ ,  $j = 1, 2, \dots, n$ ;  $b$  为参数,当  $b = 0.4$  时,  $h_j$  为  $(0, 1)$  上的均匀分布,且  $x_{ij}$  可由下式产生.

$$x_{ij} = x_{\min} + h_j \cdot (x_{\max} - x_{\min}) \quad (8)$$

其中,  $x_{ij}$  为第  $i$  个烟花在第  $j$  维的具体位置;  $[x_{\min}, x_{\max}]$  为  $x_{ij}$  的取值范围. 因为  $h_j$  为  $(0, 1)$  上的均匀分布, 所以由 (8) 式可以使  $x_{ij}$  广泛分布于区间  $[x_{\min}, x_{\max}]$  中, 有利于

烟花进行下一步搜索,这在很大程度上避免了算法陷入局部最优值,从而节省了搜索时间,能很快找到最优解,因此有利于提高整个算法的求解精度,加快了算法收敛速度.

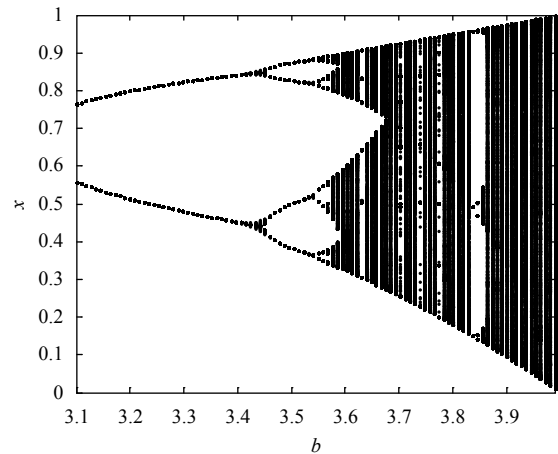


图 1 Logistic 映射分岔图

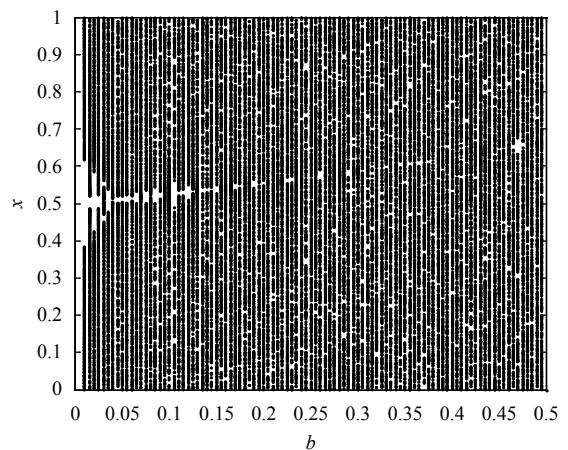


图 2 Kent 映射分岔图

### 1.2.2 爆炸半径的改进

为了使得算法爆炸半径能及时地对搜索方向进行引导和限制,那么就需要算法在迭代初期,爆炸半径取值较大来加强全局探索能力;在迭代后期,爆炸半径取值较小来加强局部搜索能力,这样就可以使得算法求解精度与搜索速度达到一种平衡来改进其寻优性能.

在人工神经网络中,神经元激活函数经常采用 Sigmoid 函数和 Tanh 函数来过渡线性与非线性,其往往都可以作为激励函数<sup>[17]</sup>,它们的曲线图如图 3 所示.但是,从图中可以看出, Sigmoid 函数在线性与非线性

行为之间表现出更好的平衡性, 过渡更加平滑. 并且由参考文献[17]可知, Sigmoid 函数连续、可导、有界、光滑和严格单调, 可将变量映射到[0, 1]范围内, 并且在区间内逐渐变小. 因此, 在这里引入 Sigmoid 函数可得到平滑渐变的爆炸半径, 从而在迭代初期, 爆炸半径取值较大, 算法的全局探索能力强, 同时有利于烟花弹较快集中在全局最优点附近; 在迭代后期, 爆炸半径取值较小, 使得算法在最优点附近能进行充分的局部搜索, 提高求解的精度, 这样就可以使得求解精度与搜索速度达到一种平衡, 并且在迭代前后平滑过渡, 而不显得突兀. 这里构造的递减因子具体如下.

$$a_{k+1} = \frac{1}{1 + e^{2 \ln 100 \cdot \frac{k}{M} - \ln 1000}} \cdot \frac{M - k + 1}{M} \cdot a \quad (9)$$

其中,  $k$ 为当前的迭代次数,  $M$ 为最大的迭代次数,  $a$ 为调节系数.

