

# 一种多策略协同的多目标萤火虫算法

谢承旺<sup>1,2</sup>, 张飞龙<sup>2</sup>, 陆建波<sup>1</sup>, 肖 驰<sup>2</sup>, 龙广林<sup>1</sup>

(1. 南宁师范大学计算机与信息工程学院, 广西南宁 530299; 2. 华东交通大学软件学院, 江西南昌 330013)

**摘 要:** 现实中的多目标优化问题不断增多且日益复杂, 需要不断发展新型启发式算法应对挑战. 提出一种多策略协同的多目标萤火虫算法 MOFA-MCS. 该算法采用均匀化与随机化相结合的方法产生初始种群; 利用档案集中的精英解个体指导萤火虫移动; 并在移动的过程施加 Lévy flights 随机扰动; 最后, 利用  $\varepsilon$ -三点最短路径策略维护档案解群的多样性. MOFA-MCS 算法与其他 6 种经典的多目标进化算法一同在 12 个基准的多目标测试问题上进行实验, 结果表明所提算法在收敛性、多样性方面总体上具有显著的性能优势.

**关键词:** 多目标优化问题; 萤火虫算法; 多目标萤火虫算法; 多策略协同

**中图分类号:** TP18 **文献标识码:** A **文章编号:** 0372-2112 (2019)11-2359-09

**电子学报 URL:** <http://www.ejournal.org.cn> **DOI:** 10.3969/j.issn.0372-2112.2019.11.018

## Multi-Objective Firefly Algorithm Based on Multiply Cooperative Strategies

XIE Cheng-wang<sup>1,2</sup>, ZHANG Fei-long<sup>2</sup>, LU Jian-bo<sup>1</sup>, XIAO Chi<sup>2</sup>, LONG Guang-lin<sup>1</sup>

(1. School of Computer and Information Engineering, Nanning Normal University, Nanning, Guangxi 530299, China;

2. School of Software, East China Jiaotong University, Nanchang, Jiangxi 330013, China)

**Abstract:** More and more complex multi-objective optimization problems have emerged in the real world, and the novel heuristic algorithms need to be developed to meet the challenge. A multi-objective firefly algorithm based on multiply cooperative strategies (MOFA-MCS) is proposed in the paper. MOFA-MCS uses the method of homogenization and randomization to generate the initial population, adopts the elite solutions in the external archive to lead the firefly to move, exerts Lévy flights to add random disturbance in the moving process, and finally, the  $\varepsilon$ -three-point shortest path strategy is also applied to maintain the diversity of the archive solutions. MOFA-MCS is compared with other six representative multi-objective evolutionary algorithms on 12 benchmark multi-objective test problems, and the experimental results show that MOFA-MCS has significant performance advantages in terms of convergence and diversity.

**Key words:** multi-objective optimization problem; firefly algorithm; multi-objective firefly algorithm; multiply cooperative strategies

## 1 引言

现实中存在大量需要同时优化多个目标的问题, 它们被称为多目标优化问题 (Multi-Objective Optimization Problem, MOP). MOP 中各目标之间彼此冲突, 使得一个解对于某个目标可能是好的, 而对于其他目标则可能较差. 因此, MOP 一般存在一个折衷的解集, 即 Pareto 解集. 进化算法 (Evolutionary Algorithm, EA) 是一种模拟生物群体进化过程的随机优化方法, 其基于群体

搜索的特性以及对待求解问题的数学性质不作严格假设等优势, 使得 EA 算法适于计算一些模型高度复杂的 MOP 问题.

近三十多年来, 多目标进化算法 (Multi-Objective Evolutionary Algorithm, MOEA) 的研究获得了长足的发展, 涌现出了许多性能优秀的 MOEA 算法, 如非支配排序遗传算法<sup>[1]</sup> (NSGA) 及其改进的版本 NSGA-II<sup>[2]</sup>、强度 Pareto 进化算法<sup>[3]</sup> (SPEA) 和改进的 SPEA2<sup>[4]</sup> 等. 之后, Zhang 等<sup>[5]</sup> 提出基于分解的多目标进化算法

收稿日期: 2018-10-17; 修回日期: 2019-03-28; 责任编辑: 孙瑶

基金项目: 国家自然科学基金 (No. 61763010); 广西八桂学者项目 (No. 桂科政字[2016]127号); 广西创新驱动重大专项 (No. AA18118047); 广西科技基地和人才专项 (No. 桂科 AD18126015); 广西自然科学基金 (No. 2018GXNSFAA138056); 广西研究生教育创新计划资助 (No. YCSW2019182)

(MOEA/D),以及 Wang 等<sup>[6]</sup>在 MOEA/D 基础上施加自适应调整约束子问题机制的 MOEA/D-ACD 算法等.这一类算法将 MOP 问题分解成一系列子问题,然后利用子问题邻域信息协同地求解,但是这类算法在获得均匀分布的解集方面尚存在挑战.近年来,利用不同启发式算法和进化方法之间的搜索机理,进行优势互补,亦产生出了若干性能较好的混合多目标进化算法,譬如, Nebro 等<sup>[7]</sup>提出的一种基于档案的混合分散搜索算法 AbYSS 等.这一类算法克服了单个方法所固有的局限,增强了算法的搜索效率和效果.最近几年,MOEA 领域的研究呈现出新的特点,一些新的进化机制和启发式方法被陆续引入到多目标优化领域,如基于烟花爆炸优化的多目标烟花爆炸算法<sup>[8,9]</sup>等.新型进化范例的引入拓宽了解决复杂 MOP 问题的思路,并已成为多目标优化领域的研究热点之一.

2008 年, Yang<sup>[10]</sup>通过对萤火虫群体行为的模拟和简化提出了萤火虫算法(Firefly Algorithm, FA), FA 算法通过萤火虫之间的相互吸引达到寻优的目的,其具有概念简单、流程清晰、参数较少和容易实现等优点,一经提出就广泛地应用于求解单目标优化问题,并取得了较好的效果.2013 年, Yang<sup>[11]</sup>提出一种多目标萤火虫算法(Multi-Objective Firefly Algorithm, MOFA),该算法利用权重比策略确定当前的 Pareto 解. Tsai 等<sup>[12]</sup>提出了基于非支配排序的多目标萤火虫算法 MONSFA,该算法在萤火虫之间无支配关系时执行随机移动或向当前种群的最优个体移动,并利用 NSGA-II 的非支配排序和拥挤距离方法进行种群更新和档案维护. Xie 等<sup>[13]</sup>提出一种混合型多目标萤火虫算法 HMOFA,该算法利用混合水平正交实验设计方法产生初始解群,采用三点最短路径方法维护档案群体的多样性,并取得了较好的效果.但 HMOFA 算法需要为混合水平正交初始化方法指定两个合适的等水平正交表,而一些 MOP 问题并不满足这样的条件.此外, HMOFA 算法采用三点最短路径策略维持解群多样性方面亦存在改进的空间.不难发现,目前关于多目标萤火虫算法的研究不仅成果较少,而且这些算法大多只是从算法的局部,而未能从算法的整体上改善其性能,它们对算法的一些重要组件并未实施协同式的改进,因而已有多目标萤火虫算法在解决复杂 MOP 问题时尚有较大的性能提升空间.

