



求解多目标点路径规划问题的离散头脑风暴算法

陈强, 马健, 杨繁

引用本文:

陈强,马健,杨繁. 求解多目标点路径规划问题的离散头脑风暴算法[J]. 智能系统学报, 2023, 18(1): 96–103.

CHEN Qiang,MA Jian,YANG Fan. Discrete brainstorm optimization algorithm for solving multi-target route planning problems[J]. *CAA Transactions on Intelligent Systems*, 2023, 18(1): 96–103.

在线阅读 View online: <https://dx.doi.org/10.11992/tis.202206018>

您可能感兴趣的其他文章

分组教学蚁群算法改进及其在机器人路径规划中应用

Improvement of ant colony algorithm in group teaching and its application in robot path planning

智能系统学报. 2022, 17(4): 764–771 <https://dx.doi.org/10.11992/tis.202108020>

基于变步长蚁群算法的移动机器人路径规划

Mobile robot path planning based on variable-step ant colony algorithm

智能系统学报. 2021, 16(2): 330–337 <https://dx.doi.org/10.11992/tis.202004011>

融合改进A*算法和Morphin算法的移动机器人动态路径规划

Mobile-robot dynamic path planning based on improved A* and Morphin algorithms

智能系统学报. 2020, 15(3): 546–552 <https://dx.doi.org/10.11992/tis.201812023>

移动机器人全覆盖信度函数路径规划算法

Complete-coverage path planning algorithm of mobile robot based on belief function

智能系统学报. 2018, 13(2): 314–321 <https://dx.doi.org/10.11992/tis.201610006>

基于改进粒子群算法的移动机器人多目标点路径规划

Mobile robot multi-goal path planning using improved particle swarm optimization

智能系统学报. 2017, 12(3): 301–309 <https://dx.doi.org/10.11992/tis.201606046>

DOI: 10.11992/tis.202206018

网络出版地址: <https://kns.cnki.net/kcms/detail/23.1538.TP.20221209.1507.001.html>

求解多目标点路径规划问题的离散头脑风暴算法

陈强, 马健, 杨繁

(浙江工业大学信息工程学院, 浙江 杭州 310013)

摘 要: 为保证移动机器人以最短路径遍历多目标点, 该文提出一种基于离散头脑风暴的多目标点路径规划算法。首先, 考虑障碍物对路径规划的影响, 将目标点间的最短避障距离作为评判依据, 提高规划路径合理性。其次, 针对传统离散头脑风暴算法在解决组合类优化问题时提前陷入局部最优的问题, 提出一种启发式自适应路径优化策略, 通过设计与迭代次数相关的适应度选择函数以及改进启发式交叉算子, 增加路径多样性和提高算法收敛速度。基于栅格法建立地图模型, 在不同环境地图中选取多个目标进行对比仿真, 验证所提算法的有效性以及对不同环境的适应性。

关键词: 移动机器人; 路径规划; 离散头脑风暴; 组合类优化问题; 局部最优; 最短避障距离; 适应度选择函数; 启发式交叉算子

中图分类号: TP399; TP18; TN911.7 **文献标志码:** A **文章编号:** 1673-4785(2023)01-0096-08

中文引用格式: 陈强, 马健, 杨繁. 求解多目标点路径规划问题的离散头脑风暴算法 [J]. 智能系统学报, 2023, 18(1): 96-103.

英文引用格式: CHEN Qiang, MA Jian, YANG Fan. Discrete brainstorm optimization algorithm for solving multi-target route planning problems[J]. CAAI transactions on intelligent systems, 2023, 18(1): 96-103.

Discrete brainstorm optimization algorithm for solving multi-target route planning problems

CHEN Qiang, MA Jian, YANG Fan

(College of Information Engineering, Zhejiang University of Technology, Hangzhou 310013, China)

Abstract: In this paper, a multi-target point path planning algorithm is proposed based on discrete brainstorm optimization (DBSO) to guarantee that a mobile robot can traverse multiple target points with the shortest path. First, considering the influence of obstacles on path planning, the shortest obstacle avoidance distance between target points is used as the judgment basis to improve rationality of the planned path. Secondly, the traditional discrete brainstorming algorithm will fall into local optimum in advance when solving combinatorial optimization problems. Therefore, a heuristic adaptive path optimization strategy is proposed. It increases path diversity and improves convergence speed of the algorithm by designing a fitness selection function related to the number of iterations and improving the heuristic crossover operator. A map model is established based on the grid method, and multiple targets are selected in different environmental maps for comparison and simulation to verify effectiveness of the proposed algorithm and its adaptability to different environments.

