

双编码动态培育遗传聚类算法及其在电池定制化配组中的应用

王子赞[†], 史伟杰, 王 艳, 纪志成

(江南大学 物联网技术应用教育部工程研究中心, 江苏 无锡 214122)

摘要: 本文提出了一种基于双编码动态培育遗传聚类算法, 同步编码聚类类别与聚类中心, 引入动态培育思想, 动态调整交叉对象与变异方向, 进而增加微变异进而提高种群多样性, 解决了传统遗传算法初期收敛缓慢和种群单一问题, 提高了算法后期跳出局部最优的能力. 同时, 分析了双编码动态培育遗传聚类算法的收敛性. 最后, 以电池定制化配组为应用案例, 验证了本文算法解决动力电池定制化配组问题的有效性, 以及算法在收敛速度和全局搜索精度方面的优越性.

关键词: 双编码; 动态培育; 遗传聚类; 电池定制化配组

引用格式: 王子赞, 史伟杰, 王艳, 等. 双编码动态培育遗传聚类算法及其在电池定制化配组中的应用. 控制理论与应用, 2025, 42(1): 118 – 126

DOI: 10.7641/CTA.2024.30457

Double-coded dynamic breeding based genetic clustering algorithm and its application in customized battery grouping

WANG Zi-yun[†], SHI Wei-jie, WANG Yan, JI Zhi-cheng

(Engineering Research Center of Internet of Things Technology Applications (Ministry of Education),
Jiangnan University, Wuxi Jiangsu 214122, China)

Abstract: A genetic clustering algorithm based on double coding dynamic cultivation is proposed, which encodes clustering categories and cluster centers at the same time, introduces the idea of dynamic cultivation, dynamically adjusts the cross object and mutation direction, increases micro-variation and thus increases population diversity. It solves the problem of slow convergence and single population in the initial stage of traditional genetic algorithm, and improves the ability of the algorithm to jump out of the local optimum in the later stage. At the same time, the convergence of the double-coded dynamic breeding genetic clustering algorithm is analyzed. Finally, this paper takes the battery customized combination as an application case and verifies that the algorithm is effective in solving the problem of customized matching of power battery, as well as its superiority in convergence speed and global search accuracy.

Key words: double-coded; dynamic breeding; genetic clustering; customized battery grouping

Citation: WANG Ziyun, SHI Weijie, WANG Yan, et al. Double-coded dynamic breeding based genetic clustering algorithm and its application in customized battery grouping. *Control Theory & Applications*, 2025, 42(1): 118 – 126

1 引言

群智能优化算法是一种解决复杂问题的高效算法^[1], 遗传算法是其算法的一种, 通过模拟自然生物进化过程来解决复杂优化问题. 近年来, 运用遗传算法解决聚类问题得到学者们的广泛关注, 主要的研究方向可以分为两种^[2-3]: 利用遗传算法改进聚类算法的参数选取^[4]和基于遗传算法的聚类算法研究^[5].

目前, 对于聚类算法的优化主要集中在聚类参数

的改进. 例如, 宋飞豹等^[6]利用遗传算法在K-means聚类算法进入迭代前计算出最佳的聚类质心分配方案, 改进聚类结果. 此外, 其他智能优化算法也可以进行初始聚类中心点的选取或是其他参数改进, 从而改进聚类算法^[7], 但是这些优化算法取得的良好聚类效果, 主要是依靠基于划分的聚类算法优化能力, 而非单一通过遗传等智能优化算法设计. 与之对应的是, 在基于遗传算法实现聚类效果的算法实现过程中, 遗

收稿日期: 2023-07-01; 录用日期: 2024-02-27.

[†]通信作者. E-mail: wangzy0601@163.com; Tel.: +86 13771237227.

本文责任编辑: 张承慧.

国家重点研发计划项目(2020YFB1710600), 江苏省自然科学基金面上项目(BK20221533), 江苏省科协青年科技人才托举工程项目(TJ-2021-006)资助.

Supported by the National Key Research and Development Program of China (2020YFB1710600), the Natural Science Foundation of Jiangsu Province (BK20221533) and the Jiangsu Science and Technology Association Young Science and Technology Talents Lifting Project (TJ-2021-006).

传算法的构建对于最终聚类结果的优化起主导性作用^[8]. 因此, 针对遗传算法自身的优化是当前研究领域的新热点. 现阶段, 遗传算法优化领域的研究成果主要围绕交叉与变异算子设计以及选择更新方式改进. 例如, 马速良等^[9]以类别编码染色体, 提出了基于聚类思想的改进遗传优化算法, 其本质是类别单编码遗传聚类算法. 然而, 采用类别编码的优化方式没有充分考虑到聚类效果高度依赖于聚类中心这一特点, 同时交叉与变异主要依靠随机选取, 没有考虑到精英保留对种群多样性的影响.

针对以上问题, 本文提出一种双编码动态培育遗传聚类算法(double-coded dynamic breeding based genetic clustering algorithm, DCDB-GCA), 并将其运用在动力电池定制化配组领域^[10].

2 问题描述与变量定义

2.1 问题描述

本文所针对的聚类问题可以描述为: 将一组对象依据提供的指标和一定的权重进行划分, 在不同定制划分需求下, 指标权重的选取是不一致的. 因此, 本文需要首先确定簇数, 在此基础上, 采用指标加权后的欧式距离作为相似性评估依据, 每个簇内的个体之间的相似度尽可能高.

2.2 模型建立

针对聚类问题, 由于不同指标数据的基础范围不同, 需先将对象指标数据归一化处理, 后续特征变量均为归一化对象指标数据. 针对 s 个对象提取 m 个指标后进行归一化处理, 得到归一化后的第 i 个对象的第 j 个特征变量为

$$b_{ij} = \frac{\tilde{b}_{ij} - \min(\tilde{b}_j)}{\max(\tilde{b}_j) - \min(\tilde{b}_j)} \times 100\%. \quad (1)$$

传统的聚类算法都是利用欧氏距离, 设计基于指标即对象特征变量的目标函数, 通过遍历不同类中各个对象的特征变量与该类平均点的欧氏距离和, 更新对象类属性. 目标函数 $F(b, z)$ 描述为

$$F(b, z) = \sum_{z=1}^Z \sqrt{\sum_{i=1}^{\text{num}_z} \sum_{j=1}^m |b_{ij} - \bar{b}_j|^2}. \quad (2)$$

设定聚类指标权重值集合 $W = [w_1 \cdots w_m]$, W 由聚类问题本身决定, 确定各项指标权重后, 将指标权重值集合 W 引入目标函数, 得到更新后的目标函数为

$$F(b, z, W) = \sum_{z=1}^Z \sqrt{\sum_{i=1}^{\text{num}_z} \sum_{j=1}^m \{w_j \times |b_{ij} - \bar{b}_j|^2\}}. \quad (3)$$

2.3 变量说明

在介绍本文主要内容之前, 先定义本文涉及到的变量, 变量的选取及含义如表1所示.