为提高多目标萤火虫算法的整体性能以更好地求解复杂的 MOP 问题,本文提出一种多策略协同的多目标萤火虫算法(Multi-Objective Firefly Algorithm based on Multiply Cooperative Strategies, MOFA-MCS). MOFA-MCS 算法具有如下特点:(1)利用均匀化与随机化相结合的方法产生均匀分布的初始化种群;(2)利用外部档案中

的精英解指导萤火虫移动,在较好平衡算法全局勘探和局部开采的同时,加快算法收敛;(3)在萤火虫移动过程中施加 Lévy flights 随机扰动,有利于萤火虫逃逸局部极值区,增强全局搜索能力;(4)采用  $\varepsilon$ -三点最短路径方法维持档案群体的多样性,以获得分布性较好的解群.上述四种策略有机结合,形成了 MOFA-MCS 算法的主要特征,这些策略在算法的不同阶段实施,以协同地提高多目标萤火虫算法的整体性能.

## 2 基本知识

### 2.1 多目标优化问题

不失一般性,一个具有  $n$  个决策变量、 $m$  个目标函数的多目标优化问题,以最大化问题为例,可表述为式(1)的形式.

$$\begin{cases} \max \mathbf{y} = F(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x})) \\ \mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathbf{X} \subset \mathbf{R}^n \\ \mathbf{y} = (y_1, y_2, \dots, y_m) \in \mathbf{Y} \subset \mathbf{R}^m \end{cases} \quad (1)$$

式(1)中,  $\mathbf{x}$  称为决策向量,  $\mathbf{X}$  是  $n$  维决策空间;  $\mathbf{y}$  称为目标向量,  $\mathbf{Y}$  是  $m$  维的目标空间;目标函数  $F(\cdot)$  定义了映射函数和需要同时优化的  $m$  个目标.若  $\mathbf{X}$  为空间  $\mathbf{R}^n$  中相连且封闭的区域,且目标函数  $f_i (i = 1, 2, \dots, m)$  关于  $\mathbf{x}$  是连续的,则式(1)为连续型多目标优化问题.除非特别说明,本文仅讨论这一类优化问题.

对于决策空间内任意的两点  $x_1, x_2 \in \mathbf{X}$ , 当  $x_1$  的目标函数都不小于并且至少存在一个大于  $x_2$  的目标函数时,称  $x_1$  Pareto 支配  $x_2$ , 记为  $x_1 > x_2$ . 若  $\mathbf{x}^* \in \mathbf{X}$  不受种群中其它个体支配,则称  $\mathbf{x}^*$  为 Pareto 非支配解个体,种群中所有非支配解个体组成的集合称为 Pareto 解集(Pareto Set, PS),其对应目标函数构成的集合称为 Pareto 前沿(Pareto Front, PF).

### 2.2 基本萤火虫算法

萤火虫算法的核心思想是萤火虫被亮度较大的萤火虫吸引,并根据位置更新公式更新自身位置,所有萤火虫移动到新的位置后,更新其亮度,并依据吸引规则进行下一次移动.一般地,萤火虫的亮度由待优化的目标函数值决定,萤火虫之间的吸引力与它们的亮度成正比,而与它们的距离成反比<sup>[10]</sup>.

萤火虫  $i$  的绝对亮度  $I_i$  表征萤火虫所在位置处解的目标函数值.在最大化问题中,为降低问题的复杂度,设在  $\mathbf{x}_i(x_{i1}, x_{i2}, \dots, x_{in})$  处的萤火虫  $i$  的绝对亮度  $I_i$  与  $\mathbf{x}_i$  处的目标函数值相等,即:

$$I_i = f(\mathbf{x}_i) \quad (2)$$

假设萤火虫  $i$  的绝对亮度比萤火虫  $j$  的绝对亮度大,则萤火虫  $j$  被  $i$  吸引而向  $i$  移动.这种吸引力的大小由萤火虫  $i$  对萤火虫  $j$  的相对亮度决定,相对亮度越大,吸引力越

大. 考虑到萤火虫  $i$  的亮度随着距离的增加以及空气吸收而减弱, 可定义萤火虫  $i$  对萤火虫  $j$  的相对亮度为:

$$I_{ij}(r_{ij}) = I_i e^{-\gamma r_{ij}^2} \quad (3)$$

式(3)中的  $I_i$  为萤火虫  $i$  的绝对亮度,  $\gamma$  为光吸收系数,  $r_{ij}$  为萤火虫  $i$  到萤火虫  $j$  的距离.

假设萤火虫  $i$  对萤火虫  $j$  的吸引力与萤火虫  $i$  对萤火虫  $j$  的相对亮度成正比, 则由萤火虫  $i$  相对亮度的定义, 可得萤火虫  $i$  对萤火虫  $j$  的吸引力:

$$\beta_{ij}(r_{ij}) = \beta_0 e^{-\gamma r_{ij}^2} \quad (4)$$

式(4)中  $\beta_0$  为最大吸引力, 即在光源处 ( $r=0$ ) 萤火虫的吸引力. 这里将  $r_{ij}$  定义为萤火虫  $i$  到萤火虫  $j$  的笛卡尔距离, 即:

$$r_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\| = \sqrt{\sum_{k=1}^n (x_{ik} - x_{jk})^2} \quad (5)$$

由于被萤火虫  $i$  吸引, 萤火虫  $j$  向其移动而更新自己的位置, 萤火虫  $j$  位置更新公式如下:

$$\mathbf{x}_j(t+1) = \mathbf{x}_j(t) + \beta_{ij}(r_{ij})(\mathbf{x}_i(t) - \mathbf{x}_j(t)) + \alpha \boldsymbol{\varepsilon}_j \quad (6)$$

式(6)的  $t$  为算法迭代次数,  $\mathbf{x}_i(t)$  和  $\mathbf{x}_j(t)$  分别为第  $t$  代萤火虫  $i$  和  $j$  所处的位置;  $\beta_{ij}(r_{ij})$  为萤火虫  $i$  对萤火虫  $j$  的吸引力;  $\alpha$  为步长因子, 其取值为区间  $[0, 1]$  内均匀分布的随机数, 而  $\boldsymbol{\varepsilon}_j$  多为高斯分布或均匀分布所得到的随机数向量. 显然, 式(6)右边的第二项取决于吸引力; 第三项是带有特定系数的随机项, 它起扰动作用.