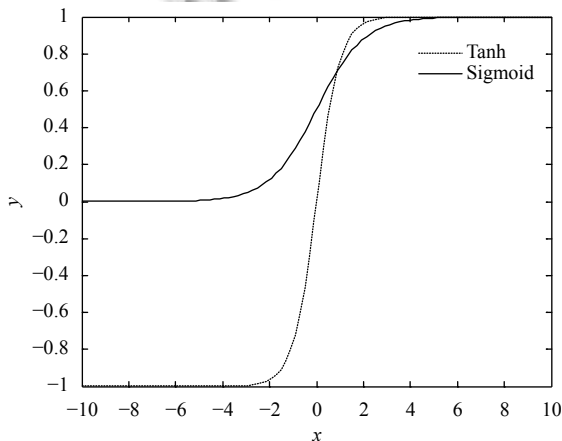


图3 两个激励函数曲线图

## 2 利用改进的烟花算法求解 0-1 背包问题

### 2.1 0-1 背包问题

0-1 背包问题可描述为<sup>[1]</sup>: 给定一个背包和  $m$  种不同的物品, 其中背包的容积、物品的体积和价值一定, 一种物品只能有装入或者不装入背包中两种状态, 要求解一种物品装入方案使装入背包中的物品总价值最大. 假设背包容积为  $C$ , 有  $m$  个物品待选择, 其中第  $i$  个物品的体积为  $w_i$ , 价值为  $p_i$ ,  $x_i = 1$  为物品被装入背包中,  $x_i = 0$  为物品不装入背包中, 其中  $i = 1, 2, \dots, m$ . 这样问题就可表述为:

$$\begin{aligned} \max f(x) &= \sum_{i=1}^m p_i x_i \\ \text{s.t.} &\begin{cases} \sum_{i=1}^m w_i x_i \leq C \\ x_i = \begin{cases} 1, \text{物品 } i \text{ 装入背包} \\ 0, \text{物品 } i \text{ 不装入背包} \end{cases} \end{cases} \end{aligned} \quad (10)$$

现需考虑在不超过背包容积  $C$  的前提下, 怎样使装入背包中的物品总价值最大, 那么对该问题的求解就可以采用下面的烟花算法.

### 2.2 建立算法与问题的对应关系

要利用改进的烟花算法求解 0-1 背包问题, 首先要明确算法与 0-1 背包问题间的相互对应关系, 其具体如表 1 所示.

表 1 对应关系

改进的烟花算法	0-1 背包问题
烟花位置	式 (10) 的一组可行解
种群规模	备选方案的数量
维数	物品的数量
适应度	式 (10) 的目标函数值

### 2.3 求解步骤

Step 1. 解的表示和初始化: 设种群规模为  $M$ , 对初始烟花利用 (7) 和 (8) 式进行混沌搜索, 对于第  $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$  个烟花, 其当前的位置向量为  $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ , 其中  $i = 1, 2, \dots, M, j = 1, 2, \dots, D, D$  为所求问题的物品数量.

Step 2. 迭代过程: 初始  $iter = 1$ , 计算烟花粒子的适应度, 即背包内物品的价值, 通过函数限制背包内物体的总体积.

Step 3. 编码: 采用二进制编码, 当背包内物体的体积超过限制时, 将其适应度设置为 0, 当烟花粒子的适应度大于其最佳适应度时, 用其替代原来的粒子适应度, 并记录下来, 同样当存在粒子的最佳适应度大于种群最佳适应度时, 用其替代原来种群的最佳适应度.