表1 算法变量选取定义

Table 1 Algorithm variable selection definitions

变量名称	含义
s	所需要分类的总对象数量
m	分类对象特征数量
b	归一化后的对象信息
b_i	归一化后第 i 个对象特征变量向量
b_{ij}	归一化后的第 i 个对象的第 j 个特征变量
\bar{b}_j	归一后所有对象的第 j 个特征变量的平均值
\tilde{b}_{ij}	归一化前第 i 个对象的第 j 个特征变量
$\min(\tilde{b}_j)$	归一化前所有对象的第 j 个特征变量的最小值
$\max(\tilde{b}_j)$	归一化前所有对象的第 j 个特征变量的最大值
Z	筛选类数
z	第 z 个类别
num_z	第 z 个类别对象的数量
w_j	第 j 个特征变量的综合权重
W	特征权重集合
F	目标函数
k	迭代次数
$I(k)$	k 次迭代下的种群
$I_{iz}(k)$	k 次迭代中第 i 条染色体下属于第 z 个类的对象集合
$A(k)$	第 k 次迭代的种群集合
n	固定种群个体数量
$C_i(k)$	第 k 次迭代的第 i 条染色体
$c_{ij}(k)$	第 k 次迭代的第 i 条染色体第 j 个基因位
$\tilde{C}_i(k)$	第 k 次迭代的第 i 条染色体的优化个体
$\tilde{c}_{ij}(k)$	第 k 次迭代的第 i 条染色体优化个体第 j 个基因位
l	个体编码长度
E_l	$l \times l$ 的单位阵
C_1	随机变异变换方向向量
$O_i(k)$	第 k 次迭代中第 i 个编码序列的目标函数值
$\min(O(k))$	第 k 次迭代中编码序列最小目标函数值
$\max(O(k))$	第 k 次迭代中编码序列最大目标函数值
acr	交叉率
mut	变异率
rep_mut	淘汰率
iter	最大迭代次数
rand	0~1的随机数

3 DCDB-GCA算法原理及实现

下面围绕DCDB-GCA算法的编码方式、目标函数定义、种群初始化、动态培育、自然选择微变异精英保留以及最佳个体记录, 介绍DCDB-GCA算法的设计思路.

3.1 个体染色体编码

首先, 由分类对象指标属性确定聚类数据集 $I = [b_1 \ b_2 \ \cdots \ b_i \ \cdots \ b_s]^T$, 其中 $b_i = [b_{i1} \ b_{i2} \ \cdots \ b_{il}]$. 遗传种群的个体染色体内容随着迭代次数 k 更新, 但染色体数量固定, 因此染色体的编码长度

$$l = s + Z \times m. \quad (4)$$

第 k 次迭代中第 i 条染色体编码构造如图1所示.

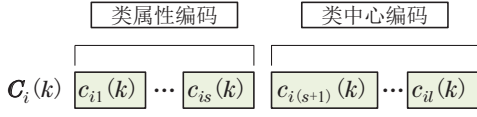


图1 第 k 次迭代中第 i 个个体的染色体编码构造

Fig. 1 Chromosome encoding construction of the i th individual in iteration k

图1中, 第 k 次迭代中第 i 条染色体编码序列为 $\mathbf{C}_i(k) = [c_{i1}(k) \ c_{i2}(k) \ \cdots \ c_{il}(k)]^T$, 前 s 位编码为对象类别, 即

$$c_{ij}(k) \in [1, Z], \quad j \in [1, s], \quad (5)$$

后续 $Z \times m$ 位编码为类特征变量中心, 即

$$0 < c_{ij}(k) \leq 1, \quad j \in [s+1, l]. \quad (6)$$

3.2 分类说明

由分类实际情况提前确定 Z 个类别, 第 k 次迭代中的第 i 个染色体编码下第 z 个对象类集合可表示为

$$\mathbf{I}_{iz}(k) = \{b_i | c_{ij}(k) = z\}, \quad j \in [1, s]. \quad (7)$$

3.3 定义目标函数

由编码序列下聚类的统计结果形成 Z 个对象特征变量子集, 根据式(3)计算此次分类的适应度函数即目标函数值, 其中第 k 次迭代中第 i 个编码序列下的目标函数值为

$$O_i(k) = F(b, z, \mathbf{W}, k), \quad (8)$$

其中分类对象类别 z 根据获得的 $\mathbf{I}_{iz}(k)$ 结果进行判断.

3.4 种群初始化

在迭代次数为0时, 初始化 n 个种群个体, 构成同一种群. 为保证遗传算法的初始种群的充分随机性, 个体的染色体初始编码采用类别随机, 类中心由各个类别包含的特征变量平均值确定, 再进行种群初始化建立, 初始化种群个体及编码如图2所示.

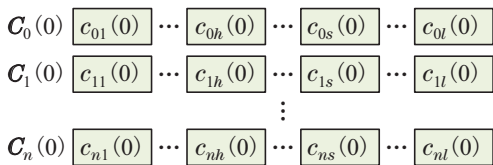


图2 种群个体及其染色体编码初始化

Fig. 2 Population individuals and their chromosome encoding initialization

3.5 动态交叉

传统的遗传算法交叉对象多为种群内随机对象或最优个体, 前者导致算法收敛速度缓慢, 后者使得算

法过早陷入局部最优解.

为解决上述问题, 本文设计动态交叉对象, 其中第 k 次迭代下第 i 个个体的染色体 $\mathbf{C}_i(k)$ 依据欧式距离的划分聚类思维得到优化个体染色体 $\tilde{\mathbf{C}}_i(k)$, 种群中各个个体的交叉对象为自身的优化个体. 同时, 根据每个点位的交叉与否由交叉算子和交叉概率上限值进行比较, 利用0和1构建交叉变换矩阵 α , 其中1代表点位需要交叉, 0相反. 且交叉算子以目标函数值为依据, 具备全局自适应的动态调整能力

$$a_1 = (1 - \frac{k}{\text{iter}}) \times \frac{O_i(k)}{\min(O(k))} \times \text{rand}, \quad (9)$$

以第 h 位($h \in [1, l]$)进行单点交叉为例, 动态交叉流程如图3所示.

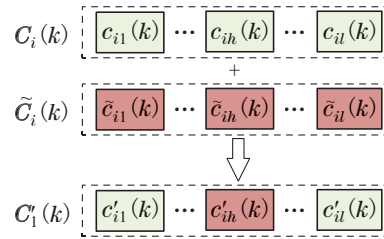


图3 动态交叉原理示意图

Fig. 3 Schematic diagram of dynamic crossover principle

动态交叉具体包括

$$\begin{cases} \mathbf{C}_i(k) = [c_{i1}(k) \ c_{i2}(k) \ \cdots \ c_{il}(k)]^T, \\ \mathbf{C}'_i(k) = (\mathbf{E}_1 - \alpha)\mathbf{C}_i(k) + \alpha\tilde{\mathbf{C}}_i(k), \quad i \in [1, n], \\ \alpha = \text{diag}\{\alpha_1, \alpha_2, \cdots, \alpha_l\}_{l \times l}, \\ \mathbf{I}_1(k) = [\mathbf{C}'_1(k) \ \cdots \ \mathbf{C}'_s(k)], \end{cases} \quad (10)$$

其中: $\mathbf{C}'_i(k)$ 为经过交叉变换后的 $\mathbf{C}_i(k)$ 个体染色体编码, $\mathbf{I}_1(k)$ 为交叉变换后的染色体种群.