Keywords: mobile robot; path planning; discrete brainstorm optimization; combinatorial optimization problem; local optimum; shortest obstacle avoidance distance; fitness selection function; heuristic crossover operator

多目标点路径规划技术是移动机器人完成自

主导航任务的前提与基础, 其目的在于使移动机器人能在具有障碍物的环境中按照路径最短的评价标准寻找一条经过所有目标点的无碰撞路径^[1]。在实际生产生活中为尽量减少往返于各个目标点之间所需的路程和能耗, 则需要优化各个任务的

收稿日期: 2022-06-11. 网络出版日期: 2022-12-12.

基金项目: 国家自然科学基金项目(61973274); 浙江省自然科学基金重点项目(LZ22F030007).

通信作者: 陈强. E-mail: sdnjchq@zjut.edu.cn.

©《智能系统学报》编辑部版权所有

执行顺序,即考虑移动机器人的多目标点路径规划问题^[2-3],其求解方法主要有精确算法^[4]和启发式算法^[5-6]两类。精确算法主要用于解决规模相对较小的组合类优化问题,其计算时间往往随问题规模的增大呈爆炸式增长。启发式算法在可接受范围内给出问题的解,相比于精确算法,在处理大规模多目标点路径规划技术时,具有更高的鲁棒性、可行性^[7]。因此国内外学者主要采用启发式算法对多目标点路径规划技术及其变体问题进行研究,并针对不同地图环境提出了多种优化方法^[8-16]。文献[8]提出一种移动机器人多目标搜寻的D*-蚁群融合算法,计算出各目标点之间的最短避障距离,并将其作为评价依据规划出一条最优路径;文献[9]针对粒子群算法存在的早熟现象,将反向学习策略引入粒子群算法,并针对相应的惯性权重和学习因子进行改进。与传统的启发式算法相比,离散头脑风暴(discrete brain storm optimization, DBSO)算法^[10]具有结构简单、设计参数少以及较强的搜索能力等优点,被广泛应用于无人机路径规划^[11]、机器人定位建图^[12]、多目标协同搜索范式^[13]、多车型绿色车辆路径优化^[14]、水面清理垃圾机器人^[15]、变电站的巡检机器人^[16]等领域。文献[13]通过在环境和机器人约束下,将机器人中的成员与DBSO算法中的个体进行匹配,提出了一种基于机器人BSO的群体机器人协同搜索框架。文献[14]设计DBSO算法解决以系统总成本最小为目标的多目标绿色车辆路径优化模型。针对DBSO算法在解决多目标路径规划时易陷入局部最优而导致早熟收敛的问题,许多学者提出了不同的解决方案^[17-18]。其中文献[17]提出一种在DBSO算法中通过保持种群多样性的策略,提高了算法跳出局部最优的概率;文献[18]提出基于讨论机制的头脑风暴算法,通过改变组内与组间讨论次数以调节算法的局部与全局搜索能力,避免了提前陷入局部最优的可能。

基于上述讨论,本文针对障碍物环境下的多目标点路径规划问题设计了一种启发式自适应离散头脑风暴算法。首先,考虑障碍物对路径规划的影响,将距离评判依据由欧式距离替换为目标点间的最短避障距离,提高规划路径的合理性。其次,针对传统DBSO算法在解决组合类优化问题时提前陷入局部最优的问题,提出一种启发式自适应路径优化策略,主要包括:1)设计交换、插入操作产生新的路径从而增加路径多样性;2)设计双向启发式交叉算子,提高路径的质量和算法