Step 4. 爆炸火花的产生: 利用式 (9) 计算烟花半径, 利用式 (4) 产生爆炸火花数, 由式 (6) 进行越界检查, 对超出边界的火花重新映射到边界内.

Step 5. 高斯变异火花的产生: 由式 (5) 产生高斯变异火花, 利用式 (6) 进行越界检查, 同样对超出边界的火花重新映射到边界内.

Step 6. 离散化函数值: 由于烟花位置只有 0, 1 两种状态, 以 0.5 为分界点对函数值进行离散化, 其中小

于 0.5 的粒子取为 0, 大于 0.5 的粒子取为 1.

Step 7. 子代选择: 将种群中烟花个体的最优个体选为烟花进入下一代, 并根据轮盘赌的方法选择其它  $M-1$  个个体作为下一代的父代烟花, 返回 Step 2,  $iter = iter + 1$ , 直至背包内的物品体积达到背包限制, 或所选物品价值达到某一可接受值, 算法结束, 输出最优函数值.

### 3 仿真实验

为了说明所给的 IFWA 的可行性, 以下对典型测试函数和 0-1 背包问题进行求解, 并与基本烟花算法 (BFWA)<sup>[9]</sup>, 文献[18]给出的改进爆炸半径的烟花算法 (RFWA), 文献[19]给出的加入惯性权重的烟花算法 (WFWA), 差分进化算法 (DE)<sup>[6]</sup>, 粒子群算法 (PSO)<sup>[7]</sup>和遗传算法 (GA)<sup>[8]</sup>的求解结果进行比较. 在仿真实验中, IFWA 的参数设置是依据文献[9]及多次试验所获得的经验值来确定的, 其中种群大小  $M = 20$ , 总的迭代次数  $N = 1000$ , 半径系数  $A_{ii} = 1500$ , 火花系数  $M_{ii} = 300$ , 高斯变异火花数  $Ma = M/3$ , 搜索空间维数  $D = 20$ , 爆炸火花最大值  $S_{imax} = 27$ , 最小值  $S_{imin} = 1$ , 极小量  $\epsilon = 10^{-250}$ , 调节系数  $a = 10$ ; 对比算法参数值设置见文献[6-9, 18,19].

#### 3.1 典型测试函数

(1) Sphere 函数:  $f(x) = \sum_{i=1}^N x_i^2$

(2) Rosenbrock 函数:

$$f(x) = \sum_{i=1}^N [100 \cdot (x_{i+1} - x_i^2)^2 + (x_i - 1)^2] \quad (11)$$

这两个函数的理论最小值均为 0. 现对两函数均在 20 维的情况下, 利用 IFWA, RFWA<sup>[18]</sup>, WFWA<sup>[19]</sup>, BFWA<sup>[9]</sup>, DE<sup>[6]</sup>, PSO<sup>[7]</sup>和 GA<sup>[8]</sup>分别进行一次求解, 得到寻优过程曲线如图 4 和图 5 所示.

在仿真过程中发现, 正如前面理论分析所言, 本文所提 IFWA 的优化性能明显改善. 这是由于 IFWA 用 Kent 混沌映射来替代随机初始化过程后, 使得初始解分布更加均匀. 同时, 引入 Sigmoid 函数得到平滑渐变的爆炸半径后, 使得算法的求解精度与搜索速度达到一种平衡. 从图 4 和图 5 均可以看出, IFWA 的寻优性能远远强于 RFWA, WFWA, BFWA, DE, PSO 和 GA. 并且从图 4 可以看出, 对 Sphere 函数而言, 虽然 RFWA 和 WFWA 在求解精度上有了一定的提高, 可是, 两个算法在迭代 920 次时还未稳定, 而 IFWA 不仅求解精度大幅度改善, 而且 IFWA 在不到 50 次迭代时

就已经收敛, 搜索速度较快. 从图 5 可以看出, 对 Rosenbrock 函数而言, RFWA 和 WFWA 收敛速度较慢, 在迭代次数接近 980 次时又出现不稳定现象, 而 IFWA 在迭代不到 100 次时已收敛, 收敛速度较快, 求解精度也大幅度提高.