3.6 动态变异

本文动态变异实现思路为: 变异只在染色体前 n 个点位进行, 根据交叉变换的 $\mathbf{C}'_i(k)$ 与其优化个体染色体 $\tilde{\mathbf{C}}'_i(k)$, 迭代前期变异方向随机, 迭代后期变异方向与 $\tilde{\mathbf{C}}'_i(k)$ 方向相反.

参考前面交叉变换矩阵构建思路, 构建变异变换矩阵 β . 变异算子以目标函数值为依据, 具备全局自适应的动态调整能力, 即

$$a_2 = (1.5 - \frac{k}{\text{iter}}) \times \frac{\min(O(k))}{O_i(k)} \times \text{rand}. \quad (11)$$

不失一般性, 以迭代步中间段到3/4迭代步动态调整变异方向, 其余为随机变异为例. 对于第 k 次迭代中第 i 条染色体第 h 位($h \in [1, s]$), 当 $k < \text{iter}/2$ 或 $k > 3\text{iter}/4$ 时, 此时为随机变异, 随机变异原理如图4所示, 变异点位选取 $1 \sim Z$ 内的随机数.

随机变异过程包括

$$\left\{ \begin{array}{l} \mathbf{C}'_i(k) = [c'_{i1}(k) \ c'_{i2}(k) \ \cdots \ c'_{il}(k)]^T, \\ \mathbf{C}''_i(k) = (\mathbf{E}_1 - \beta)\mathbf{C}'_i(k) + \beta\mathbf{C}_1, \\ \beta = \text{diag}\{\beta_1, \beta_2, \cdots, \beta_n, 0, \cdots, 0\}_{l \times l}, \\ \mathbf{I}_2(k) = [\mathbf{C}'_1(k) \ \cdots \ \mathbf{C}'_s(k)], \\ \mathbf{C}_1 = \text{diag}\{\text{ceil}(Z \times \text{rand})_1, \cdots, \\ \text{ceil}(Z \times \text{rand})_n, 0, \cdots, 0\}_{l \times 1}, \end{array} \right. \quad c = \text{ceil}(n \times \text{rand}), \quad (14)$$

当 $k \in [1, \text{iter}/2] \cap [3\text{iter}/4 + 1, \text{iter}]$, $\mathbf{C}''_i(k)$ 为经过随机变异后的 $\mathbf{C}'_i(k)$ 个体染色体编码, \mathbf{C}_1 为随机变异变换方向向量, 前 n 为类别随机整数, ceil 为向上取整函数, rand 为 0 到 1 的随机数, $\mathbf{I}_2(k)$ 为经历过交叉与变异的种群。

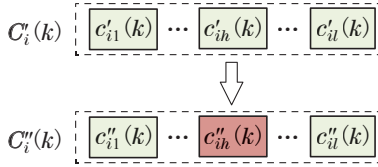


图 4 随机变异原理示意图

Fig. 4 Schematic diagram of random variation principle

在动态变异环节, 对于第 k 次迭代时染色体与其优化个体的第 h 位 ($h \in [1, l]$, 变异方向与优化个体方向相反。当 $\text{iter}/2 < k < 3\text{iter}/4$ 时, 此时动态变异原理图如图 5 所示。

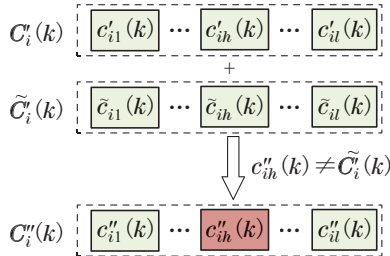


图 5 动态变异原理示意图

Fig. 5 Schematic diagram of dynamic variation principle

动态变异过程如下:

$$\left\{ \begin{array}{l} \mathbf{C}'_i(k) = [c'_{i1}(k) \ c'_{i2}(k) \ \cdots \ c'_{il}(k)]^T, \\ \mathbf{C}''_i(k) = (\mathbf{E}_1 - \beta)\mathbf{C}'_i(k) + \beta\tilde{\mathbf{C}}'_i(k), \\ \beta = \text{diag}\{\beta_1, \beta_2, \cdots, \beta_n, 0, \cdots, 0\}_{l \times l}, \\ \tilde{\mathbf{C}}'_i(k) = [\tilde{c}'_{i1}(k) \ \tilde{c}'_{i2}(k) \ \cdots \ \tilde{c}'_{il}(k)]^T, \\ \mathbf{I}_2(k) = [\mathbf{C}'_1(k) \ \cdots \ \mathbf{C}'_s(k)]. \end{array} \right. \quad (13)$$

3.7 自然选择微变异精英保留

本文的自然选择微变异精英保留步, 与传统轮盘赌的方式不同, 本文自然选择充满随机性, 个体进入下一代的几率完全相同, 在更新完种群后进行微变异, 增加多样性, 并执行精英保留操作。随机性表示为

其中 ceil 为向上取整函数。以 c 选中染色体进入下一代, 该步选择进行 n 次, 构建自然选择矩阵 \mathbf{Q}_u 。

在自然选择之后将更新得到的染色体进行微变异, 在一定程度上减少过早成熟。微变异方式为纯随机变异, 仅提高染色体多样性, 但是点位变异率 rep_mut 减小。依据随机变异的方式构建微变异矩阵 \mathbf{W}_u 。

对于微变异的染色体种群采用精英保留, 将 $\mathbf{A}(k)$ 的目标函数从小到大排序, 选择前 10% 微变异后的种群个体, 通过目标函数比较确定是否进行保留替换, 构建精英保留变换矩阵 \mathbf{S}_u 。

因此, 得出优化后的种群为

$$\mathbf{I}(k+1) = \mathbf{Q}_u \times \mathbf{W}_u \times \mathbf{S}_u \times \mathbf{I}_3(k). \quad (15)$$

3.8 记录最佳个体

更新记录最佳目标函数以及所对应的染色体个体情况 (第 1 次为初始记录), 判断迭代次数是否满足设定迭代次数, 未满足, 则重新进入迭代; 达到设定迭代次数, 则算法结束, 输出最佳个体, 并将其解码, 作为最终的分类型。