### 3 MOFA-MCS 算法

#### 3.1 种群初始化策略

多数多目标萤火虫算法随机产生初始群体, 由于随机误差的存在, 这种初始化方法不能保证群体较好地分布于决策空间. 鉴于此, 利用我们之前提出的一种均匀化与随机化相结合的方法产生初始种群<sup>[14]</sup>. 这种方法的特点在于: 既保持选取子区间以及从子区间产生基因值的随机性, 又能保证决策变量区间划分的均匀性. 初始化方法的流程如算法 1 所示.

算法 1 种群初始化方法

输入: 种群规模  $NP$ , 决策向量的维数  $n$ , 各决策变量  $\mathbf{x}_j (j \in [1:n])$  的区间  $[a_j, b_j]$ .

输出: 多样性的初始群体.

Step1 FOR  $j = 1$  TO  $n$

Step2  $\Delta_j = (b_j - a_j) / NP$  // 将决策变量  $\mathbf{x}_j$  的区间均匀划分成  $NP$  等份

Step3  $A_j = \{[a_j, a_j + \Delta_j], [a_j + \Delta_j, a_j + 2\Delta_j], \dots, [a_j + (NP-2)\Delta_j, a_j + (NP-1)\Delta_j], [a_j + (NP-1)\Delta_j, b_j]\}$

Step4 FOR  $i = 1$  TO  $NP$

Step5 从集合  $A_j$  中随机选择一个子区间, 并在该子区间内随机产生一个基因值赋给  $x_j^i$

Step6 更新集合  $A_j$ : 从集合  $A_j$  中删除在 Step5 中已选中的子区间

Step7 END FOR

Step8 END FOR

Step9 输出多样性的初始群体  $\{x^1, x^2, \dots, x^{NP}\}$

这种初始化方法执行后可获得一组在决策空间中均匀分布的初始解点, 这就为 MOFA-MCS 算法的搜索提供了一个良好的开端.

#### 3.2 随机扰动策略

从萤火虫移动公式(见式(6))右边第三项可知, 萤火虫移动的随机步长仅仅依赖于最大迭代次数, 不能适用于不同的个体, 并且随着迭代次数的增加, 每个萤火虫个体的步长差别将越来越小. 这种状况不利于多目标萤火虫算法求解具有复杂 Pareto 前沿的 MOP 问题. 为此, 这里提出在萤火虫的移动公式中施加 Lévy flights 随机扰动, 以更好地平衡萤火虫个体局部开采和全局勘探的能力.

Lévy flights 是一类非高斯随机过程<sup>[15]</sup>, 常用以步构成描述人类行程分步、生物的觅食搜索轨迹等连续随机步构成轨迹的数学形式, 也是随机游走模型中最好的策略之一. Lévy flights 轨迹是一种 Markov 随机过程, 行走的步长满足一个重尾的 Lévy 分布, 如式(7)所示:

$$L(s) \sim |s|^{-1-\beta}, 0 < \beta \leq 2 \quad (7)$$

Lévy flights 的随机步长  $s$  可由式(8)得到:

$$s = u / |v|^{1/\beta} \quad (8)$$

其中,  $u, v$  均服从正态分布,  $u \sim N(0, \sigma_u^2)$ ,  $v \sim N(0, \sigma_v^2)$ , 而  $\sigma_u$  和  $\sigma_v$  满足式(9):

$$\begin{cases} \sigma_u = \left\{ \frac{\Gamma(1+\beta) \sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \beta 2^{(\beta-1)/2}} \right\}^{1/\beta} \\ \sigma_v = 1 \end{cases} \quad (9)$$

其中  $\Gamma$  是标准的 Gamma 函数.

因此, 引入 Lévy flights 扰动后的萤火虫的公式更新为:

$$\mathbf{x}_j(t+1) = \mathbf{x}_j(t) + \beta_{ij}(r_{ij})(\mathbf{x}_i(t) - \mathbf{x}_j(t)) + \alpha \otimes s \quad (10)$$

其中  $\otimes$  表示矢量运算.

施加 Lévy flights 扰动的萤火虫搜索轨迹中, 小范围内的开采性局部搜索与随机较长距离探索性移动相间, 可以使得寻优过程中的部分萤火虫在当前最优解附近搜索, 从而加速了局部搜索; 另一部分萤火虫可以在远离当前最优解的区域内进行搜索, 从而保证了算法的全局搜索能力, 有利于逃逸局部极值区. 显然, 在 MOFA-MCS 算法中引入 Lévy flights 扰动会产生较大跳跃且方向多次急剧改变的随机移动, 能够扩大搜索范围, 增加种群多样性, 可有效解决多目标萤火虫算法在求解具有多局部 Pareto 前沿的 MOP 问题时难以逼近全

局 Pareto 前沿的问题.

### 3.3 档案精英解引导萤火虫移动

基本 FA 算法中萤火虫  $j$  被萤火虫  $i$  所吸引而向  $i$  移动至新的位置, 其中  $j$  的新位置由 3 个方面共同确定: (1)  $j$  的当前位置; (2)  $i$  和  $j$  之间的吸引力; (3) 随机扰动. 将基本 FA 算法用于求解 MOP 问题需要解决如下问题: (a) 彼此非劣的解个体之间如何移动的问题; (b) 如果利用档案精英个体引导萤火虫移动以改善算法的性能. 由于 MOFA-MCS 算法任一代档案集合中至少存在一个精英解个体, 所以, 当萤火虫  $i$  和  $j$  之间因相互吸引而更新位置时, 随机地从档案中选取一个精英解  $g^*$  引导萤火虫的移动, 因而萤火虫位置的更新可按式(11)和式(12)进行.

