

# 控制与决策

Control and Decision

## 基于Tent混沌和透镜成像学习策略的平衡优化器算法

周鹏, 董朝轶, 陈晓艳, 赵肖懿, 王启来

引用本文:

周鹏,董朝轶,陈晓艳,赵肖懿,王启来. 基于Tent混沌和透镜成像学习策略的平衡优化器算法[J]. 控制与决策, 2023, 38(6): 1569–1576.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2021.1537>

---

### 您可能感兴趣的其他文章

#### Articles you may be interested in

##### [基于改进蚁群和DWA算法的机器人动态路径规划](#)

Robot dynamic path planning based on improved ant colony and DWA algorithm  
控制与决策. 2022, 37(9): 2211–2216 <https://doi.org/10.13195/j.kzyjc.2021.1804>

##### [基于改进双层蚁群算法的移动机器人路径规划](#)

Mobile robot path planning using improved double-layer ant colony algorithm  
控制与决策. 2022, 37(2): 303–313 <https://doi.org/10.13195/j.kzyjc.2020.0610>

##### [多策略融合的改进麻雀搜索算法及其应用](#)

Improved sparrow search algorithm with multi-strategy integration and its application  
控制与决策. 2022, 37(1): 87–96 <https://doi.org/10.13195/j.kzyjc.2021.0582>

##### [嵌入Circle映射和逐维小孔成像反向学习的鲸鱼优化算法](#)

Whale optimization algorithm for embedded Circle mapping and one-dimensional oppositional learning based small hole imaging  
控制与决策. 2021, 36(5): 1173–1180 <https://doi.org/10.13195/j.kzyjc.2019.1362>

##### [基于自适应正态云模型的灰狼优化算法](#)

Grey wolf optimization algorithm based on adaptive normal cloud model  
控制与决策. 2021, 36(10): 2562–2568 <https://doi.org/10.13195/j.kzyjc.2020.0233>

# 基于Tent混沌和透镜成像学习策略的平衡优化器算法

周 鹏<sup>1,2</sup>, 董朝轶<sup>1,2†</sup>, 陈晓艳<sup>1,2</sup>, 赵肖懿<sup>1,2</sup>, 王启来<sup>1,2</sup>

(1. 内蒙古工业大学 电力学院, 呼和浩特 010080;

2. 内蒙古自治区高校智慧能源技术与装备工程研究中心, 呼和浩特 010080)

**摘要:** 针对平衡优化器算法存在收敛速度慢且容易陷入局部最优的问题, 提出一种改进的平衡优化器算法. 引入 Tent 混沌映射初始化种群提高迭代前期的收敛速度, 通过透镜成像学习策略避免迭代后期陷入局部最优. 选取 12 个通用的标准测试函数进行仿真实验, 并与多个智能群优化算法进行对比, 通过实验验证改进后算法寻优性能的优越性. 最后, 将改进后的算法应用于移动机器人路径规划任务, 结果表明: 相比较原算法, 改进后的算法不但具有较高的搜索效率, 而且能够搜索到更短的安全路径.

**关键词:** 平衡优化器算法; Tent 混沌映射; 透镜成像学习策略; 移动机器人; 路径规划

中图分类号: TP301.6

文献标志码: A

DOI: 10.13195/j.kzyjc.2021.1537

引用格式: 周鹏, 董朝轶, 陈晓艳, 等. 基于 Tent 混沌和透镜成像学习策略的平衡优化器算法[J]. 控制与决策, 2023, 38(6): 1569-1576.

## An equilibrium optimizer algorithm based on a Tent chaos and lens imaging learning strategy

ZHOU Peng<sup>1,2</sup>, DONG Chao-yi<sup>1,2†</sup>, CHEN Xiao-yan<sup>1,2</sup>, ZHAO Xiao-yi<sup>1,2</sup>, WANG Qi-lai<sup>1,2</sup>

(1. Electric Power College, Inner Mongolia University of Technology, Hohhot 010080, China; 2. Intelligent Energy Technology and Equipment Engineering Research Center of Colleges and Universities in Inner Mongolia Autonomous Region, Hohhot 010080, China)

**Abstract:** To solve the problems of local optimization and slow convergence in the process of optimization and iteration, the paper proposes an improved optimization algorithm for the equilibrium optimizer, introduces the Tent chaotic map to initialize the population to improve the convergence speed in the early stage of the iteration, and uses the lens imaging learning strategy to avoid falling into the local optimum in the later stage of the iteration. Twelve general standard test functions are selected for simulation experiments to compare with multiple intelligent group optimization algorithms. The experimental results verify the superiority of the improved algorithm in optimization performance. Finally, the improved algorithm is applied to the path planning task of mobile robots. Compared with the equilibrium optimizer algorithm, the improved algorithm not only has high search efficiency, but also can search shorter secure paths.

**Keywords:** equilibrium optimizer algorithm; Tent chaotic mapping; lens imaging learning strategy; mobile robot; path planning

## 0 引言

平衡优化器(equilibrium optimizer, EO)算法是由美国学者 Faramarzi 等<sup>[1]</sup>在 2020 年提出的一种新的元启发式算法, 其灵感来源于控制固定体积上的一个混合动态质量平衡方程, 质量平衡方程描述了一个质量在控制体积内的进入、离开和产生过程中守恒的基础物理现象. EO 算法原理简单且易编程实现, 近期在

电力系统<sup>[2]</sup>、新能源<sup>[3]</sup>、微电子<sup>[4]</sup>等领域有着成功的应用.

在前期的研究中, 研究者们发现: 相较于传统的元启发式算法, 如樽海鞘(SSA)算法<sup>[5]</sup>、蚁狮(ALO)算法<sup>[6]</sup>、粒子群(PSO)算法<sup>[7]</sup>等, EO 算法在基础性性能测试上具有求解精度高、全局探索能力强等优势, 但在算法中初始化种群多样性差, 粒子种群位置更新方式

收稿日期: 2021-08-31; 录用日期: 2022-03-15.

基金项目: 国家自然科学基金项目(61863029, 61364018); 内蒙古科技成果转化项目(CGZH2018129); 内蒙古自治区科技计划项目(2021GG0264, 2020GG0268); 内蒙古自然科学基金项目(2020MS06020, 2016JQ07, 2021MS06017).

†通讯作者. E-mail: dongchaoyi@hotmail.com.