4 DCDB-GCA 算法收敛性分析

本文提出的双编码动态培育遗传聚类算法与经典遗传算法相比, 交叉和变异虽然是动态变异, 但是本质上还是随机变化, 可以看成是 $\mathbf{C}_i(k-1)$ 到 $\mathbf{C}_{i1}(k)$, 以及 $\mathbf{C}_{i1}(k)$ 到 $\mathbf{C}_{i2}(k)$ 的随机变换, 且只与上一代种群有关, 符合马尔科夫链特点。另外针对选择复制环节, 本算法优化为自然选择与微变异且进行精英保留操作。

本文算法没有选择操作, 因此与传统的遗传算法收敛性证明相比^[11], 选择概率 \mathbf{S} 为单位阵, 不计入计算过程, 动态交叉后概率变为 \mathbf{H} , 动态变异概率变为 \mathbf{M} , 淘汰重插操作为目标函数比较随机阈值, 进行重插操作, 保留的概率记作 \mathbf{G} , 保留选中后变异率为 \mathbf{M}_1 , 由遗传算法引起的概率变异为 $\mathbf{P}_1 = \mathbf{H} \times \mathbf{M} \times \mathbf{G} \times \mathbf{M}_1$ 。

考虑染色体种群空间有限记为 V , 其基数记作 $N = |V|$, 染色体编码方式不限。每次迭代的种群看作一个状态, 取每次迭代种群中最佳个体为 $Z(k)$, 针对问题的最佳个体为 f^* , 因此, 交叉 \mathbf{H} 、变异 \mathbf{M} 、淘汰重插 \mathbf{G} 、重插变异 \mathbf{M}_1 均可以看作从当前迭代次数下 V 状态到新 V 状态的随机调整变换, 状态转移矩阵 \mathbf{H} , \mathbf{M} , \mathbf{G} , \mathbf{M}_1 都与时间无关, 只与当前状态有关, 故具有马尔科夫性和齐次性, 因此都可以看作随机矩阵^[12]。

定义 1 设 Z 阶方阵 $\mathbf{A} = (a_{ij})$, 满足

1) 若 $a_{ij} > 0$, $i, j \in [1, Z]$, 则 \mathbf{A} 可以称为正的, 记作 $\mathbf{A} > 0$ 。

2) 若 $\mathbf{A} \geq 0$, 且 $\sum_{j=1}^Z a_{ij} = 1, i \in [1, Z]$, 则可称该方阵 \mathbf{A} 为一个随机矩阵.

3) 若 $\mathbf{A} \geq 0$, 且 \mathbf{A} 每列至少有一个元素大于 0, 则称 \mathbf{A} 为列可容许的^[13].

引理 1 若 \mathbf{P} 是一个基本随机矩阵, 当 $t \rightarrow \infty$ 时, \mathbf{P}^t 收敛于正定稳定的随机矩阵, 即 $\mathbf{P}^\infty = \mathbf{P}^0 \times \lim_{t \rightarrow \infty} \mathbf{P}^t = \mathbf{P}^0 \times \mathbf{P}^\infty$ 是随机阵, \mathbf{P}^0 为初始分布状态^[13].

引理 2 \mathbf{P} 是一个 y 阶可约随机阵, \mathbf{C} 是 x 阶基本随机阵, 且 \mathbf{R} 与 \mathbf{T} 为非零阵. 于是有

$$\mathbf{P} = \begin{pmatrix} \mathbf{C} & \mathbf{0} \\ \mathbf{R} & \mathbf{T} \end{pmatrix}, \quad (16)$$

$$\lim_{t \rightarrow \infty} \mathbf{P}^t = \lim_{t \rightarrow \infty} \begin{pmatrix} \mathbf{C}^t & \mathbf{0} \\ \sum_{i=0}^{t-1} \mathbf{T}^i \mathbf{R} \mathbf{C}^{t-1-i} & \mathbf{T}^t \end{pmatrix} = \begin{pmatrix} \mathbf{C}^\infty & \mathbf{0} \\ \mathbf{R}^\infty & \mathbf{0} \end{pmatrix}, \quad (17)$$

随机阵 $\mathbf{P}^\infty = \mathbf{P}^0 \times \mathbf{P}^\infty$, 与初始分布无关, 都是唯一的. 根据引理 1, \mathbf{P}^∞ 非负, 对于 $x < i \leq y$, 满足 $P_{ix}^\infty = 0$ ^[13].

定义 2 假定每次迭代种群中最佳个体 $Z(k)$, 解决聚类问题的最佳个体为 f^* , 遗传算法收敛到全局最优值, 当且仅当^[14]

$$\lim_{k \rightarrow \infty} P\{Z(k) = f^*\} = 1, \quad (18)$$

其中: k 为迭代次数, $k \rightarrow \infty$ 表示无限的迭代步.

基于文献[13]可知, \mathbf{M} 与 \mathbf{M}_1 均为正随机矩阵, 两者的变异方式一致, 且鉴于 \mathbf{M} 为常数随机矩阵, 选取第 k 次迭代下的状态空间 V 中, 变异前后状态 $\mathbf{C}_i(k)$ 与 $\mathbf{C}_j(k)$, $\mathbf{M}_{N \times N} = (m_{ij})_{N \times N}$, 从 $\mathbf{C}_i(k)$ 变异到 $\mathbf{C}_j(k)$ 的概率用 m_{ij} 表示, 有

$$\forall i, j, m_{ij} = \text{num}\{\mathbf{C}_i(k) \xrightarrow{\mathbf{M}} \mathbf{C}_j(k)\} > 0. \quad (19)$$

定理 1 \mathbf{G} 为列容许的随机矩阵.

证 \mathbf{G} 为常数随机矩阵, 对状态空间 V 中染色体, $\mathbf{C}_i(k)$ 染色体经过淘汰选择后依然被选上概率为

$$\text{num}\{\mathbf{C}_i(k) \xrightarrow{\mathbf{G}} \mathbf{C}_i(k)\},$$

对于任意染色体 $\mathbf{C}_i(k) \in V$, 有

$$\mathbf{G}_{N \times N} = (g_{ij})_{N \times N},$$

$\mathbf{C}_i(k)$ 被选择中保留为 $\mathbf{C}_j(k)$ 的概率用 g_{ij} 表示, 有

$$\forall i, j, g_{ij} = \text{num}\{\mathbf{C}_i(k) \xrightarrow{\mathbf{G}} \mathbf{C}_j(k)\} > 0. \quad (20)$$

因此, \mathbf{G} 是列可容许的. 证毕.

定理 2 $\mathbf{H}, \mathbf{M}, \mathbf{G}, \mathbf{M}_1$ 是随机矩阵, \mathbf{H} 为随机矩阵, 当 \mathbf{M} 与 \mathbf{M}_1 是正随机矩阵, 且 \mathbf{G} 是列容许的随机矩阵时, $\mathbf{P}_1 = \mathbf{C} \times \mathbf{M} \times \mathbf{G} \times \mathbf{M}_1$ 是正定矩阵.