(i) 当萤火虫  $i$  和  $j$  彼此非劣时, 利用精英解  $g^*$  引导萤火虫移动, 则个体  $i$  和  $j$  的位置更新如式(11)所示:

$$\begin{cases} \mathbf{x}_i(t+1) = \omega_0 \mathbf{x}_i(t) + (1 - \omega_0) \beta_{g^*i}(r_{g^*i})(\mathbf{x}_{g^*}(t) - \mathbf{x}_i(t)) + 2r_{g^*i} \otimes s \text{ for firefly } i \\ \mathbf{x}_j(t+1) = \omega_0 \mathbf{x}_j(t) + (1 - \omega_0) \beta_{g^*j}(r_{g^*j})(\mathbf{x}_{g^*}(t) - \mathbf{x}_j(t)) + 2r_{g^*j} \otimes s \text{ for firefly } j \end{cases} \quad (11)$$

(ii) 当萤火虫  $i$  和  $j$  之间存在支配关系时, 例如若  $i > j$ , 则  $j$  不仅向  $i$  移动, 而且  $j$  还会向精英个体  $g^*$  移动, 式(12)给出了个体  $i$  和  $j$  的移动公式:

$$\begin{cases} \mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \omega_0 \beta_{ij}(r_{ij})(\mathbf{x}_j(t) - \mathbf{x}_i(t)) + (1 - \omega_0) \beta_{g^*i}(r_{g^*i})(\mathbf{x}_{g^*}(t) - \mathbf{x}_i(t)) + (r_{ij} + r_{g^*i}) \otimes s \text{ if } j > i \\ \mathbf{x}_j(t+1) = \mathbf{x}_j(t) + \omega_0 \beta_{ij}(r_{ij})(\mathbf{x}_i(t) - \mathbf{x}_j(t)) + (1 - \omega_0) \beta_{g^*j}(r_{g^*j})(\mathbf{x}_{g^*}(t) - \mathbf{x}_j(t)) + (r_{ij} + r_{g^*j}) \otimes s \text{ if } j > i \end{cases} \quad (12)$$

式(11)和式(12)中的  $\omega_0$  为定义在区间  $[0, 1]$  上均匀分布的随机数,  $r_{ij}$  为萤火虫  $i$  和  $j$  之间的欧氏距离,  $r_{g^*i}$  和  $r_{g^*j}$  分别为档案精英个体  $g^*$  与萤火虫  $i$  和  $j$  之间的欧氏距离,  $s$  为 Lévy flights 随机扰动,  $\otimes$  为内积运算. 式(12)中被支配个体不仅向支配个体移动, 而且还向精英解  $g^*$  靠近, 这样可以更好地平衡算法的全局勘探和局部开采. 这里采用随机方式选取档案精英个体客观上起到了改善算法性能的作用, 其原因在于: (1) 萤火虫接受精英解的引导可直接促进优良基因的继承和发展, 使得种群能够更快地逼近 Pareto 前沿, 有利于加快算法收敛; (2) 随机选取精英解个体引导萤火虫移动间接阻止种群向局部区域聚拢, 有利于拓展群体的搜索范围, 增加发现全局最优解的几率.

### 3.4 $\varepsilon$ -三点最短路径策略

MOEA 算法通常采用容量有限的外部档案保存算法迄今获得的非劣解, 当非劣解数目超过档案允许容纳的最大值时, 需要对档案进行修剪并尽可能地维持档案群体的多样性. 目前使用较多的修剪技术包括拥挤距离机制<sup>[2]</sup>、 $k$ -最近邻方法<sup>[4]</sup>以及 Deb 等提出的  $\varepsilon$ -占优<sup>[16]</sup>等. 最近, 谢等<sup>[17]</sup>提出了一种三点最短路径方法计算档案中互不相同的三点之间的距离来评估解个体的分布性, 由于三点最短路径评估解个体的分布性无需引入额外的参数, 而且能够获得多样性较好的解群而受到关注. 但三点最短路径方法通常是在目标空间的局部区域内评估解个体的分布性, 因而在全局的空间内可能会出现解个体分布不平衡的问题, 图 1 以 2-目标空间为例说明三点最短路径方法可能存在的问题.

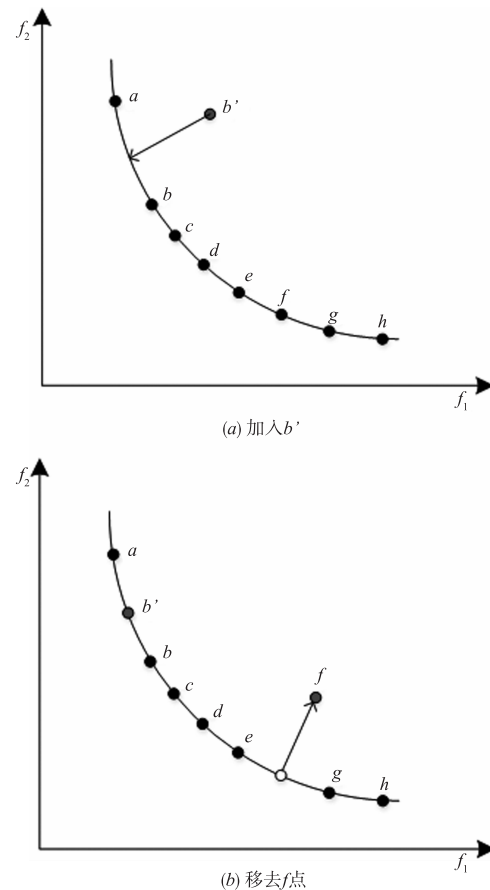


图1 三点最短路径方法示意图

例如, 若在图 1 的子图(a)中加入  $b'$  点, 则根据三点最短路径的评估方法, 由于  $b'$  点的三点最短路径距离要小于  $b$  点的三点路径距离, 从而使得  $b'$  点成为被舍弃的解点. 但结合 Pareto 前沿上所有解点来看,  $d, e, f, g$  这些点分布得较为稠密, 而  $a, b, c$  这些点分布较为稀疏, 加入  $b'$  反而会改善 Pareto 前沿左上部的解点分布的均

匀性,而移去  $d, e, f, g$  中的任一点则会使 Pareto 前沿上的解整体上分布得更均匀(如图 1(b)所示).

基于此,提出一种  $\varepsilon$ -三点最短路径策略维护档案的多样性, $\varepsilon$ -三点最短路径方法是将  $\varepsilon$ -占优策略和三点最短路径方法相结合,以更好地维护档案群体的多样性.图 2 示意了  $\varepsilon$ -三点最短路径策略.