始终依赖于平衡状态方程,使得迭代过程中也存在收敛速度慢和易陷入局部最优等缺点. 针对上述问题,多位学者提出了一些改进的方法. 从改进更新方程角度: Wunnava等<sup>[8]</sup>引入了自适应决策修改浓度向量更新方程,通过增加种群的多样性提升全局搜索的能力; Abder-Basset等<sup>[9]</sup>引入高斯变异策略控制浓度向量个体的搜索范围,使种群以最优解为中心向四周呈放射状搜索,增强了全局探索能力. 从引入新算子角度: Abdel-Basset等<sup>[10]</sup>引入了局部最小值消除策略修改初始化方程,通过增加随机算子解扩大搜索范围,防止陷入局部最优解; Tang等<sup>[11]</sup>引入高斯分布估计策略,利用群体优势信息引导群体演化,提高了算法的性能; Kardani等<sup>[12]</sup>采用极限学习机与人工神经网络结合的突变机制策略增加种群的多样性,提升了对致密碳酸盐岩渗透率预测的准确性.

虽然上述改进取得了不错的算法提升效果,但EO的寻优精度和收敛速度仍有待提高. 以上对EO算法的某一缺陷的改进较单一,而本文针对EO算法的缺陷进行了较为全面的分析和提升,针对前期收敛速度慢和后期易陷入局部最优的问题,提出一种基于Tent混沌和透镜成像学习策略的平衡优化器算法(Tent chaos and lens imaging learning strategy equilibrium optimizer algorithms, TLIL-EO). 本文主要贡献总结如下:

- 1) 算法迭代前期,采用Tent混沌映射初始化种群,增强种群的均匀性和遍历性,提高前期收敛速度.
- 2) 算法迭代后期,引入透镜成像学习策略增加在极值点附近的扰动,从而提升跳出局部最优的能力.
- 3) 通过经典函数测试和3次样条插值或B样条结合的移动机器人路径规划任务验证TLIL-EO算法的有效性,相比较于其他算法,TLIL-EO算法在搜索效率和精度方面具有显著的优势.

## 1 平衡优化器算法

EO算法是一种基于物理概念的元启发式算法,试图模拟质量在控制体积上的动态平衡行为. EO算法使用一组个体(粒子)模拟质量在体积上的浓度向量,其中每个浓度向量代表优化问题的一个解. EO算法与其他元启发式算法一样,首先在优化问题的搜索空间内随机初始化构造初始浓度,其定义<sup>[1]</sup>如下:

$$X_i^d = X_{\min}^d + \text{rand}_i^d (X_{\max}^d - X_{\min}^d), \quad (1)$$

$$i = 1, 2, \dots, N, \quad d = 1, 2, \dots, D.$$

其中:  $X_i$  为第  $i$  个粒子的位置向量,  $X_i^d$  为第  $i$  个粒子的第  $d$  维变量,  $X_{\max}^d$  和  $X_{\min}^d$  为合理搜索空间内  $d$  维变量的上限和下限,  $\text{rand}_i^d$  为  $[0, 1]$  内随机数,  $N$  为粒

子的总数,  $D$  为维度. 生成初始种群后,用一个特定的适应度函数对粒子进行评估,识别出平衡候选粒子.

在EO算法的优化过程中,使用一个平衡状态池存储迄今为止4个最佳的平衡候选粒子及其平均值,可以帮助EO算法开发接近最佳解决方案的区域,从而找到全局最优,平衡状态池为

$$X_{\text{eq, pool}} = \{X_{\text{eq}(1)}, X_{\text{eq}(2)}, X_{\text{eq}(3)}, X_{\text{eq}(4)}, X_{\text{eq(ave)}}\}. \quad (2)$$

其中:  $X_{\text{eq}(1)} \sim X_{\text{eq}(4)}$  为最好的平衡候选粒子,  $X_{\text{eq(ave)}}$  为前4个候选粒子的平均值. 在每次迭代中,每个粒子在这5个候选粒子中以相同的概率随机选择1个候选粒子,定义为候选者  $X_{\text{eq}}$ . 参考文献[1]引入了指数项系数  $F$  来平衡EO算法的全局探索能力与局部开发能力,其定义如下:

$$F = \exp(-\lambda(t - t_0)). \quad (3)$$

其中:  $\lambda$  为  $[0, 1]$  之间的  $D$  维随机向量; 时间  $t$  被定义为迭代次数的函数,且随迭代次数的增加而衰减,即

$$t = \left(1 - \frac{\text{Iter}}{\text{MaxIter}}\right)^{(\omega_2 \frac{\text{Iter}}{\text{MaxIter}})}. \quad (4)$$

这里:  $\text{Iter}$  为当前迭代次数,  $\text{MaxIter}$  为定义的最大迭代次数;  $\omega_2$  为局部探索的权重系数,  $\omega_2$  越大其局部探索能力越强. 另一方面,引入参数  $t_0$  用于调节全局开发能力和局部探索能力,其定义如下:

$$t_0 = \frac{1}{\lambda} \circ \ln(-\omega_1 \text{sign}(r - 0.5) \circ [1 - e^{-\lambda t}]) + t. \quad (5)$$

其中: 运算  $\circ$  为 Hadamard 积;  $\omega_1$  为全局探索的权重系数,  $\omega_1$  越大其全局开发能力越强;  $r$  为  $[0, 1]$  之间的  $D$  维随机向量. 将式(4)和(5)代入(3)中,指数项系数  $F$  可重新定义为

$$F = \omega_1 \text{sign}(r - 0.5) \circ [e^{-\lambda t} - 1]. \quad (6)$$

引入质量生成速率  $G$ , 用于加强算法的局部寻优能力,其定义如下:

$$G = G_0 \circ \exp(-\lambda(t - t_0)) = G_0 \circ F. \quad (7)$$

$$G_0 = \text{GCP} \circ (X_{\text{eq}} - \lambda \circ X_i). \quad (8)$$

$$\text{GCP} = \begin{cases} 0.5r_1, & r_2 \geq \text{GP}; \\ 0, & r_2 < \text{GP}. \end{cases} \quad (9)$$

其中:  $r_1$  和  $r_2$  分别为  $[0, 1]$  之间的两个  $D$  维随机向量;  $\text{GCP}$  为质量生成速率控制参数;  $\text{GP}$  为生成概率,取0.5时,全局探索能力和局部开发能力达到平衡. 最后,EO按照如下规则进行更新:

$$X_i = X_{\text{eq}} + (X_i - X_{\text{eq}}) \circ F + \frac{G}{\lambda V} \circ (1 - F). \quad (10)$$

其中:  $F$  为式(6)中的指数项,  $G$  为式(7)中的质量生成速率,  $X_{\text{eq}}$  为平衡池中的候选者,  $V$  为一个常数单位. 需注意,式(5)~(10)的除法运算和对数运算亦为按元素除和取对数.