证 定义 $\mathbf{H}, \mathbf{M}, \mathbf{G}, \mathbf{M}_1$ 第 i 行 j 列元素分别为

$h_{ij}, m_{ij}, g_{ij}, m_{1ij}$, 由式(19)可知, \mathbf{M} 与 \mathbf{M}_1 是正随机矩阵, 故矩阵 $\mathbf{X} = \mathbf{H} \times \mathbf{M}$ 亦为正矩阵. 因为 \mathbf{H} 为随机矩阵, 即

$$\mathbf{H} \geq 0, \sum_{j=1}^N h_{ij} = 1, i \in [1, N]. \quad (21)$$

又由 \mathbf{M} 为正随机矩阵, 即

$$\begin{cases} m_{ij} > 0, & i, j \in [1, N], \\ \sum_{j=1}^N m_{ij} = 1, & i \in [1, N], \end{cases} \quad (22)$$

可得 \mathbf{X} 为正的, 即

$$\begin{cases} \forall i, \exists k \rightarrow h_{ik} > 0, \forall m_{kj} > 0, \\ \forall i, j, \rightarrow x_{ij} = \sum_{k=1}^N h_{ik} m_{kj} > 0. \end{cases} \quad (23)$$

针对正矩阵 \mathbf{X} 与列可容许的随机矩阵 \mathbf{G} , 有 $\mathbf{Y} = \mathbf{X} \times \mathbf{G}$, 其中 \mathbf{Y} 矩阵为正矩阵. 由式(23)可知

$$\forall i, z, x_{iz} > 0. \quad (24)$$

结合式(20)得

$$\begin{cases} \forall j, \exists z \rightarrow g_{zj} > 0, \\ \forall i, j, \rightarrow y_{ij} = \sum_{z=1}^N x_{iz} g_{zj} > 0. \end{cases} \quad (25)$$

对于正矩阵 \mathbf{Y} 与正随机矩阵 \mathbf{M}_1 , 有 $\mathbf{P}_1 = \mathbf{Y} \times \mathbf{M}_1$, 其中 \mathbf{P}_1 矩阵为正矩阵. 因为 \mathbf{Y} 是正矩阵, 即

$$\forall i, z, \rightarrow y_{iz} > 0. \quad (26)$$

又由 \mathbf{M}_1 为正随机矩阵, 即

$$m_{1ij} > 0, \#i, j \in [1, N]. \quad (27)$$

因此,

$$\begin{cases} \forall i, z, \rightarrow y_{iz} > 0, \forall z, j \rightarrow m_{1zj} > 0, \\ \forall i, j, \rightarrow p_{ij} = \sum_{z=1}^N y_{iz} m_{1zj} > 0. \end{cases} \quad (28)$$

故对于 $\mathbf{P}_1 = \mathbf{Y} \times \mathbf{M}_1 = \mathbf{H} \times \mathbf{M} \times \mathbf{G} \times \mathbf{M}_1$, 有 \mathbf{P}_1 为正定矩阵. 证毕.

性质 1 精英保留更新染色体种群会保留更优值.

证 本文的精英个体保留后状态空间基数仍然是 N , 为了方便分析, 精英个体存放在最左边位置, 精英个体的所对应的目标函数越低, 在矩阵中的位置越高, 交叉、变异、淘汰重插、微变异的转移矩阵可以写成块对角阵 $\mathbf{H}^+, \mathbf{M}^+, \mathbf{G}^+, \mathbf{M}_1^+$.

$$\mathbf{H}^+ = \text{diag}\{\mathbf{H}\}, \mathbf{M}^+ = \text{diag}\{\mathbf{M}\},$$

$$\mathbf{G}^+ = \text{diag}\{\mathbf{G}\}, \mathbf{M}_1^+ = \text{diag}\{\mathbf{M}_1\}.$$

因此, 经过遗传操作后的矩阵为

$$\hat{\mathbf{P}}_1 = \mathbf{H}^+ \mathbf{M}^+ \mathbf{G}^+ \mathbf{M}_1^+,$$

$$P_1 = \begin{bmatrix} H^+ M^+ G^+ M_1^+ & & & \\ & \ddots & & \\ & & \ddots & \\ & & & H^+ M^+ G^+ M_1^+ \end{bmatrix}, \quad (29)$$

且 $\hat{P}_1 = H^+ M^+ G^+ M_1^+ > 0$, 在遗传算法的交叉、变异、选择以及微变异后, 精英保留操作作用矩阵 U 表示. 第 k 次迭代时状态 i 对应种群中除精英个体外的最佳个体为

$$C_i(k) = \operatorname{argmin}\{O_z(k) | z = 1, \dots, n\}. \quad (30)$$

对于目标函数值为 $O_j(k)$, 如果

$$O_j(k) < O_i(k), \quad (31)$$

即精英保留时 i 状态的最佳目标函数值小于未保留 j 状态的最佳目标函数值, 若取 $U = (u_{ij})_{N \times N}$ 表示精英保留引起的状态转移矩阵, 此时, 将有 $C_i(k)$ 被 $C_j(k)$ 替换, 精英操作矩阵有

$$u_{ij} = 1, u_{iz} = 0 (z \neq j), \quad (32)$$

否则, 精英保留操作失败, 原状态保留, 此时

$$u_{ii} = 1, u_{ij} = 0, \quad (33)$$

因此, U 中每行只有一个元素为1, 其余皆为0. 精英保留操作矩阵可以由矩阵 U 可由分块矩阵 U_{rk} ($1 \leq r, k \leq N$) 表示, 有

$$U = \begin{bmatrix} U_{11} & & & \\ U_{21} & U_{22} & & \\ \vdots & \vdots & \ddots & \\ U_{N1} & U_{N2} & \dots & U_{NN} \end{bmatrix}, \quad (34)$$

其中除 U_{11} 为单位阵, 其余 U_{ii} 都是含有单位阵的对角矩阵, U_{rk} ($r < k$) 为零矩阵, 在精英保留操作后, 转移矩阵转变为 $P = \hat{P}_1 U$, 即

$$P = \begin{bmatrix} \hat{P}_1 U_{11} & & & \\ \hat{P}_1 U_{21} & \hat{P}_1 U_{22} & & \\ \vdots & \vdots & \ddots & \\ \hat{P}_1 U_{N1} & \hat{P}_1 U_{N2} & \dots & \hat{P}_1 U_{NN} \end{bmatrix}, \quad (35)$$

其中 $\hat{P}_1 U_{11}$ 是一个正随机矩阵, 当 $i \geq 2$ 时, 由于 U_{i1} 为非零阵, 因此 R 为非零矩阵, 且

$$R = \begin{pmatrix} \hat{P}_1 U_{21} \\ \vdots \\ \hat{P}_1 U_{N1} \end{pmatrix}, \quad (36)$$

同理可得, 有 T 为非零矩阵, 且

$$T = \begin{pmatrix} \hat{P}_1 U_{22} & & \\ \vdots & \ddots & \\ \hat{P}_1 U_{N2} & \dots & \hat{P}_1 U_{NN} \end{pmatrix}, \quad (37)$$

所以根据引理2, 有新的状态转移矩阵

$$P = \begin{pmatrix} C & 0 \\ R & T \end{pmatrix}, \quad (38)$$

其子矩阵 $\hat{P}_1 U_{i1}$ ($1 < i \leq N$) 可聚集成矩阵 R 中. 根据引理2, 可知 P 每次均保留更优值. 证毕.