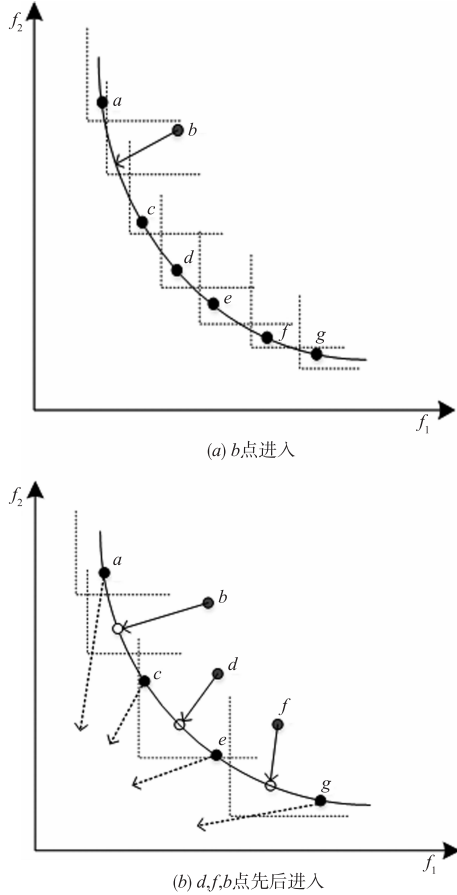


图2  $\varepsilon$ -三点最短路径方法示意图

如图 2(a)所示,假设  $b$  点进入 Pareto 前沿,根据前沿上已有点的  $\varepsilon$ -支配区域的情况, $b$  点可以进入档案.而在图 2(b)中,假设  $d$  点首先进入档案, $d$  点形成其  $\varepsilon$ -支配区域, $c$  和  $e$  两点在其区域内,需要删除  $c, e$  两点.而后, $f$  点进入档案, $g$  在  $f$  点的  $\varepsilon$ -支配区域内,则删除  $g$  点,最后  $b$  点进入档案,此时  $a$  点在  $b$  点的  $\varepsilon$ -支配区域内,删除  $a$  点.至此,档案集中留下  $b, d, f$  三个点. $\varepsilon$ -三点最短路径方法是对采用了三点最短路径策略维护的档案集合进一步施加  $\varepsilon$ -占优的保持技术,从而弥补了三点最短路径策略存在不均衡的问题,通过选取合适的参数  $\varepsilon$ ,可以获得预设规模的档案集合.

### 3.5 MOFA-MCS 算法流程

下面在第 3.1 节~3.4 节叙述的基础上给出多策略协同的多目标萤火虫算法 MOFA-MCS 的流程,如算法 2 所示.

#### 算法 2 MOFA-MCS 算法

输入:种群规模  $NP$ ,外部档案  $Arc$  规模  $N'$ ,最大迭代次数  $T_{max}$ ,光吸收系数  $\gamma$ ,萤火虫最大吸引力  $\beta_0$ ,参数  $\varepsilon$ .

输出:外部档案中  $N'$  个个体.

Step1 利用算法 1 的初始化方法产生规模为  $NP$  的初始群体  $Pop(0)$ ,  $Arc = \emptyset$ ,迭代器  $t = 0$

Step2 基于 Pareto 关系评价  $Pop(0)$ ,并将非劣解复制到  $Arc$  中

Step3 WHILE ( $t < T_{max}$ )

Step4 FOR ( $i = 0; i < NP; i++$ )

Step5 FOR ( $j = 0; j < NP; j++$ )

Step6 IF (萤火虫  $i$  和  $j$  之间存在支配关系)

Step7 从  $Arc$  中随机选取  $g^*$ ,被支配个体基于式(12)向精英解  $g^*$  和支配解移动

Step8 ELSE

Step9 从  $Arc$  中随机选取  $g^*$ ,非支配个体  $i$  和  $j$  按照式(11)执行移动

Step10 END IF

Step11 END FOR

Step12 END FOR

Step13 更新萤火虫位置,评估萤火虫群体,并利用  $\varepsilon$ -三点最短路径方法维持外部档案

Step14  $t = t + 1$

Step15 END WHILE

Step16 输出档案中的解

## 4 实验结果与分析

### 4.1 实验设置

#### (1) 对等比较算法

为检验 MOFA-MCS 算法的性能,这里另外选取 6 个代表性 MOEA 算法作为对等比较算法以检测本文所提算法的性能,它们分别是:①Deb 等提出的改进的非劣分类遗传算法 NSGA-II<sup>[2]</sup>;②Zitzler 等提出的改进型强度 Pareto 进化算法 SPEA2<sup>[4]</sup>;③Nebro 等提出的基于档案的混合分散搜索算法 AbYSS<sup>[7]</sup>;④Wang 等提出的自适应调整约束子问题的 MOEA/D-ACD 算法<sup>[6]</sup>;⑤Yang 等提出的多目标萤火虫算法 MOFA<sup>[11]</sup>;⑥Tsai 等提出基于非支配排序的多目标萤火虫算法 MONSFA<sup>[12]</sup>.为标记方便,这里将 MOFA-MCS, NSGA-II, SPEA2, AbYSS, MOEA/D-ACD, MOFA 和 MONSFA 算法分别简记为 A1, A2, A3, A4, A5, A6, A7.

#### (2) 多目标测试函数集

为检验本文算法的有效性,这里选取 12 个基准 MOP 测试题组成测试集合,以考察算法求解不同难度类型的 MOP 问题时的性能.测试题包括 5 个 2-目标的 ZDT 系列函数<sup>[18]</sup> ZDT1、ZDT2、ZDT3、ZDT4 和 ZDT6,以及 7 个 3-目标的 DTLZ 系列函数<sup>[19]</sup> DTLZ1~DTLZ7. ZDT 系列和 DTLZ 系列测试函数的 Pareto 前沿形状各异,对 MOEA 算法构成了很大的挑战.此外这些测试函

数的 Pareto 前沿是已知的,这对于算法性能的量化提供了必要的条件.