## 2 基于Tent混沌和透镜成像学习策略的平衡优化器算法

### 2.1 Tent混沌映射序列

混沌映射具有随机性、遍历性和有序性等特点,可用于增加种群的多样性,加快算法前期的收敛速度.因此,本文选用Tent混沌产生的序列初始化种群改进EO算法,表达式<sup>[13]</sup>如下:

$$Y_i^d = \begin{cases} 2Y_i^d, & 0 \leq Y_i^d \leq 0.5; \\ 2(1 - Y_i^d), & 0.5 \leq Y_i^d \leq 1. \end{cases} \quad (11)$$

其中:  $i = 1, 2, \dots, N$  对应粒子序号,  $d = 1, 2, \dots, D$  为变量维度.可行域内产生Tent映射序列过程中可能存在小周期  $\{0.2, 0.4, 0.6, 0.8\}$ , 也存在不稳定点  $\{0, 0.25, 0.5, 0.75\}$ .为了避免迭代中陷入小周期和不稳定周期点,在上述表达式中引入随机变量对序列进行扰动,在可控的随机范围内,也保证了Tent映射序列的随机性、遍历性和有序性.

$$Y_i^d = \begin{cases} 2Y_i^d + \text{rand}(0, 1) \times \frac{1}{N}, & 0 \leq Y_i^d \leq 0.5; \\ 2(1 - Y_i^d) + \text{rand}(0, 1) \times \frac{1}{N}, & 0.5 \leq Y_i^d \leq 1. \end{cases} \quad (12)$$

通过式(12)取得Tent映射初始值后,代入式(1)可产生基于Tent混沌序列的初始化种群,可表示为

$$X_i^d = X_{\min}^d + Y_i^d(X_{\max}^d - X_{\min}^d), \quad i = 1, 2, \dots, N, \quad d = 1, 2, \dots, D. \quad (13)$$

### 2.2 透镜成像反学习策略

为提高EO算法跳出局部最优的能力,引入一种基于镜头成像原理的反学习策略,基于如下假设:对于解空间的可行解  $X$ ,总存在与之相对应的反向解  $X^*$ ,若反向解  $X^*$  的适应度值优于可行解  $X$ ,则反向解  $X^*$  更新为最优解.

**定义1** 反向点<sup>[14]</sup>: 假设第  $i$  个粒子位置  $X_i = [X_i^1, X_i^2, \dots, X_i^D]^T$  为  $D$  维空间中的一个点,且  $X_i^d \in [a_d, b_d], d \in 1, 2, \dots, D$ , 则  $X_i$  的反向点表示为  $X_i^* = [X_i^{1*}, X_i^{2*}, \dots, X_i^{D*}]^T$ , 其中  $X_i^{d*} = a_d + b_d - X_i^d$ .

**定义2** 基点<sup>[15]</sup>: 若  $D$  维空间中存在若干点  $o_1, o_2, \dots, o_M$ , 对于任意一点  $X_i = [X_i^1, X_i^2, \dots, X_i^D]^T$ , 其反向点  $X_i^* = [X_i^{1*}, X_i^{2*}, \dots, X_i^{D*}]^T$  到  $o_m (m = 1, 2, \dots, M)$  的欧氏距离为  $l_m$  和  $l'_m$ , 令缩放因子  $k = l_m/l'_m$ , 且  $k = 1, 2, \dots, n$ , 则  $o_m$  称为  $X$  和  $X^*$  在  $k = m$  时的基点.

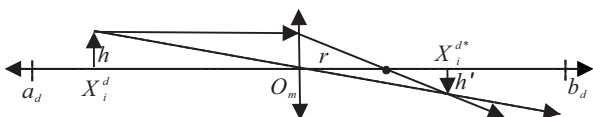


图1 透镜成像反学习策略

粒子在解空间中求取反向解的过程可以看作透镜成像的过程,如图1所示,以  $D$  维空间为例,假设在区间  $[a_d, b_d]$  上,  $X_i^d$  处有一个高为  $h$  的物体,基点位置  $o_m$  处(本文取基点为区间  $[a_d, b_d]$  的中点)放置焦距为  $r$  的凸透镜,另一侧通过透镜成像得到一个高度为  $h'$  的影像,坐标投影的位置为  $X_i^{d*}$ ,其成像原理表达为

$$\frac{\frac{(a_d + b_d)}{2} - X_i^d}{X_i^{d*} - \frac{(a_d + b_d)}{2}} = \frac{h}{h'}, \quad (14)$$

$$\frac{r}{X_i^{d*} - \frac{(a_d + b_d)}{2} - r} = \frac{h}{h'}. \quad (15)$$

利用相似三角形原理,缩放因子  $k = l_m/l'_m = h/h'$ , 式(14)和(15)变换为求解反向解  $X_i^{d*}$ , 即

$$X_i^{d*} = \frac{(a_d + b_d)}{2} + \frac{(a_d + b_d)}{2k} - \frac{X_i^d}{k}, \quad (16)$$

$$X_i^{d*} = (1 + 1/k)r + \frac{(a_d + b_d)}{2}. \quad (17)$$

在式(17)中,假定基点的位置和焦距  $r$  都是固定的,缩放因子  $k$  越大,反向解越接近基点  $o_m$ , 同样也越接近可行解,故缩放因子  $k$  被称为微观调控因子,只在可行解周边小范围内搜索,增加了种群的多样性,通过调节缩放因子  $k$  提升EO算法跳出局部最优的能力.