性质2 双编码动态培育遗传聚类算法能收敛到全局最优解.

证 转移矩阵 P 中子矩阵 $\hat{P}_1 U_{11} = P_1 > 0$, 包含了精英个体(目标函数低个体)的转移概率. 由于 \hat{P}_1 是随机矩阵, 且对应式(16)中 R 不为零矩阵, 根据引理2推断出所有非全局最优解收敛概率都等于0, 因此, 处于全局最优状态的概率收敛于1, 所以由式(18), 即 $k \rightarrow \infty$ 时, $P\{Z_i = f^*\}$ 收敛于1. 证毕.

5 算法定制化配组应用验证

5.1 配组电池数据

选取浙江天能公司生产的1850 mAh方形动力电池. 筛选出满足: 电池容量1850~2030 mAh, 充电截止电压3.52~3.55 V, 满电电阻20~30 mΩ条件的电池228节. 依据式(1)将电池数据归一化, 归一化的电池数据在特征空间分布如图6所示.

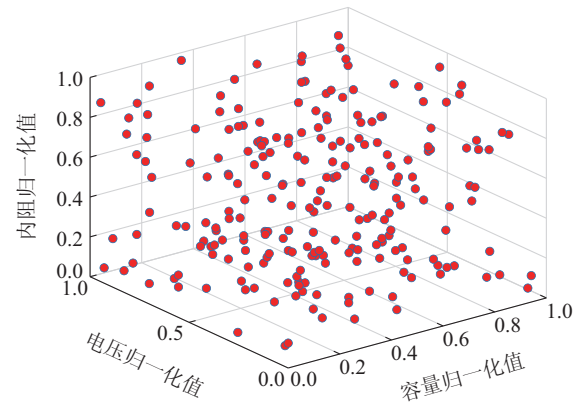


图6 228节电池样本在三维归一化的特征空间分布

Fig. 6 Distribution of 228 battery samples in three-dimensional normalized feature space

5.2 定制化配组分类指标选取

定制化配组聚类指标权重选取借助德尔菲专家打分法, 根据定制化需求, 邀请专家打分, 考虑到专家们权威程度, 对他们给出的权重值加权平均, 设定权重值集合 $W = [w_1 \dots w_m]$, 其中

$$w_j = \sum_{i=1}^{np} \{r_i \times w_{ij}\} \times 100\%. \quad (39)$$

根据每位专家的资历以及工作岗位来确定专家权威值 r_i , 且所有专家权威和为1. 专家对 m 个电池特征变量在定制化需求下的相对重要程度给与打分, 第 i 位专家对第 j 个指标的打分为 w_{ij} , 每位专家可用总分为100. 得出最终聚类指标权重值集合 W .

本文在定制化指标选取中, 联系电池生产行业领域的多位专家, 针对成组电池需在高温条件下工作^[15]

这一定制化需求,对配组指标权重进行打分,得出最后八位专家打分如表2所示.

表2 专家对3个影响指标基于定制化的打分表

Table 2 Experts score three impact indicators based on customized scoring table

专家编号/%	专家权重/%	容量占比/%	电压占比/%	电阻占比/%
1	10	65	25	10
2	13	45	30	25
3	9	35	30	35
4	17	80	15	5
5	13	55	25	20
6	11	40	30	30
7	14	70	18	12
8	13	60	25	15

根据式(39),最终得到容量、电压、内阻这3个电池特征变量在高温需求定制化的权重为

$$W = (0.5825, 0.2397, 0.1778).$$

5.3 DCDB-GCA应用参数选取

为验证本文所提算法用于电池定制化配组优越性,以图6数据为基础,了解到实际生产需要将228节电池分为3类,采用高温定制化需求设定目标函数.算法参数如表3所示.

表3 DCDB-GCA应用参数设定

Table 3 DCDB-GCA application parameter settings

参数名称	数值
筛选组数 Z	3
最大迭代次数iter	3000
种群中个体数量 n	50
电池对象数量 s	288
电池特征数量 m	3
个体编码长度 l	297
1 ~ s 编码位	ceil($Z \times \text{rand}$)
($s + 1$) ~ l 编码位	rand
交叉率acr	20
变异率mut	20
淘汰率rep_mut	10

5.4 电池定制化配组优化评价指标

结合前文容量、电压、内阻定制化的权重 W 依据式(3)构建电池定制化配组优化目标函数 $F(b, z, W, k)$,种群中各个染色体 $C_i(k)$ 适应度值计算方式与目标函数一致.第 k 次迭代的目标函数值为种群 $A(k)$ 中最优个体对应的函数值,即 k 次迭代下种群中所有个体对应的适应度函数值的最小数值,各个电池组之间的单体电池的性能指标越接近,目标函数 $F(b, z, W, k)$ 越小,因此,电池的配组优化效果值直接对应目标函数

值.

5.5 DCDB-GCA算法电池配组结果

利用本文提出的DCDB-GCA算法,获得的各时刻电池配组效果如图7所示,为避免初始随机配组效果值影响作图效果,图7从第1次迭代完开始记录.图7中的横坐标为迭代次数,纵坐标为电池配组优化效果值.由图可知目标函数值从第1次迭代82.6234降至最终的35.1714,筛选的各类样本也从最初的混乱无序的状态变为区域性显著明显的聚集形态,最终形成了3个差异性显著的电池组别,验证了所提双编码动态培育遗传聚类算法可以有效地完成电池筛选配组的工作.

5.6 微变异与精英保留遗传聚类电池配组应用优势

将文献[9]单编码遗传聚类算法(single coding genetic clustering, SC-GC)与精英保留遗传聚类算法(elitist preserving genetic clustering, EP-GC)进行分析对比.

由图8可以看出,未引入微变异与精英保留操作的SC-GC算法在前期迭代的收敛速度远低于引入微变异与精英保留后的EP-GC收敛速度,且SC-GC算法在前300次迭代过程中的效果不如EP-GC算法,改进后的EP-GC最终配组优化结果为42.7749,也优于文献[9]的SC-GC算法.因此,可以看出引入微变异与精英保留对单编码遗传聚类算法的优化效果有较显著的改进.

5.7 动态培育遗传聚类电池配组应用优势

在EP-GC基础上引入动态培育思想改进为动态培育精英保留遗传聚类算法(elitist preserving dynamic cultivation genetic clustering, EPDC-GC).