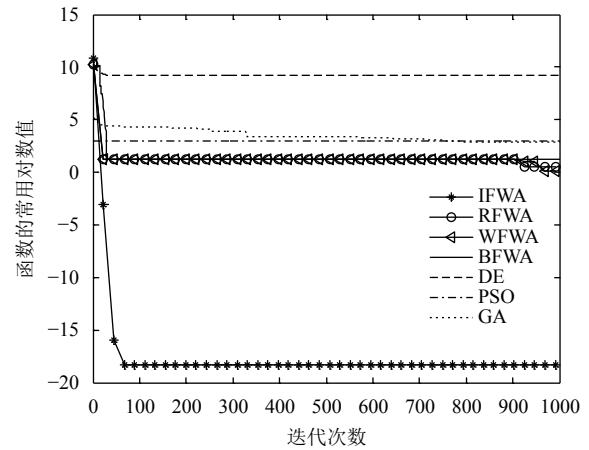


图 4 对 Sphere 函数的优化曲线

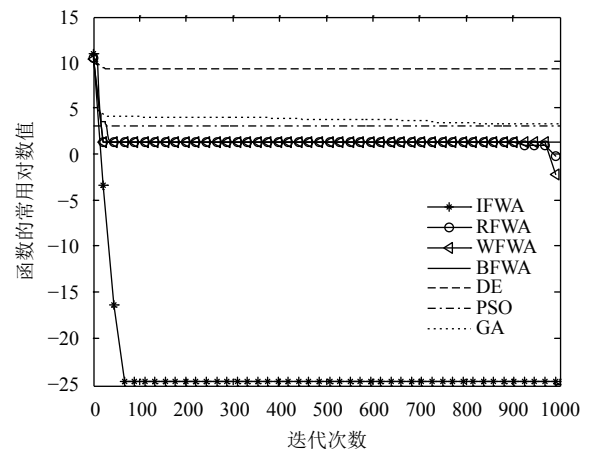


图 5 对 Rosenbrock 函数的优化曲线

进一步利用 IFWA, RFWA<sup>[18]</sup>, WFWA<sup>[19]</sup>, BFWA<sup>[9]</sup>, DE<sup>[6]</sup>, PSO<sup>[7]</sup>和 GA<sup>[8]</sup>分别进行 50 次求解, 将得到的最优解, 平均解和方差分别罗列在表 2 中. 从表 2 可看出, 相对于 DE, PSO, GA, BFWA, RFWA 和 WFWA, 所提 IFWA 的混沌初始化改进有效地搜索到了最优解, 且对于爆炸半径的改进也避免了最优解的浪费, 故在 50 次求解中利用 IFWA 得到的最优解和平均解都较小, 更接近于理论最优值, 并且方差较小, 即具有更强的稳定性. 对于文献[18]所提改进半径烟花算法和文献[19]引入惯性权重的烟花算法的最优解虽然优于基本烟花算法, 但与本文所提 IFWA 仍有较大差距.

### 3.2 0-1 背包问题

这里选用文献[20]中所给的 0-1 背包问题, 它们的规模由小到大, 具体数据见算例 1 和算例 2.

算例 1: 物品数量  $m=50$ , 背包容量  $C=1000$ , 物品价值  $p=[220, 208, 195, 192, 180, 180, 165, 162, 160, 158, 155, 130, 125, 122, 120, 118, 115, 110, 105, 101, 100, 100, 98, 96, 95, 90, 88, 82, 80, 77, 75, 73, 72, 70, 69, 66, 65, 63, 60, 58, 56, 50, 30, 20, 15, 10, 8, 5, 3, 1]$ , 物品体积  $w=[80, 82, 85, 70, 72, 70, 66, 50, 55, 25, 50, 55, 40, 48, 50, 32, 22, 60, 30, 32, 40, 38, 35, 32, 25, 28, 30, 22, 25, 30, 45, 30, 60, 50, 20, 65, 20, 25, 30, 10, 20, 25, 15, 10, 10, 10, 4, 4, 2, 1]$ .