### 2.3 改进的EO算法(TLIL-EO)具体步骤

改进的EO算法(TLIL-EO)具体步骤如下:

step 1: 设置算法初始化输入参数:粒子总数  $N$ , 最大迭代次数  $\text{MaxIter}$ , 权重系数  $\omega_1, \omega_2$ , 常数单位  $V$ , 区间常数  $\text{GP}$ ;

step 2: 引入Tent混沌产生的序列初始化  $N$  个粒子种群;

step 3: 计算适应度值:计算每个粒子个体的适应度值,记录并筛选出迄今为止4个最佳的平衡候选粒子;

step 4: 构建平衡状态池并选定候选者:计算出平衡候选粒子的平均值  $X_{\text{eq(ave)}}$ , 根据式(2)组成平衡状态池,在这5个候选粒子中以相同的概率随机选择1个候选粒子作为候选者  $X_{\text{eq}}$ ;

step 5: 引入透镜成像反向学习策略更新最优解:根据式(16)对  $X$  引入透镜成像反向学习策略,求取反向解  $X_*$ , 若适应值  $f(X_i) < f(X_i^*)$ , 则反向解  $X_i^*$  代替  $X_i$  加入粒子种群,记录最优粒子浓度;

step 6: 根据式(6)求解指数项系数  $F$ 、式(7)求解质量生成速率  $G$ 、式(10)进一步更新粒子浓度;

step 7: 若当前迭代次数小于最大迭代次数,则重复step 5和step 6的迭代过程,直至达到设定的精度要求或最大迭代次数,输出最优个体位置及其适应度值.

算法流程如图2所示。

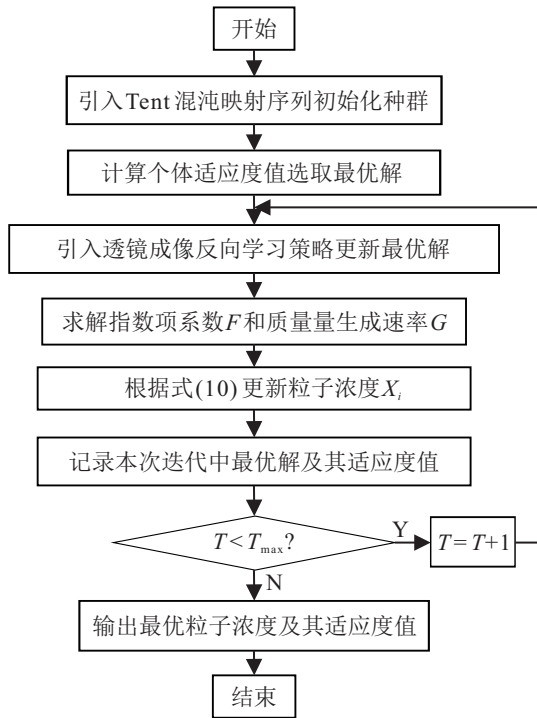


图2 TLIL-EO算法流程

### 3 基准测试函数仿真实验结果与分析

#### 3.1 基准测试函数

为验证提出的TLIL-EO算法的性能,本文选取合计12个单峰、多峰Benchmark基准函数进行算法寻优测试<sup>[16]</sup>,将TLIL-EO算法与传统EO算法、樽海鞘群算法(SSA)<sup>[17]</sup>、粒子群优化算法(PSO)<sup>[7]</sup>、飞蛾扑火优化算法(MFO)<sup>[18]</sup>、蜻蜓优化算法(DA)<sup>[19]</sup>、多神经优化算法(MVO)<sup>[20]</sup>和蚁狮优化算法(ALO)<sup>[16]</sup>进行函数测试比较。

#### 3.2 TLIL-EO算法性能分析

##### 3.2.1 基准函数寻优性能分析

测试环境与参数设置:测试环境采用微软64位操作系统,所有的算法性能检验均采用Matlab编码实现;种群规模设置为100,最大迭代次数设置为1000,重复运算50次,取最优收敛值、平均值和标准差见表1。为进一步验证改进后EO算法的性能优势,对表1中的实验结果进行无重复双因素分析和Friedman秩和验证<sup>[2]</sup>,其显著性水平值设定为5%,验证分析结果见表2和表3。

表1 验证结果

函数值		TLIL-EO	EO	SSA	MVO	ALO	DA	PSO
$f_1$	最佳值	0	2.20 e-126	1.95 e-10	0.000 34	2.70 e-10	0.136 2	6.12 e-18
	均值	0	1.4 e-122	1.83 e-10	1.16 e-03	8.41 e-10	4.494	7.44 e-12
	方差	0	1.3 e-243	1.87 e-13	2.65 e-07	8.30 e-20	119.35	8.36 e-28
$f_2$	最佳值	<b>2.24 e-242</b>	3.13 e-69	3.91 e-05	0.108 9	0.000 89	0.004 2	1.49 e-09
	均值	<b>2.31 e-236</b>	2.19 e-67	0.168 7	0.193 1	22.237	10.223	1.62 e-07
	方差	0	6.53 e-134	0.088 4	0.002	1 220.27	31.225	8.79 e-14
$f_3$	最佳值	0	9.84 e-44	0.019 1	3.058 9	4.057 4	0	0.801 8
	均值	0	2.02 e-37	1.026 5	7.096 4	34.873	4.20e+03	2.516
	方差	0	1.25 e-72	1.876 7	8.994 3	622.13	2.60e+07	1.286 3
$f_4$	最佳值	<b>1.76 e-199</b>	3.76 e-35	0.006 9	0.016 62	0.667	0	0.08
	均值	<b>1.92 e-193</b>	6.55 e-31	0.004 9	0.376	4.934 8	6.519 6	0.204 1
	方差	0	7.45 e-60	0.766	0.018 6	6.029 2	27.112	0.006
$f_5$	最佳值	21.047 5	22.754 1	<b>0.055</b>	26.313	13.312	28.734	<b>0.571</b>
	均值	21.603 2	23.203 3	27.679	215.81	136.1	7.01e+03	41.76
	方差	0.080 851	0.037 928	6 585.9	1.47e+07	3.54e+05	2.26e+08	1 049.2
$f_6$	最佳值	6.83 e-27	2.18 e-24	4.29 e-09	0.055 3	3.24 e-08	1.252 9	69.52
	均值	5.25 e-23	1.25 e-20	7.23 e-09	0.119 8	7.86 e-08	232.26	110.62
	方差	1.54 e-44	4.38 e-39	2.24 e-18	0.001 48	2.21 e-15	1.44e+05	215.46
$f_7$	最佳值	<b>1.54 e-06</b>	6.06 e-05	0.010 2	0.000 24	0.010 2	3.59 e-05	0.011 7
	均值	2.54 e-05	2.20 e-04	0.026 3	0.006 9	0.022 6	0.065 4	0.032 22
	方差	2.85 e-10	1.34 e-08	9.82 e-05	5.40 e-06	5.06 e-05	0.014 81	0.000 01
$f_8$	最佳值	0	0	11.939	46.833	30.843	0.002 3	17.295
	均值	0	0	35.042	100.16	64.195	51.005	32.312
	方差	0	6.32 e-60	166.8	584.17	433.07	2 049.7	84.286
$f_9$	最佳值	<b>8.88 e-16</b>	4.44 e-15	1.70 e-05	0.066 7	4.70 e-05	0.001 1	9.19 e-10
	均值	2.88 e-15	4.44 e-15	1.154 2	0.419 9	1.808 3	4.559 7	3.30 e-09
	方差	3.17 e-30	0	0.740 9	0.291 7	0.416 3	7.109 3	2.78 e-15
$f_{10}$	最佳值	0	0	1.48 e-08	0.167 8	6.06 e-06	0	0
	均值	0	0	0.01	0.286 3	0.010 4	5.438 5	0.009 7
	方差	0	0	0.000 1	0.005	0.000 1	168.64	0.000 1
$f_{11}$	最佳值	1.71 e-28	1.50 e-25	2.21 e-11	0.000 4	0.675	0.146 5	<b>2.04 e-20</b>
	均值	<b>1.80 e-24</b>	1.80 e-22	2.316 9	0.721 5	6.515 8	26.296	<b>0.002</b>
	方差	<b>3.62 e-47</b>	2.10 e-43	3.603	0.89	12.627	2.27e+04	<b>0.000 2</b>
$f_{12}$	最佳值	<b>0.000 3</b>	<b>0.000 3</b>	0.000 3	0.000 3	0.000 3	0.000 31	0.000 3
	均值	0.000 74	0.001 58	0.000 7	0.000 42	0.001 9	0.001 4	0.000 71
	方差	8.05 e-06	2.30 e-05	1.03 e-07	5.81 e-05	2.22 e-05	2.66 e-07	2.77 e-08