从图8可以看出,对比EP-GC以及EPDC-GC在电池配组应用方面,引入动态培育思想后,EPDC-GC算法的在前50次迭代后,目标函数的优化效果远优于EP-GC算法.

同时,从图9可以看出,EPDC-GC在1500次迭代之后调整了变异方向,EPDC-GC跳出前二分之一迭代步的局部最优点.可见,动态交叉过程对算法的前期收敛速度具有提升作用,动态调整变异方向对跳出局部最优的能力有增强作用. EPDC-GC的最终优化结果为36.2665,优于EP-GC的42.7749.

5.8 双编码遗传聚类电池配组应用优势

在前期分析的基础上,对比基于SC-GC引入精英保留、微变异以及动态培育后改进的单编码EPDC-GC算法,文献[16]中基于传统K-means算法改进的加权特征集成聚类算法(weighted feature ensemble clustering algorithm, WFE-CA)和最终引入双编码的DCDB-GCA算法.

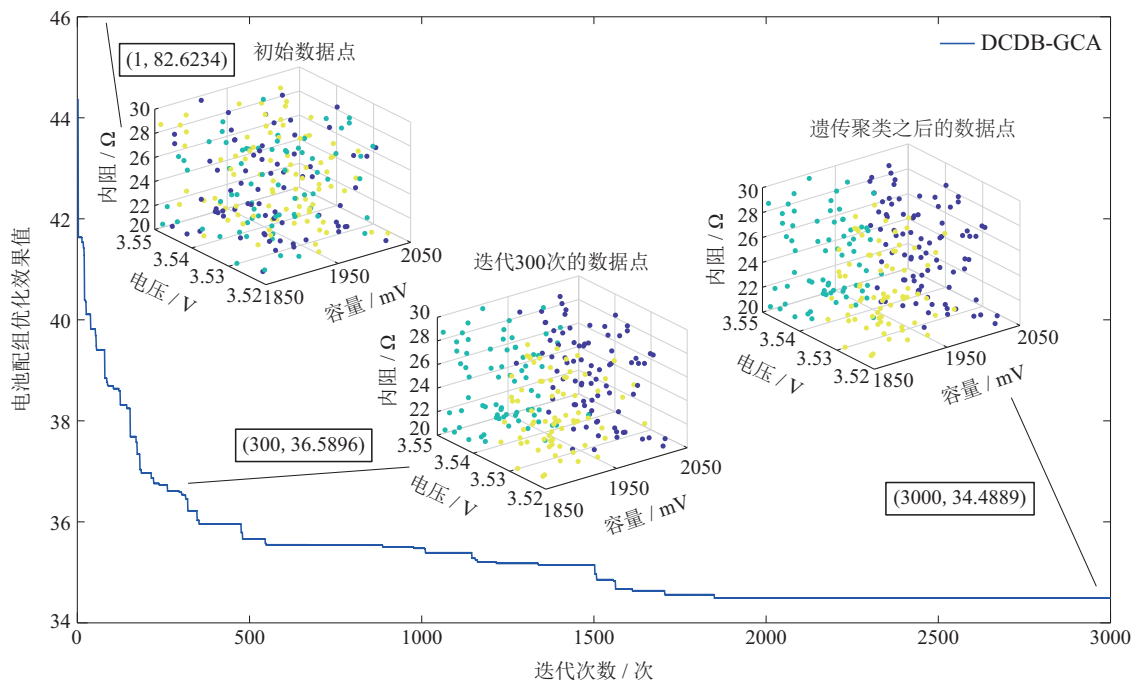


图 7 定制化迭代优化过程中的聚类效果

Fig. 7 Clustering effect in the process of customized iterative optimization

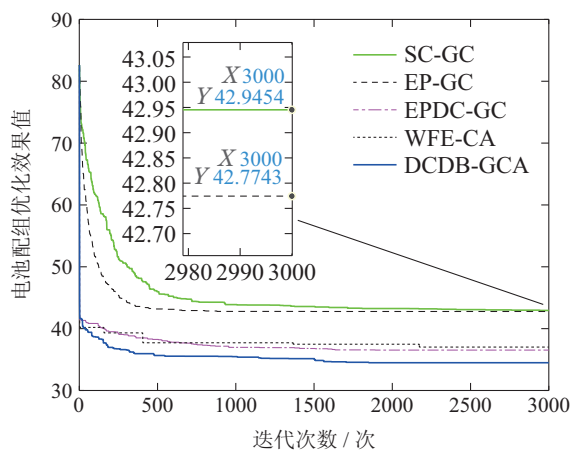


图 8 微变异与精英保留遗传聚类电池配组应用优势对比

Fig. 8 Comparison of the advantages of microvariation and elite preserved genetic clustering in battery pairing applications

优于其他4种对比算法。

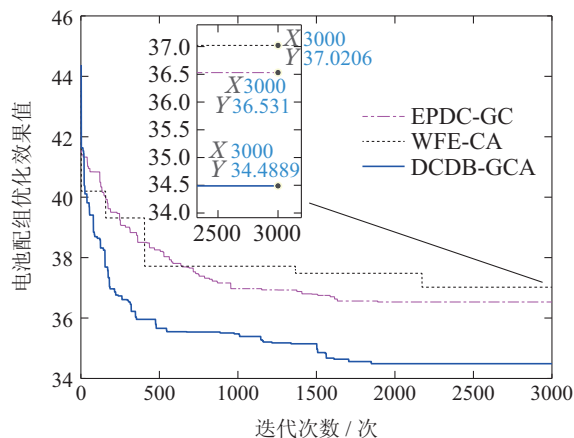


图 9 动态培育遗传聚类电池配组应用优势对比

Fig. 9 Comparison of advantages in dynamic cultivation genetic clustering battery pairing application

从图9可以看出, 引入双编码染色体编码方式后, DCDB-GCA算法在初期的收敛速度优于单编码, 并且DCDB-GCA的优化结果为35.1714, 优于EPDC-GC的36.2665以及WFE-CA的36.1289. 可见本文所提双编码方式对于遗传聚类算法的收敛速度以及收敛效果有较显著的改进.

5.9 DCDB-GCA算法优势总结

表4中记录了4种对比算法与DCDB-GCA算法于电池定制化配组应用中在300次迭代、1500次迭代、2250次迭代以及最终3000次迭代的运行情况, 可以看出, DCDB-GCA 算法在迭代前期的收敛速度, 以及迭代中后期跳出局部最优的能力和最终的配组结果都

表 4 电池定制化配组效果对比

Table 4 Comparison of customized battery grouping effects

算法名称	$k=300$	$k=1500$	$k=2250$	$k=3000$
SC-GC	53.6196	43.3698	43.1897	42.9053
EP-GC	43.5364	42.7749	42.7749	42.7749
WFE-CA	39.2914	36.2665	36.2665	36.2665
EPDC-GC	37.5329	36.2665	36.1289	36.1289
DCDB-GCA	36.6691	35.5869	35.192	35.1714

6 结论

本文提出了一类基于双编码动态培育的遗传聚类

算法,同时编码聚类中心与聚类类别,动态调整交叉对象与变异方向,改变淘汰重插方式,加入微变异操作,确保算法前期寻找到优化方向从而快速收敛,以及后期跳出局部最优的可能性.本文同时也分析了所提算法的收敛性,并通过电池定制化配组工程案例,验证了所提双编码动态培育的遗传聚类算法在实际分类应用的可行性和优越性.本文所提算法可进一步推广应用于机械加工、物料生产等制造业的定制化产品单体组合领域.