表 2 对 2 个测试函数的寻优结果

函数	算法	最优解	平均解	方差
Sphere	DE	7.5412e+8	9.4916e+8	3.4215e-4
	PSO	3.2341e+3	4.3428e+3	5.6147e-14
	GA	5.2631e+3	7.8973e+4	8.4090e+1
	RFWA	8.3620e-1	5.2438e+1	6.7559e-6
	WFWA	6.9244e-2	1.2574e-1	3.2874e-2
	BFWA	2.9813e+2	3.4115e+2	7.1552e-3
	IFWA	2.2717e-18	6.4129e-15	9.2470e-16
	DE	2.5814e+9	5.8514e+9	3.5256e-7
	PSO	3.9714e+3	6.1225e+3	2.3546e-7
	GA	9.3216e+2	5.1414e+3	7.1432e-3
Rosenbrock	RFWA	3.9212e-2	2.9477e-1	7.9244e-3
	WFWA	8.2741e-3	3.3471e-1	5.2966e-6
	BFWA	2.9884e+1	3.9899e+2	5.5303e-5
	IFWA	7.2526e-25	7.8492e-24	1.3718e-12

算例 2: 物品数量  $m=100$ , 背包容量  $C=2010$ , 物品价值  $p=[68, 101, 125, 159, 65, 146, 28, 92, 143, 37, 5, 154, 183, 117, 96, 127, 139, 113, 100, 95, 12, 134, 65, 112, 69, 45, 158, 60, 142, 179, 36, 43, 107, 143, 49, 6, 130, 151, 102, 149, 24, 155, 41, 177, 109, 40, 124, 139, 83, 142, 116, 59, 131, 107, 187, 146, 73, 30, 174, 13, 91, 37, 168, 175, 53, 151, 125, 31, 192, 138, 88, 184, 110, 159, 189, 147, 31, 169, 192, 56, 160, 138, 84, 42, 151, 37, 21, 22, 200, 85, 135, 200, 139, 189, 68, 94, 84, 22, 18, 115]$ , 物品体积  $w=[42, 35, 70, 79, 63, 6, 82, 62, 96, 28, 92, 3, 93, 22, 19, 48, 72, 70, 68, 36, 4, 23, 74, 42, 54, 48, 63, 38, 24, 30, 17, 91, 89, 41, 65, 47, 91, 71, 7, 94, 30, 85, 57, 67, 32, 45, 27, 38, 19, 30, 34, 40, 5, 78, 74, 22, 25, 71, 78, 98, 87, 62, 56, 56, 32, 51, 42, 67, 8, 8, 58, 54, 46, 10, 22, 23, 7, 14, 1, 63, 11, 25, 49, 96, 3, 92, 75, 97, 49, 69, 82, 54, 19, 1, 28, 29, 49, 8, 11, 14]$ .

利用 IFWA, RFWA<sup>[18]</sup>, WFWA<sup>[19]</sup>, BFWA<sup>[9]</sup>, DE<sup>[6]</sup>, PSO<sup>[7]</sup>和 GA<sup>[8]</sup>算法分别对上述两个算例进行一次求

解, 得到相应寻优曲线如图 6 和图 7 所示.

在仿真过程中发现, 所提 IFWA 在对初始解进行了混沌初始化和爆炸半径进行了渐变改进后, 算法的局部搜索能力与全局搜索能力达到了平衡, 能迅速跳出局部最优, 加快了算法的求解精度与收敛速度. 从图 6 和图 7 均可以看出, IFWA 的寻优能力均明显优于 RFWA, WFWA, BFWA, DE, PSO 和 GA. 并且从图 6 和图 7 可以看出, 当算例 1 的物品数量为 50 和例 2 的物品数量为 100 时, IFWA 均在迭代 20 次左右时就已经收敛, 收敛速度较快, 求解精度得到明显提高.