表2 无重复双因素分析结果

$F$	$P$ -value	Fcrit
0.7394	0.6198	2.2395

表3 Friedman秩和验证结果

算法	秩均值
EO	2.54
SSA	3.83
MVO	5.71
ALO	5.33
DA	4.63
PSO	4.17

测试结果如表1所示,表1中的最佳值、平均值都可以反映算法的收敛精度和寻优能力,方差可反映算法的稳定性.对于单峰和多峰测试函数,TLIT-EO算法在绝大多数函数的计算性能方面均优于其他算

法.相比较于EO算法,TLIT-EO算法在收敛精度、稳定性方面有了明显的提升,进一步验证了其在全局探索和跳出局部最优解能力上的优势.图3给出了12个基准测试函数的收敛图,相比于其他算法,TLIT-EO算法在迭代前期收敛速度明显更快,中后期收敛的精度更高.这是由于引入Tent混沌映射序列初始化种群,增加种群多样性进而加快了寻优效率,透镜成像反学习策略在中后期增强了跳出局部最优的能力并持续全局探索,达到或接近较高的收敛精度.

对实验结果进行无重复双因素分析如表2所示,因为 $P$ -value > 0.05,且 $F < F_{crit}$ ,所以7种算法在显著性水平值下无明显差异.Friedman秩和验证中TLIT-EO算法的秩均值最小,进一步验证了其算法性能优于其他智能算法.