参考文献:

- [1] ZHANG Yong, YANG Kang, HAO Guosheng, et al. An evolutionary optimization framework based on transfer learning of similar historical information. *Acta Automatica Sinica*, 2021, 47(3): 652 – 665.
(张勇, 杨康, 郝国生, 等. 基于相似历史信息迁移学习的进化优化框架. 自动化学报, 2021, 47(3): 652 – 665.)
- [2] WANG Jie, WANG Yan. Quality control method based on quantum genetic clustering algorithm. *Journal of System Simulation*, 2019, 31(12): 2591 – 2599.
(王杰, 王艳. 基于量子遗传聚类算法的质量控制方法. 系统仿真学报, 2019, 31(12): 2591 – 2599.)
- [3] GUNJAN, SHARMA A K, VERMA K. GA-UCR: Genetic algorithm based unequal clustering and routing protocol for wireless sensor networks. *Wireless Personal Communications*, 2022, 128(1): 537 – 558.
- [4] BRITO J A D, FADEL A C, SEMAAN G S, et al. Heuristics applied to minimization of the maximum-diameter clustering problem. *IEEE Latin America Transactions*, 2021, 19(4): 652 – 659.
- [5] DONG Zhaoxin, HUA Xiang, JIANG Bingqing, et al. An improved genetic clustering topological clustering algorithm. *Journal of Xi'an University of Technology*, 2019, 39(1): 93 – 98.
(董兆鑫, 华翔, 姜冰清, 等. 一种改进的遗传聚类拓扑分簇算法. 西安工业大学学报, 2019, 39(1): 93 – 98.)
- [6] SONG Feibao, JIA Ruiyu. Elite genetic K-medoids clustering algorithm. *Computer Engineering and Applications*, 2018, 54(22): 144 – 149.
(宋飞豹, 贾瑞玉. 精英遗传K-medoids聚类算法. 计算机工程与应用, 2018, 54(22): 144 – 149.)
- [7] LI Yue, MU Weisong, CHU Xiaoquan, et al. K-means clustering algorithm based on improved quantum particle swarm optimization and its application. *Control and Decision*, 2022, 37(4): 839 – 850.
(李月, 穆维松, 褚晓泉, 等. 基于改进量子粒子群的K-means聚类算法及其应用. 控制与决策, 2022, 37(4): 839 – 850.)
- [8] ISMKHAN H. Black box optimization using evolutionary algorithm with novel selection and replacement strategies based on similarity between solutions. *Applied Soft Computing*, 2017, 64(3): 260 – 271.
- [9] MA Suliang, LI Jianlin, LI Yaxin, et al. Customized clustering optimization method for battery reutilization screening requirements. *Proceedings of the CSEE*, 2022, 42(17): 6208 – 6220.
(马速良, 李建林, 李雅欣, 等. 面向电池梯次利用筛选需求的定制化聚类优化方法. 中国电机工程学报, 2022, 42(17): 6208 – 6220.)
- [10] ZHANG Chaolong, ZHAO Shaishai, ZHANG Bo. A fast sorting method for retired power batteries based on factor analysis and K-means clustering. *Power System Protection and Control*, 2021, 49(12): 41 – 47.
(张朝龙, 赵筛筛, 章博. 基于因子分析与K-means聚类的退役动力电池快速分选方法. 电力系统保护与控制, 2021, 49(12): 41 – 47.)
- [11] QI Zhan, LI Maojun, MO Hong, et al. Improved state space model genetic algorithm and its global convergence analysis. *Control Theory & Applications*, 2020, 37(10): 2115 – 2122.
(齐战, 李茂军, 莫红, 等. 改进的状态空间模型遗传算法及其全局收敛性分析. 控制理论与应用, 2020, 37(10): 2115 – 2122.)
- [12] DING Lichao, HUANG Feng, PAN Wei. Artillery fire allocation method based on improved chaotic genetic algorithm. *System Simulation Technology*, 2021, 17(1): 12 – 16.
(丁立超, 黄枫, 潘伟. 基于改进混沌遗传算法的炮兵火力分配方法. 系统仿真技术, 2021, 17(1): 12 – 16.)
- [13] PAN Wei, DING Lichao, HUANG Feng, et al. Adaptive genetic algorithm based on chaotic “micro mutation”. *Control and Decision*, 2021, 36(8): 2042 – 2048.
(潘伟, 丁立超, 黄枫, 等. 基于混沌“微变异”自适应遗传算法. 控制与决策, 2021, 36(8): 2042 – 2048.)
- [14] PENG Qian, WANG Aijuan, LI Junyang, et al. Research on power grid demand side dispatching optimization based on efficient genetic algorithm and its convergence analysis. *Power System Protection and Control*, 2022, 50(6): 33 – 42.
(彭茜, 王爱娟, 李峻阳, 等. 基于高效遗传算法的电网需求侧调度优化研究及其收敛性分析. 电力系统保护与控制, 2022, 50(6): 33 – 42.)
- [15] XIE Yucen, ZOU Jianxiao, PENG Chao. Temperature control of proton exchange membrane fuel cell based on variable universe fuzzy increment theory. *Control Theory & Applications*, 2019, 36(3): 428 – 435.
(谢雨岑, 邹见效, 彭超. 基于变论域模糊增量理论的质子交换膜燃料电池温度控制. 控制理论与应用, 2019, 36(3): 428 – 435.)
- [16] BAI Liang, GAO Jin. A weighted clustering ensemble algorithm based on clustering criterion fusion. *Journal of Shanxi University (Natural Science Edition)*, 2018, 41(2): 302 – 307.
(白亮, 高锦. 基于聚类准则融合的加权聚类集成算法. 山西大学学报(自然科学版), 2018, 41(2): 302 – 307.)

作者简介:

王子赞 副教授, 硕士生导师, 目前研究方向为高端电池智能制造, E-mail: wangzy0601@163.com;

史伟杰 硕士研究生, 目前研究方向为动力电池智能配组技术, E-mail: 15358096095@163.com;

王艳 教授, 博士生导师, 目前研究方向为生产过程建模与优化控制, E-mail: yanwang@jiangnan.edu.cn;

纪志成 教授, 博士生导师, 目前研究方向为工业过程建模与状态估计, E-mail: zcji@jiangnan.edu.cn.