### (3) 性能指标

反转世代距离 IGD (Inverted Generational Distance)<sup>[20]</sup>度量了真实的 Pareto 前沿到算法获得的近似 Pareto 前沿之间的距离.由于 ZDT 函数集和 DTLZ 函数集的真实 Pareto 解集是已知的,通过对真实 Pareto 解集进行多样性采样,计算这些采样点到近似 Pareto 解点之间的距离既反映了算法的收敛性又能反映出算法所获解集的多样性.一般而言,IGD 指标值越小则说明近似 Pareto 前沿的收敛性和多样性越好.设  $P$  是多目标优化问题的真实 Pareto 解集,则 IGD 性能指标值可通过式 (13) 进行计算.

$$IGD = \frac{1}{|P|} \sum_{i=1}^{|P|} Dist_i \quad (13)$$

其中,  $Dist_i = \min_{j=1}^{|A|} \sqrt{\sum_{m=1}^M \left( \frac{f_m(p_i) - f_m(a_j)}{f_m^{\max} - f_m^{\min}} \right)^2}$  为归一化后的最小欧氏距离;  $f_m^{\max}$  和  $f_m^{\min}$  分别表示集合  $P$  在第  $m$  个目标上获得的最大值和最小值;  $p_i \in P, i = 1, 2, \dots, |P|$ ;  $a_j \in A, j = 1, 2, \dots, |A|$ . 实验对各测试函数采样 10000 个均匀分布的 Pareto 最优解点作为真实 Pareto 前沿的近似解集来计算 IGD 值.

### (4) 实验参数

为客观、公平地评估 MOFA-MCS 算法的性能,这里将参与对比的算法 NSGA-II、SPEA2、AbYSS、MOEA/D-ACD、MOFA 和 MONSFA 等算法的部分参数保持与对应参考文献一致的设置,表 1 给出了各对比算法的部分参数取值.

表 1 七种对比算法的参数设置

Algorithm	Parameter settings
NSGA-II	$p_c = 0.9, p_m = 1/n, \eta_c = 20, \eta_m = 20$
SPEA2	$p_c = 0.9, p_m = 1/n, \eta_c = 20, \eta_m = 20$
AbYSS	$N_{RefSet1} = 10, N_{RefSet2} = 10, p_c = 0.9, p_m = 1/n, \eta_c = 20, \eta_m = 20$
MOEA/D-ACD	$CR = 1.0, F = 0.5, p_m = 1/n, \eta_m = 20, T = 20, \delta = 0.9, n_r = 2$
MOFA	$\alpha = 0.25, \beta_0 = 1, \gamma = 1.0$
MONSFA	$\alpha = 0.25, \beta_0 = 1.0, \gamma = 1.0$
MOFA-MCS	$\alpha = 0.25, \beta_0 = 1.0, \gamma = 1.0$

表 1 中  $p_c$  表示杂交概率,  $p_m$  为变异概率,  $\eta_c$  和  $\eta_m$  分别表示仿二进制交叉算子 (SBX) 和多项式变异的分布指数. 对于 AbYSS 算法而言,  $N_{RefSet1}$  和  $N_{RefSet2}$  分别为 Refset1 和 Refset2 的规模. 在 MOEA/D-ACD 算法中,  $T$  为邻域规模的大小,  $\delta$  为从大小为  $T$  的邻域中选

取父体的概率,  $n$  则表示允许被子代替换掉的最大的父体数. MOFA-MCS 算法中与对比算法相同的参数将设置成与它们相同的值. 例如: 2-目标的种群规模  $NP = 100$ , 外部档案 Arc 的规模为 100, 每个测试函数评估 50000 次; 对于 3-目标测试问题的种群规模  $NP = 200$ , 外部档案的最大容量为 200, 各测试函数的评估次数为 200000 次.

为减少性能分析中随机因素的干扰, 每种算法在所有测试函数上均独立运行 30 次. 本文仿真实验是在笔记本电脑上运行, 电脑配置 4G 内存和 3.5GHz 双核 CPU, 安装 Windows 10 X64 操作系统, 所有算法利用 Java Myeclipse 2015 平台编程实现, 并利用 Matlab R2015b 环境出图.

## 4.2 实验结果与分析

**实验 1** 通过比较 7 个对等算法在 12 个测试函数上获得的 IGD 值来评估各算法的性能. 表 2 给出了各算法在所有测试例上获得的 IGD 均值 (Mean) 和方差 (Std.), 并利用水平为 5% 的双尾  $t$ -检验来判断算法所获结果的显著性差异. 需要指出的是, 表 3 中的均值和方差是用一种算法在相同测试问题上独立执行 30 次 (每次使用不同的随机数种子) 的统计结果.  $t$ -检验值则是本文算法与其他对比算法在同一测试函数上进行  $t$ -检验的结果. 表中的“+”, “=”和“-”分别表示本文算法获得的性能指标值在显著性水平为 5% 的双尾  $t$ -检验中优于, 等于和劣于对应列的算法在对应行的测试问题上的显著性区分结果; “Score”表示本文算法显著优于对应列的对比算法在 12 个测试问题中的净胜得分, 即得“+”与得“-”的测试问题个数之差 (下文同). 同时, 采用粗体显示不同算法在同一测试问题上获得的最优值.

从表 2 可以看出, 本文的 MOFA-MCS 算法 (A1) 在 12 个测试函数上获得了 11 个最优的 IGD 均值, NSGA-II 算法在 DTLZ4 测试问题上获得了最优的 IGD 均值, 而其余的 5 种对等比较算法在这 12 个测试范例上无一能获得最佳的 IGD 均值. 需要指出的是, 在 DTLZ4 问题上, MOFA-MCS 算法获得的 IGD 均值 ( $4.45E-03$ ) 与最佳的 IGD 均值 ( $3.93E-03$ ) 具有相同的数量级 ( $10^{-3}$ ), 表明二者获得的 IGD 均值差异不大. 这表明 MOFA-MCS 算法在 12 个测试函数上相对于其他 6 种对比算法而言具有最优的 IGD 性能指标. 从  $t$ -检验的结果来看, MOFA-MCS 算法对比 NSGA-II、SPEA2、AbYSS 算法的净胜得分均为 10, 对比 MOEA/D-ACD、MOFA 和 MONSFA 算法的净胜得分均为 12 分, 表明本文的算法在全部 12 个测试函数上的 IGD 性能指标总体上显著优于其他 6 种对等比较算法.