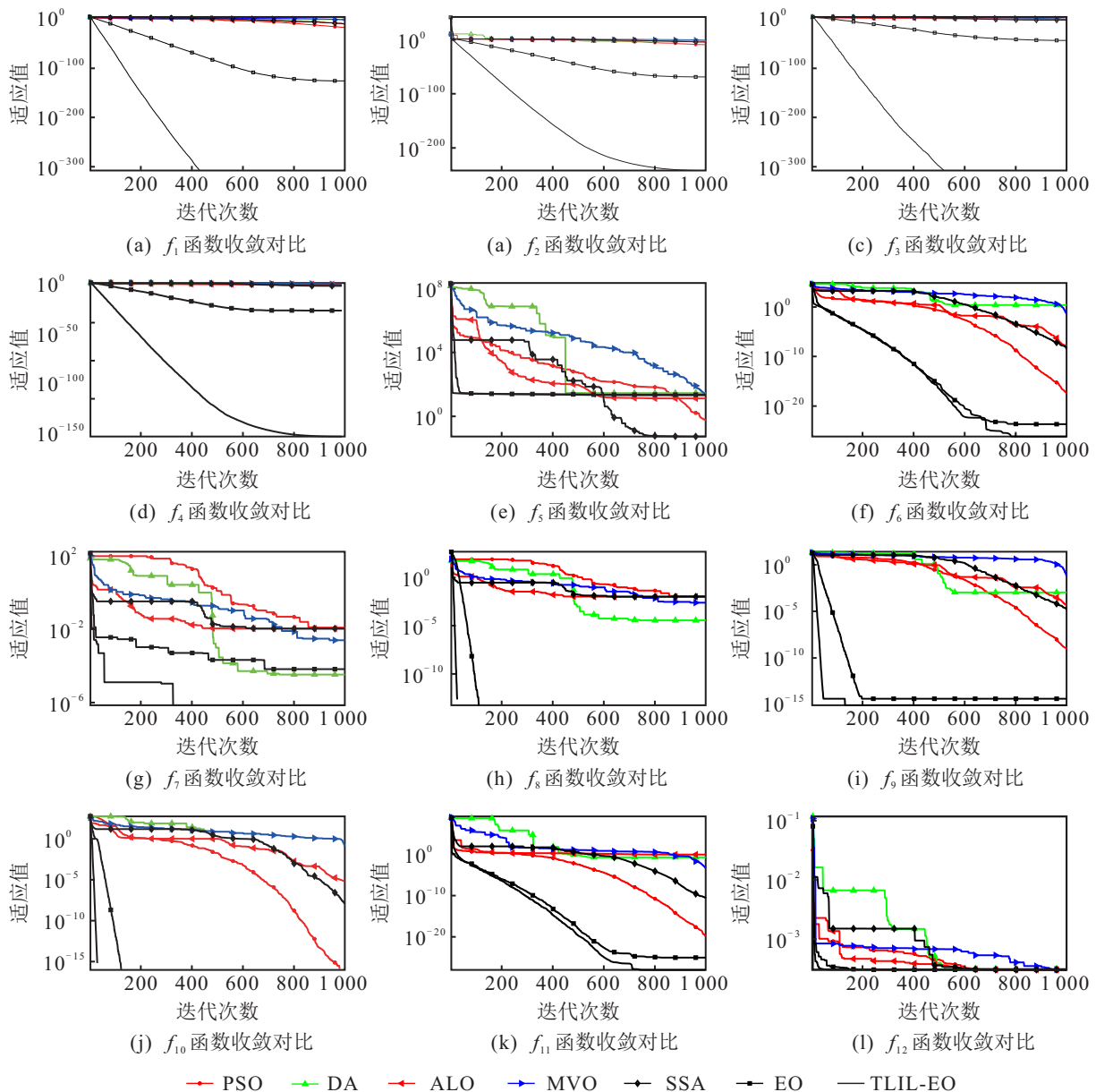


图3 测试函数收敛曲线

3.2.2 其他改进算法的寻优性能对比分析

为进一步验证 TLIT-EO 算法的优化性能,将其与其他基于 EO 算法的改进算法进行寻优效率对比,如 Tent 阶梯式策略平衡优化器算法 (Tent chaos step-by-step equilibrium optimizer algorithms, TSBS-EO)<sup>[21]</sup>、Tent 自适应权值策略平衡优化器算法 (Tent chaos adaptive weight equilibrium optimizer algorithms, TAW-

EO)<sup>[22]</sup>、Halton 序列初始化透镜成像学习策略的平衡优化器算法 (Halton lens imaging learning strategy equilibrium optimizer algorithms, HLIL-EO)<sup>[23]</sup>,参照测试函数的全局最小值,设定一个固定寻优值,各算法迭代过程中停止条件为小于固定寻优值或达到最大迭代次数,在相同测试环境和参数设置的情况下,表 4 通过寻优均值、方差和平均时间 3 个性能指标评估各算法的寻优性能。

表 4 实验结果

函数值	固定寻优值		TLIL-EO	EO	TSBS-EO	TAW-EO	HLIL-EO
$f_1$	1.0e-300	均值	3.79 e-301	1.44 e-107	2.12 e-301	4.41 e-301	4.14 e-301
		方差值	0	4.13 e-122	0	0	0
		平均时间	<b>0.712 37</b>	1.302 4	1.237 9	<b>0.927 75</b>	0.785 61
$f_2$	1.0e-100	均值	6.28 e-101	2.27 e-59	4.32 e-101	6.75 e-101	5.96 e-101
		方差值	3.39 e-202	4.64 e-118	5.48 e-202	3.07 e-202	1.28 e-116
		平均时间	<b>0.463 09</b>	1.323 3	1.201 6	0.747 3 3	0.497 8
$f_3$	1.0e-100	均值	3.78 e-101	1.05 e-29	3.15 e-101	5.33 e-101	3.91 e-101
		方差值	7.45 e-202	2.02 e-57	6.93 e-202	2.79 e-202	6.01 e-202
		平均时间	<b>0.652 06</b>	2.151 9	1.897 7	1.039 5	0.703 91
$f_4$	1.0e-100	均值	6.19 e-101	8.29 e-27	3.81 e-101	7.37 e-101	6.03 e-101
		方差值	4.03 e-202	4.03 e-52	4.97 e-202	1.57 e-202	6.03 e-202
		平均时间	<b>0.505 96</b>	1.305 1	1.225 8	0.809 1	0.531 75
$f_5$	25	均值	24.986 1	24.981 9	26.981 7	28.121 8	24.988 7
		方差值	0.000 179	0.001 047	0.042 21	0.061 81	0.000 19
		平均时间	1.146 5	<b>0.525 1</b>	1.356 3	1.358	1.196 4
$f_6$	1.0e-10	均值	9.23 e-11	9.32 e-11	1.000 9	3.341 8	9.34 e-11
		方差值	8.97 e-23	3.62 e-23	0.015 333	0.040 57	2.99 e-23
		平均时间	1.246 8	<b>0.585 91</b>	1.259 4	1.254 8	1.302 2
$f_7$	1.0e-05	均值	<b>3.94 e-05</b>	0.000 34	8.56 e-05	4.45 e-05	4.04 e-05
		方差值	1.34 e-09	1.74 e-08	4.20 e-09	1.12 e-09	9.84 e-10
		平均时间	3.099 3	1.648 5	<b>1.667 2</b>	<b>1.598 8</b>	3.142 5
$f_8$	1.0e-200	均值	0	0	0	0	0
		方差值	0	0	0	0	0
		平均时间	<b>0.053 63</b>	0.199 34	0.624 97	0.102 93	0.061 87
$f_9$	1.0e-15	均值	1.24 e-15	4.55 e-15	8.88 e-16	2.31 e-15	1.00 e-15
		方差值	1.17 e-30	4.21 e-31	0	3.13 e-31	4.21 e-31
		平均时间	2.193 6	1.343 4	<b>1.371 5</b>	<b>1.35</b>	2.224 2
$f_{10}$	1.0e-100	均值	0	0	0	0	0
		方差值	0	0	0	0	0
		平均时间	<b>0.069 87</b>	0.220 46	0.747 2	0.122 13	0.083 37
$f_{11}$	1.0e-10	均值	9.09 e-11	9.23 e-11	0.058 09	0.326 42	9.15 e-11
		方差值	7.45 e-23	6.35 e-23	0.000 158	0.001 208	5.94 e-23
		平均时间	3.363 8	1.066 5	2.603 9	2.586 7	3.454 6
$f_{12}$	3.0e-03	均值	0.003 35	0.003 36	0.002 287	0.001 849	0.005 22
		方差值	2.17 e-05	2.17 e-05	1.82 e-07	4.42 e-07	0.000 11
		平均时间	0.162 08	0.103 31	<b>0.033 614</b>	<b>0.017 713</b>	0.239 18

表 4 中的均值、方差和平均时间分别反映算法的收敛精度、寻优稳定性和寻优速度。相较于 EO 算法, TLIT-EO 算法在  $f_1, f_2, f_3, f_4, f_7$  函数上均能寻优到固定寻优值, 除  $f_9, f_{10}, f_{12}$  函数的寻优速度略差, 其他函数在计算性能方面均优于 EO 算法。相较于 TSBS-EO 算法, 除了  $f_7, f_9, f_{12}$  函数的寻优速度略