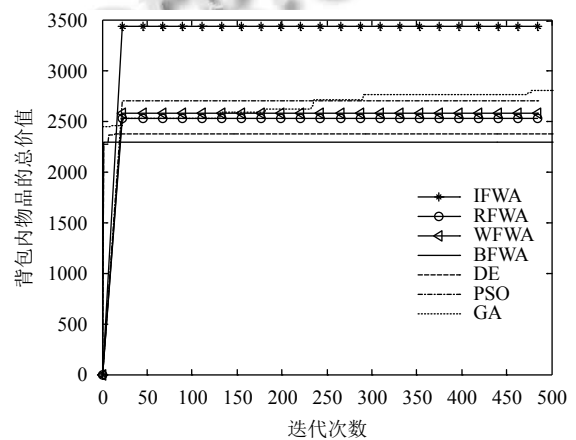


图 6 算例 1 寻优曲线图

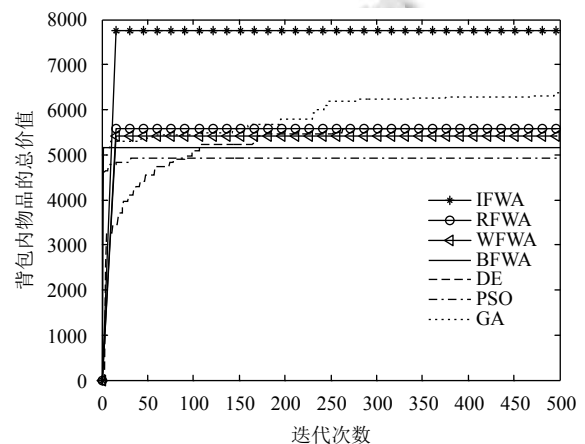


图 7 算例 2 寻优曲线图

进一步利用 IFWA, RFWA<sup>[18]</sup>, WFWA<sup>[19]</sup>, BFWA<sup>[9]</sup>, DE<sup>[6]</sup>, PSO<sup>[7]</sup>和 GA<sup>[8]</sup>分别对两个算例进行 50 次求解, 得到的最优解, 最差解, 平均解和方差如表 3 所示. 从表 3 可看出, 所给 IFWA 的寻优结果相对于其它所提的算法明显更优, 且方差较小, 说明算法具有较好的稳定性. 尤其随着物品数量的增大, 问题复杂度的加大,

相对于其它所提到的对比优化算法, 所给 IFWA 仍然具有较高的求解精度和稳定性, 说明改进算法是有效的。

表3 0-1 背包问题的寻优结果

物品量	算法	最优解	最差解	平均解	方差
m=50	DE	2520	2054	2132	8.4391e+4
	PSO	2829	2008	2182	9.2430e+4
	GA	2861	2544	2720	54 911
	BFWA	2420	2105	2366	86 223
	WFWA	2830	2526	2642	45 271
m=100	RFWA	2755	2440	2591	69 531
	IFWA	3520	3400	3495	11 902
	DE	5659	5028	5241	3.8724e+5
	PSO	5282	4827	5074	2.3415e+5
	GA	6384	5971	6072	3.2759e+5
m=100	BFWA	5527	5049	5209	89 271
	WFWA	5629	5032	5360	92 510
	RFWA	5872	5234	5569	76 941
	IFWA	8254	7753	8161	21 602

综上, 通过以上对测试函数和 0-1 背包问题的仿真结果可看出, 相对于 BFWA<sup>[9]</sup>, RFWA<sup>[18]</sup>, WFWA<sup>[19]</sup>, DE<sup>[6]</sup>, PSO<sup>[7]</sup>和 GA<sup>[8]</sup>, 所给的 IFWA 由于采用了混沌初始化和渐变的爆炸半径, 因此在搜索能力上明显更优, 能够更好地搜索到全局最优解, 且稳定性较好。从而说明所给 IFWA 算法是有效的, 达到了预期的目的, 为求解 0-1 背包问题提供了新的途径。

## 4 结论

为了克服基本烟花算法的缺点, 利用 Kent 混沌映射对算法的解初始化进行了改进, 同时引入 Sigmoid 函数得到渐变的烟花爆炸半径后, 使得得到的改进烟花算法的求解精度与搜索速度达到某种平衡, 提高了算法的寻优性能。通过仿真实验验证了所给改进烟花算法是有效的。

## 参考文献

- Merkle R, Hellman M. Hiding information and signatures in trapdoor knapsacks. *IEEE Transactions on Information Theory*, 1978, 24(5): 525-530. [doi: 10.1109/TIT.1978.1055927]
- Caprara A, Pisinger D, Toth P. Exact solution of the quadratic knapsack problem. *Inform Journal on Computing*, 1999, 11(1): 125-139.
- 李鸣山, 郑海虹. 0-1 背包问题的多重分枝-限界算法. *武汉测绘科技大学学报*, 1995, 20(1): 84-85.
- 王梦竹. 求解 0-1 背包问题算法研究. *软件导刊*, 2013, 12(8): 59-61.