表 2 各算法获得的 IGD 均值和方差

Test instance		A1	A2	A3	A4	A5	A6	A7
ZDT1	Mean	<b>1.03E-02</b>	4.97E-02	5.20E-02	5.41E-02	6.31E-02	7.36E-02	7.12E-02
	Std.	8.50E-03	2.90E-03	3.70E-03	5.10E-03	4.80E-03	5.40E-03	5.60E-03
	<i>t</i> -test		+	+	+	+	+	+
ZDT2	Mean	<b>1.54E-02</b>	8.62E-02	9.02E-02	9.01E-02	1.02E-01	1.13E-01	1.11E-01
	Std.	8.40E-03	5.00E-03	4.20E-03	6.30E-03	7.90E-03	6.20E-03	7.80E-03
	<i>t</i> -test		+	+	+	+	+	+
ZDT3	Mean	<b>7.91E-03</b>	2.81E-02	2.94E-02	2.97E-02	3.35E-02	3.91E-02	3.94E-02
	Std.	3.50E-03	1.30E-03	1.50E-03	1.60E-03	2.80E-03	3.30E-03	3.40E-03
	<i>t</i> -test		+	+	+	+	+	+
ZDT4	Mean	<b>7.99E-01</b>	1.51E+00	1.51E+00	1.73E+00	1.69E+00	1.96E+00	2.35E+00
	Std.	3.90E-02	2.10E-01	1.70E-01	1.90E-01	2.80E-01	3.50E-01	2.40E-01
	<i>t</i> -test		+	+	+	+	+	+
ZDT6	Mean	<b>1.94E-02</b>	1.96E-01	2.03E-01	2.03E-01	2.16E-01	2.33E-01	2.33E-01
	Std.	8.40E-03	8.80E-03	5.60E-03	9.50E-03	9.40E-03	6.60E-03	8.90E-03
	<i>t</i> -test		+	+	+	+	+	+
DTLZ1	Mean	<b>7.49E-03</b>	2.57E-01	2.85E-01	3.53E-01	3.64E-01	8.53E-01	5.08E-01
	Std.	0.00E+00	5.60E-02	7.00E-02	7.70E-02	5.50E-02	2.70E-01	1.90E-01
	<i>t</i> -test		+	+	+	+	+	+
DTLZ2	Mean	<b>1.67E-03</b>	1.95E-03	1.98E-03	2.16E-03	2.76E-03	4.41E-03	4.37E-03
	Std.	1.30E-04	7.50E-05	9.80E-05	1.20E-04	1.20E-04	2.40E-04	3.60E-04
	<i>t</i> -test		+	+	+	+	+	+
DTLZ3	Mean	<b>1.37E-02</b>	3.44E+00	3.71E+00	4.34E+00	4.22E+00	7.61E+00	5.89E+00
	Std.	3.30E-10	6.10E-01	6.00E-01	5.70E-01	6.00E-01	1.10E+00	1.30E+00
	<i>t</i> -test		+	+	+	+	+	+
DTLZ4	Mean	4.45E-03	<b>3.93E-03</b>	4.29E-03	4.29E-03	5.10E-03	8.73E-03	8.34E-03
	Std.	5.60E-04	2.40E-04	3.00E-04	3.30E-04	3.80E-04	1.20E-03	1.10E-03
	<i>t</i> -test		-	-	-	+	+	+
DTLZ5	Mean	<b>2.14E-04</b>	4.07E-04	4.43E-04	4.97E-04	6.39E-04	1.20E-03	1.20E-03
	Std.	5.30E-05	3.30E-05	3.20E-05	4.50E-05	5.40E-05	1.80E-04	1.80E-04
	<i>t</i> -test		+	+	+	+	+	+
DTLZ6	Mean	<b>6.35E-03</b>	4.91E-02	4.89E-02	4.88E-02	5.10E-02	5.24E-02	5.23E-02
	Std.	1.00E-04	8.30E-04	6.00E-04	9.40E-04	1.00E-03	1.20E-03	9.00E-04
	<i>t</i> -test		+	+	+	+	+	+
DTLZ7	Mean	<b>3.19E-02</b>	6.70E-02	7.17E-02	7.32E-02	7.58E-02	1.01E-01	9.84E-02
	Std.	9.10E-03	5.30E-03	4.50E-03	4.60E-03	3.90E-03	6.50E-03	9.90E-03
	<i>t</i> -test		+	+	+	+	+	+
Better( + )			11	11	11	12	12	12
Worse( - )			1	1	1	0	0	0
Same( = )			0	0	0	0	0	0
Score			10	10	10	12	12	12

表3采用Friedman检验给出了7种对比算法的平均排名,可以看出,MOFA-MCS算法的名次是最好的,次优的是NSGA-II算法,随后依次是SPEA2、AbYSS、MOEA/D-ACD和MONSFA,排名最差的则是MOFA. Friedman检验的结果与表2中各算法所获得的IGD值所反映的结果吻合.表2和表3的结果均表明MOFA-MCS较之其他6种对比算法具有最优的IGD性能.究其原因:MOFA-MCS算法采用均匀性与随机性相结合的种群初始化方法产生均匀分布于搜索空间的解点,这为后续的进化过程提供了良好的开端;其次,算法利用档案精英个体指导萤火虫移动,有利于促进算法的收敛;最后,算法采用 $\epsilon$ -三点最短路径策略有效地维护了档案解群的多样性.以上多种策略有机结合,协同地提高了算法的收敛性与多样性的总体性能.

表3 七种对比算法基于IGD指标的平均排名

平均排名	算法	秩
1	MOFA-MCS	1.0821
2	NSGA-II	2.1428
3	SPEA2	3.2499
4	AbYSS	4.0714
5	MOEA/D-ACD	5.2142
6	MOFA	5.8928
7	MONSFA	5.6071

## 5 结论

现实中的多目标优化问题不断产生且日益复杂,需要发展新的有效的方法应对挑战.近些年来,基于新型启发式算法和协同策略的多目标优化算法逐渐成为多目标优化领域的新的研究热点.本文提出一种多策略协同的多目标萤火虫算法MOFA-MCS,该算法将均匀化和随机化相结合产生均匀分布于决策空间的初始群体,在萤火虫的移动中施加Lévy flights随机扰动,并利用档案精英解指导萤火虫的移动,最后采用 $\epsilon$ -三点最短路径方法维持档案解群的多样性.实验结果表明,本文的MOFA-MCS相对于其他6种对等比较算法而具有总体上显著较优的IGD性能指标,表明了MOFA-MCS算法是一种有前途的多目标进化算法.未来将从两个方面进一步做工作:(1)利用更多、更困难的MOP问题测试MOFA-MCS算法;(2)将MOFA-MCS算法应用于工程实践中MOP问题的求解.