差, TLIT-EO 在其他函数的计算性能方面均优于 TSBS-EO 算法。相较于 TAW-EO 算法, TLIT-EO 算法在  $f_5, f_6, f_{11}$  函数上均能寻优到固定寻优值, 除  $f_7, f_9$  函数的寻优速度略差, 其他函数在计算性能方面均优于 TAW-EO 算法。通过 TLIT-EO 算法与 TSBS-EO 算法、TAW-EO 算法的性能对比可知, 基于阶梯式策略

改进的EO算法相较于其他改进算法具有更大的优势. 相比于HLIL-EO算法, 无论单峰还是多峰测试函数, TLIL-EO算法在寻优能力和寻优效率方面均优于HLIL-EO算法.

## 4 移动机器人路径规划

### 4.1 适应度函数

最优路径规划既要找到最短路径, 也要避免与障碍物发生碰撞. 文献[22]中提到的适应度函数为

$$F = L \cdot (1 + \omega \cdot V). \quad (18)$$

其中:  $\omega$ 为安全距离因子, 通常取100;  $L$ 为机器人起点到终点移动路径的长度, 其公式为

$$L = \sum_{i=1}^n \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}. \quad (19)$$

这里:  $n$ 为路径转向结点数,  $(x_i, y_i)$ 和 $(x_{i+1}, y_{i+1})$ 为两个相邻的转向节点;  $V$ 为标志变量, 当初始值 $V = 0$ 时, 适应度值为路径长度 $L$ , 其表达式为

$$V = \sum_{k=1}^d \text{mean}(\max(1 - D_k/R(k), 0)). \quad (20)$$

这里:  $d$ 为设置障碍物的数量,  $D_k$ 为路径上所有的插值点到第 $k$ 个障碍点的距离,  $R(k)$ 为第 $k$ 个障碍圆的半径. 式(20)为用来判断路径是否通过障碍物, 如没有通过障碍物, 则标志变量 $V$ 恒等于零, 否则大于0.

### 4.2 实验环境及参数设置

实验模拟环境: 本文采用模拟机器人在障碍物房间路径规划的场景, 在 $600 \times 600$ 的方形结构设置若干个圆形障碍物, 设置起点和终点, 在模拟环境中对TLIT-EO、EO和PSO算法进行路径规划仿真; 为了进一步保证几种算法具有可比性, 采用在三次样条插值<sup>[24]</sup>与B样条插值<sup>[25]</sup>两种编码环境下进行算法性能对比.

参数设置: 3种不同的环境模式, 设置起点为 $[-300, -300]$ , 终点为 $[300, 300]$ , 每个环境模式的障碍物数量、位置以及大小都不相同, 要求规划出与障碍物无碰撞的最优路径, 种群规模设置为50, 最大迭代次数设置为30, 重复运算50次, 取两种编码环境下最优收敛解、平均解记录于表5. TLIT-EO算法中全局探索权重系数 $\omega_1 = 1$ , 局部探索权重系数 $\omega_2 = 1.5$ , 其他参数设置与基本EO算法保持一致. PSO算法中的学习因子 $c_1$ 、 $c_2$ 分别决定粒子个体经验和群体经验对粒子运行轨迹的影响, 在PSO算法中, 个体经验和群体经验有着同等重要的影响力, 一般设置为 $c_1 = c_2 = 1.5$ , 本文采用这一取值.

表5 Friedman秩和验证结果

插值	函数值		TLIL-EO	EO	PSO
三样条	环境1	最优解	<b>870.3</b>	872.5	871.9
		平均解	887	882.8	1118.2
	环境2	最优解	<b>869.6</b>	870.5	873.3
		平均解	<b>872.7</b>	872.8	1967.1
	环境3	最优解	<b>873.5</b>	880.9	891.7
		平均解	<b>888.2</b>	909.9	985.4
B样条	环境1	最优解	<b>877.3</b>	878.9	959.4
		平均解	918.1	905.4	1112.5
	环境2	最优解	874.3	874.6	<b>870.7</b>
		平均解	<b>890.1</b>	892.4	1826.8
	环境3	最优解	<b>906.4</b>	912.8	929.9
		平均解	1045.8	1001.5	1097.3

### 4.3 路径规划实验与分析

如表5所示, 在采用三样条插值编码场景下可看出: TLIT-EO算法在3种环境下的最优解都小于其他两种对比算法, 平均解除了环境1大于EO算法以外, 其他环境均优于其他两种对比算法; 在采用B样条插值编码场景下, 最优解除了环境2大于PSO算法以外, 其他环境均优于其他两种算法.

## 5 结论

为解决传统EO算法收敛速度和易陷入局部最优等问题, 本文提出了一种改进的EO算法. 引入了Tent混沌映射和透镜成像学习策略提升算法的整体寻优性能; 选取12个Benchmark基准函数和移动机器人路径规划任务进行算法性能测试, 使用无重复双因素分析和Friedman秩和验证对算法性能进行验证. 实验结果表明: 改进后的EO算法具有更快的收敛速度和更高的收敛精度, 提升了迭代后期跳出局部最优解的能力, 验证了改进后的算法在移动机器人路径规划问题方面的有效性和优越性.