- 王乐, 王世卿, 张静乐. 基于 Matlab 的 0-1 背包问题的动态规划方法求解. *计算机技术与发展*, 2006, 16(4): 88-89, 92. [doi: 10.3969/j.issn.1673-629X.2006.04.031]
- 荆源. 背包问题的差分进化算法. *才智*, 2011, 12(7): 92-93.
- 赵传信, 季一木. 粒子群优化算法在 0/1 背包问题的应用. *微机发展*, 2005, 15(10): 23-25. [doi: 10.3969/j.issn.1673-629X.2005.10.009]
- 乐天. 遗传算法求解 0/1 背包问题的综述. *浙江海洋学院学报(自然科学版)*, 2013, 32(1): 71-74. [doi: 10.3969/j.issn.1008-830X.2013.01.014]
- Tan Y, Zhu YC. Fireworks algorithms for optimization. *Proceeding of International Conference on Swarm Intelligence (ICSI2010)*. Beijing, China. 2010. 355-364.
- Zhang SQ, Tan Y. A unified distance measure scheme for orientation coding in identification. *2013 International Conference on Information Science and Technology*. Yangzhou, China. 2013. 979-985.
- Imran AM, Kowsalya M. A new power system reconfiguration scheme for power loss minimization and voltage profile enhancement using fireworks algorithm. *International Journal of Electrical Power & Energy Systems*, 2014, 62: 312-322.
- Janecek A, Tan Y. Swarm intelligence for non-negative matrix factorization. *International Journal of Swarm Intelligence Research*, 2011, 2(4): 12-34. [doi: 10.4018/IJSIR]
- Zheng YJ, Song Q, Chen SY. Multiobjective fireworks optimization for variable-rate fertilization in oil crop production. *Applied Soft Computing*, 2013, 13(11): 4253-4263. [doi: 10.1016/j.asoc.2013.07.004]
- 曾德良, 邓志光, 陈彦桥. 改进烟花算法在火电厂负荷优化分配中的应用. *应用科学学报*, 2018, 36(3): 542-552. [doi: 10.3969/j.issn.0255-8297.2018.03.014]
- 陈璇, 樊永生, 余红英, 等. 自适应烟花算法在重型装备装载中的应用. *科学技术与工程*, 2016, 16(25): 296-300. [doi: 10.3969/j.issn.1671-1815.2016.25.051]
- 孙宪坤, 陈涛, 韩华, 等. 基于 Kent 混沌测量矩阵的压缩感知图像重构算法. *计算机应用与软件*, 2017, 34(4): 213-220. [doi: 10.3969/j.issn.1000-386x.2017.04.036]
- 吴湘锋, 李志军, 向林波. 可编程双极性 Sigmoid 函数及其导数发生器. *电子测量与仪器学报*, 2015, 29(8): 1137-1143.
- 杜振鑫. 烟花算法中爆炸半径的改进研究. *计算机时代*, 2013, (1): 28-29. [doi: 10.3969/j.issn.1006-8228.2013.01.011]
- 尚亚菲, 刘雪英, 贾敏南. 引入惯性权重的烟花算法. *内蒙古工业大学学报*, 2016, 35(3): 168-177. [doi: 10.3969/j.issn.1001-5167.2016.03.002]
- 王秋芬, 梁道雷. 一种求解 0-1 背包问题的启发式遗传算法. *计算机应用与软件*, 2013, 30(2): 33-37, 57. [doi: 10.3969/j.issn.1000-386x.2013.02.009]