### 参考文献

- [1] SRINIVAS N, DEB K. Multi-objective optimization using non-dominated sorting in genetic algorithms [J]. *Evolutionary Computation*, 1994, 2(3): 221 - 248.
- [2] DEBK, PRATAP A, AGARWAL S, et al. A fast and elitist multi-objective genetic algorithm: NSGA-II [J]. *IEEE Transactions on Evolutionary Computation*, 2002, 6(2): 182 - 197.
- [3] ZITZLER E, THIELE L. Multi-objective evolutionary algorithms: A comparative case study and the strength Pareto approach [J]. *IEEE Transactions on Evolutionary Computation*, 1999, 3(4): 257 - 271.
- [4] ZITZLER E, LAUMANN S M, THIELE L. SPEA2: Improving the strength Pareto evolutionary algorithm [A]. *Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems* [C]. Berlin: Springer-Verlag, 2002. 95 - 100.
- [5] ZHANG Q, LI H. MOEA/D: A multiobjective evolutionary algorithm based on decomposition [J]. *IEEE Transactions on Evolutionary Computation*, 2007, 11(6): 712 - 731.
- [6] WANG LP, ZHANG QF, ZHOU AM, et al. Constrained subproblems in a decomposition-based multi-objective evolutionary algorithm [J]. *IEEE Transactions on Evolutionary Computation*, 2016, 20(3): 475 - 480.
- [7] Nebro AJ, Luna F, Alba E, et al. AbYSS: Adapting scatter search to multi-objective optimization [J]. *IEEE Transactions on Evolutionary Computation*, 2008, 12(4): 439 - 457.
- [8] 谢承旺, 许雷, 赵怀瑞, 等. 应用精英反向学习的多目标烟花爆炸算法 [J]. *电子学报*, 2016, 44(5): 1180 - 1188. XIE Cheng-wang, XU Lei, ZHAO Huai-rui, et al. Multi-objective fireworks optimization algorithm using elite opposition-based learning [J]. *Acta Electronica Sinica*, 2016, 44(5): 1180 - 1188. (in Chinese)
- [9] 谢承旺, 许雷, 汪慎文, 等. 一种增强型多目标烟花爆炸算法 [J]. *电子学报*, 2017, 45(10): 2323 - 2331. XIE Cheng-wang, XU Lei, WANG Shen-wen, et al. Enhanced multi-objective fireworks explosion optimization algorithm [J]. *Acta Electronica Sinica*, 2017, 45(10): 2323 - 2331. (in Chinese)
- [10] YANG X S. *Nature-Inspired Metaheuristic Algorithms* [M]. US: Luniver Press, 2008.
- [11] YANG X S. Multiobjective firefly algorithm for continuous optimization [J]. *Engineering with Computers*, 2013, 29(2): 175 - 184.
- [12] TSAI C W, HUANG Y T, CHIANG M C. A non-dominated sorting firefly algorithm for multi-objective optimization [A]. *International Conference on Intelligent Systems Design and Applications* [C]. US: IEEE, 2015. 62 - 67.
- [13] 谢承旺, 肖驰, 丁立新, 等. HMOFA: 一种混合型多目标萤火虫算法 [J]. *软件学报*, 2018, 29(4): 1143 - 1162. XIE CW, XIAO C, DING LX, et al. HMOFA: A hybrid multi-objective firefly algorithm [J]. *Journal of Software*,

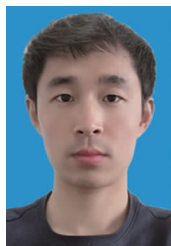
- 2018,29(4):1143-1162. (in Chinese)
- [14] 谢承旺,邹秀芬,夏学文,等.一种多策略融合的多目标粒子群优化算法[J].电子学报,2015,43(8):1538-1544.  
XIE Cheng-wang, ZOU Xiu-fen, XIA Xue-wen, et al. A multi-objective particle swarm optimization algorithm integrating multiply strategies[J]. Acta Electronica Sinica, 2015, 43(8): 1538-1544. (in Chinese)
- [15] YANG X S, DEB S. Eagle strategy using lévy walk and firefly algorithms for stochastic optimization[J]. Studies in Computational Intelligence, 2010, 284: 101-111.
- [16] LAUMANN S M, THIELE L, DEB K, et al. Combining convergence and diversity in evolutionary multi-objective optimization[J]. Evolutionary Computation, 2002, 10(3): 263-282.
- [17] 谢承旺,王志杰,夏学文.应用档案精英学习和反向学习的多目标进化算法[J].计算机学报,2017,40(3): 757-772.  
XIE Cheng-wang, WANG Zhi-Jie, XIA Xue-Wen. Multi-objective evolutionary algorithm based on archive-elite learning and opposition-based learning[J]. Chinese Journal of Computers, 2017, 40(3): 757-772. (in Chinese)
- [18] ZITZLER E, DEB K, THIELE L. Comparison of multi-objective evolutionary algorithms: Empirical results[J]. Evolutionary Computation, 2000, 8(2): 173-195.
- [19] DEB K, THIELE L, LAUMANN S M, et al. Scalable multi-objective optimization test problems[A]. Proceedings of the IEEE Congress on Evolutionary Computation (CEC) [C]. Piscataway: IEEE Service Center, 2002. 825-830.
- [20] BOSMAN P A N, THIERENS D. The balance between proximity and diversity in multiobjective evolutionary algorithms[J]. IEEE Transactions on Evolutionary Computation, 2003, 16(1): 51-69.

#### 作者简介



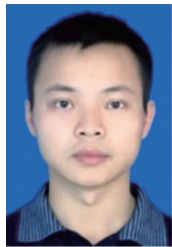
谢承旺 男,1974年10月生于湖北武汉.计算机博士,数学博士后,CCF高级会员,南宁师范大学计算机与信息工程学院教授,硕士生导师,主要研究方向为智能计算与多目标优化等.

E-mail:chengwangxie@163.com



张飞龙 男,1992年6月生于河南周口.硕士研究生,主要研究领域为智能计算与多目标优化.

E-mail:862677277@qq.com



陆建波(通信作者) 男,1977年1月生于广西灵山.2003年毕业于广西大学控制理论控制工程专业,获硕士学位,南宁师范大学计算机与信息工程学院副教授,硕士生导师,主要研究领域为智能计算、软件工程等.

E-mail:lujianbo@nnnu.edu.cn