### 参考文献(References)

- [1] Faramarzi A, Heidarinejad M, Stephens B, et al. Equilibrium optimizer: A novel optimization algorithm[J]. Knowledge-Based Systems, 2020, 191: 105190.
- [2] Abdul-hamied D T, Shaheen A M, Salem W A, et al. Equilibrium optimizer based multi dimensions operation of hybrid AC/DC grids[J]. Alexandria Engineering Journal, 2020, 59(6): 4787-4803.
- [3] Nusair K, Alhmoud L. Application of equilibrium optimizer algorithm for optimal power flow with high penetration of renewable energy[J]. Energies, 2020, 13(22): 6066.
- [4] Rabehi A, Nail B, Helal H, et al. Optimal estimation of Schottky diode parameters using advanced swarm intelligence algorithms[J]. Semiconductors, 2020, 54(11): 1398-1405.
- [5] Mirjalili S, Gandomi A H, Mirjalili S Z, et al. Salp swarm



- algorithm: A bio-inspired optimizer for engineering design problems[J]. *Advances in Engineering Software*, 2017, 114: 163-191.
- [6] Mirjalili S. The ant lion optimizer[J]. *Advances in Engineering Software*, 2015, 83(5): 80-98.
- [7] Kennedy J, Eberhart R. Particle swarm optimization[C]. *Proceedings of the IEEE International Conference on Neural Networks*. Perth, 1995: 1942-1948.
- [8] Wunnava A, Naik M K, Panda R, et al. A novel interdependence based multilevel thresholding technique using adaptive equilibrium optimizer[J]. *Engineering Applications of Artificial Intelligence*, 2020, 94: 103836.
- [9] Abdel-Basset M, Mohamed R, Mirjalili S, et al. MOEO-EED: A multi-objective equilibrium optimizer with exploration-exploitation dominance strategy[J]. *Knowledge-Based Systems*, 2021, 214: 106717.
- [10] Abdel-Basset M, Mohamed R, Mirjalili S, et al. Solar photovoltaic parameter estimation using an improved equilibrium optimizer[J]. *Solar Energy*, 2020, 209: 694-708.
- [11] Tang A D, Han T, Zhou H, et al. An improved equilibrium optimizer with application in unmanned aerial vehicle path planning[J]. *Sensors*, 2021, 21(5): 1814.
- [12] Kardani N, Bardhan A, Gupta S, et al. Predicting permeability of tight carbonates using a hybrid machine learning approach of modified equilibrium optimizer and extreme learning machine[J]. *Acta Geotechnica*, 2022, 17(4): 1239-1255.
- [13] 张振兴, 杨任农, 房育寰, 等. 自适应 Tent 混沌搜索的蚁狮优化算法[J]. *哈尔滨工业大学学报*, 2018, 50(5): 152-159.  
(Zhang Z X, Yang R N, Fang Y H, et al. Ant lion optimization algorithm based on self-adaptive Tent chaos search[J]. *Journal of Harbin Institute of Technology*, 2018, 50(5): 152-159.)
- [14] 喻飞, 李元香, 魏波, 等. 透镜成像反学习策略在粒子群算法中的应用[J]. *电子学报*, 2014, 42(2): 230-235.  
(Yu F, Li Y X, Wei B, et al. The application of a novel OBL based on lens imaging principle in PSO[J]. *Acta Electronica Sinica*, 2014, 42(2): 230-235.)
- [15] 龙文, 伍铁斌, 唐明珠, 等. 基于透镜成像学习策略的灰狼优化算法[J]. *自动化学报*, 2020, 46(10): 2148-2164.  
(Long W, Wu T B, Tang M Z, et al. Grey wolf optimizer algorithm based on lens imaging learning strategy[J]. *Acta Automatica Sinica*, 2020, 46(10): 2148-2164.)
- [16] Mirjalili S. The ant lion optimizer[J]. *Advances in Engineering Software*, 2015, 83(5): 80-98.
- [17] Mirjalili S, Gandomi A H, Mirjalili S Z, et al. Salp swarm algorithm: A bio-inspired optimizer for engineering design problems[J]. *Advances in Engineering Software*, 2017, 114(6): 163-191.
- [18] Mirjalili S. Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm[J]. *Knowledge-Based Systems*, 2015, 89: 228-249.
- [19] Mirjalili S. Dragonfly algorithm: A new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems[J]. *Neural Computing and Applications*, 2016, 27(4): 1053-1073.
- [20] Mirjalili S, Mirjalili S M, Hatamlou A. Multi-verse optimizer: A nature-inspired algorithm for global optimization[J]. *Neural Computing and Applications*, 2016, 27(2): 495-513.
- [21] 方群, 徐青. 基于改进粒子群算法的无人机三维航迹规划[J]. *西北工业大学学报*, 2017, 35(1): 66-73.  
(Fang Q, Xu Q. 3D route planning for UAV based on improved PSO algorithm[J]. *Journal of Northwestern Polytechnical University*, 2017, 35(1): 66-73.)
- [22] 张达敏, 陈忠云, 辛梓芸, 等. 基于疯狂自适应的樽海鞘群算法[J]. *控制与决策*, 2020, 35(9): 2112-2120.  
(Zhang D M, Chen Z Y, Xin Z Y, et al. Salp swarm algorithm based on craziness and adaptive[J]. *Control and Decision*, 2020, 35(9): 2112-2120.)
- [23] 刘成汉, 何庆. 融合多策略的黄金正弦黑猩猩优化算法[J/OL]. DOI: 10.16383/j.aas.c210313.  
(Liu C H, He Q. Golden sine chimp optimization algorithm integrating multiple strategies[J/OL]. DOI: 10.16383/j.aas.c210313.)
- [24] 刘景森, 吉宏远, 李煜. 基于改进蝙蝠算法和三次样条插值的机器人路径规划[J]. *自动化学报*, 2021, 47(7): 1710-1719.  
(Liu J S, Ji H Y, Li Y. Robot path planning based on improved bat algorithm and cubic spline interpolation[J]. *Acta Automatica Sinica*, 2021, 47(7): 1710-1719.)
- [25] 方建文, 晁永生, 袁逸萍. 改进 B 样条插值法的焊接机器人关节轨迹优化[J]. *机械设计与制造*, 2021(10): 1-4.  
(Fang J W, Chao Y S, Yuan Y P. Research on the trajectory optimization of welding robot based on improved B-spline interpolation method[J]. *Machinery Design & Manufacture*, 2021(10): 1-4.)

## 作者简介

周鹏(1991—), 男, 硕士, 从事智能优化算法、系统辨识与控制方法的研究, E-mail: zhoupeng\_hz@163.com;

董朝轶(1976—), 男, 教授, 博士, 从事地面移动机器人的动态建模、导航、复杂生物网络建模与网络结构辨识等研究, E-mail: dongchaoyi@hotmail.com;

陈晓艳(1975—), 女, 副教授, 硕士, 从事模式识别、控制工程的研究, E-mail: chenxiaoyan@imut.edu.cn;

赵肖懿(1993—), 女, 硕士, 从事图像识别、深度学习的研究, E-mail: zhaoxiaoyi310@163.com;

王启来(1994—), 男, 硕士, 从事移动机器人导航、制导与控制的研究, E-mail: wangqilai0723@163.com.