收敛速度;3)设计与迭代次数相关的适应度选择函数,克服算法迭代前期收敛速度慢和后期路径多样性下降的问题,提高算法跳出局部最优的能力。

1 多目标点路径规划问题

机器人多目标路径规划问题一般性描述为:某机器人要遍历 n 个目标点,且每个目标只访问一次,最终回到初始地点,求一条以最短距离经过所有目标点的路径^[19]。

定义遍历 n 个目标点的路径为 $S_i=(s_{i,1}, s_{i,2}, \dots, s_{i,n}, s_{i,n+1})$, $s_{i,n+1}=s_{i,1}$, 目标点 $s_{i,j}$ 的坐标为 $(x_{i,j}, y_{i,j})$ 。令 d 为目标点 $s_{i,j}$ 和 $s_{i,j+1}$ 之间的最短避障距离^[20], 则遍历路径的总距离为

$$D_i = \sum_{j=1}^n d(s_{i,j}, s_{i,j+1}) \quad (1)$$

本文的目标是,设计离散头脑风暴算法,使 D_i 值最小,即实现移动机器人以最短路径遍历所有目标点。

2 启发式自适应离散头脑风暴算法

针对移动机器人多目标路径规划问题,本文设计一种启发式路径优化策略产生新路径。同时,为解决传统DBSO算法早熟的问题,设计适应度选择函数降低算法过早陷入局部最优的可能。算法总体流程图如图1所示,具体实现步骤如下:

- 1)初始化参数并产生 $m(m=n)$ 条路径;
- 2)设计K-Means++聚类算法,将 n 条路径分为 K 个组,每组中距离最小的路径为中心个体;
- 3)设计与迭代次数相关的适应度选择函数,确定路径优化方式;提出启发式自适应优化路径策略,通过交换、插入操作增加路径多样性,而通过双向启发式交叉算子以提高路径质量和算法收敛速度;
- 4)保留适应度值低的路径,直至最大迭代次数输出最优路径,采用A*路径规划算法根据输出的最优路径依次规划行驶路线。

2.1 初始化路径及各参数

本文将所有目标点进行 $1 \sim n$ (n 为目标点个数)整数编码。在多目标点路径规划中,一条路径代表一个潜在的可行解,路径代表机器人经过目标点的顺序序列。本文采用随机生成的方式构造初始路径,并采用式(1)计算各路径的适应度值,适应度值越低表明路径越短。

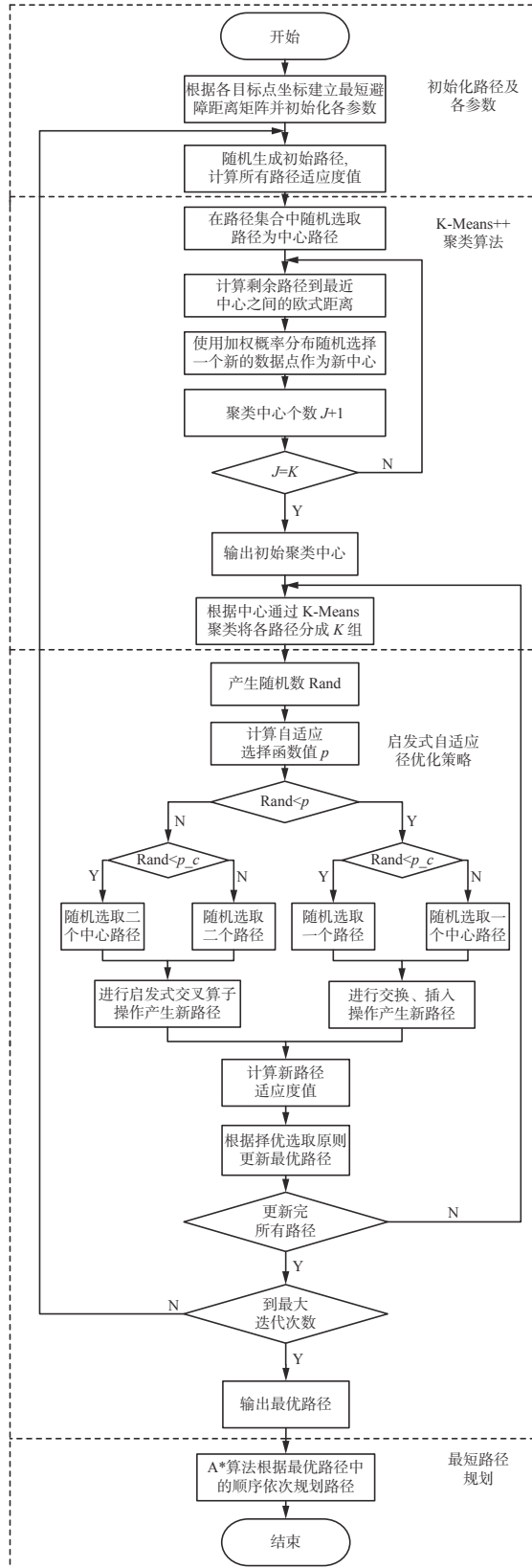


图 1 算法总体流程

Fig. 1 Overall flow chart of algorithm

2.2 K-Means++聚类算法

为了减小初始聚类中心的选取对算法的影响, 本文采用聚类效果更好的 K-Means++ 聚类算法^[21]替代传统的 K-Means 聚类算法。K-Means++

聚类算法是在 K-Means 聚类算法基础上, 通过设置不同位置作为聚类中心的概率来获得相对距离更远的聚类中心, 降低对聚类结果的影响, 从而提高算法鲁棒性。其算法实现过程包括:

- 1) 在路径集合 X 中随机选择一个路径作为中心;
- 2) 计算每个路径与当前聚类中心之间的最短距离 (即与最近一个聚类中心的距离), 用 $D(x)$ 表示;
- 3) 计算每个路径被选中为下一个聚类中心的概率为 $D(x)^2 / \sum_{x \in X} D(x)^2$, 采用轮盘赌的方法选择下一个聚类中心;
- 4) 重复 2) 和 3), 直到选择 K 个中心 (K 为人为预先设定);
- 5) 利用选择的初始中心采用 K-Means 聚类算法。

2.3 启发式自适应路径优化策略

启发式自适应路径优化策略包括 3 部分: 1) 适应度选择函数; 2) 单条路径优化; 3) 两条路径间的组合优化。其中, 适应度选择函数是通过设计与迭代次数相关的函数以调节被选择概率; 单条路径优化是指对某条路径中的一个位置或几个位置进行操作; 两条路径间的组合优化是指对不同组中的路径进行双向启发式交叉算子操作。

2.3.1 适应度选择函数

由于算法对多样性和收敛速度随迭代次数的变化的需求符合 $\theta = \frac{\text{gen}}{G_{\max}}$ 的变化曲线, 为降低人为设置固定参数或分段设置参数方式带来的调试难度, 本文设计和迭代次数相关的适应度选择函数, 使得优化算法在迭代前期通过提高两个路径间优化的操作概率, 加快算法的收敛速度。此外, 随着迭代次数的增加, 通过提高单条路径优化的操作概率, 从而增加路径多样性。为避免当迭代次数较少时计算出的 θ 过小, 以及迭代后期 θ 无限接近 1 的不合理曲线变化, 本文设计 β 和 γ 参数, 保证算法在前期的多样性, 以及后期的收敛速度。适应度选择函数 p 构造如下:

$$p = \beta + \frac{\text{gen}}{G_{\max}} \times \gamma \quad (2)$$

式中: β 为最小单条路径优化概率; gen 为当前迭代次数; G_{\max} 为最大迭代次数; γ 为控制最大单条路径优化概率的系数。

本文中单条路径优化概率为 p , 两条路径间的组合优化概率为 $1-p$ 。算法迭代前期 p 值较小, 两条路径间的组合优化概率高, 收敛速度快; 当迭代次数增加, gen 值变大, 使得式 (2) 中 p 值增大,

此时单条路径优化概率增大, 多样性提高, 算法跳出局部最优的可能变大。

2.3.2 单条路径优化

为增加路径的多样性, 在迭代过程中对一个路径进行交换、插入操作^[22]。解 $S_i = \{s_{i,1}, s_{i,2}, \dots, s_{i,n}\}$ 优化的具体规则如下:

交换操作 (如图 2 所示): 在 $1 \sim n$ 区间内随机产生不同的整数, 并将对应位置的元素交换。以 8 个目标点为例来具体说明交换操作原理: 其中, S_{old} 为当前 S_i 路径的子代路径, S_{new} 为当前 S_i 路径更新后的子代路径, 选取一个路径如图 2(a) 所示, 从 $1 \sim 8$ 中随机产生不同的整数 2 和 5 后交换 S_{old} 对应位置的元素, 更新后的子代 S_{new} 如图 2(b) 所示。

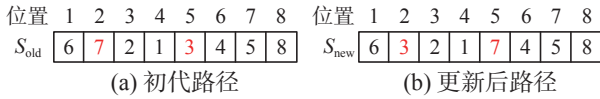


图 2 交换操作

Fig. 2 Swap operation

插入操作 (如图 3 所示): 在 $1 \sim n$ 区间内随机产生不同的整数 a 、 b ($a \neq b$) 然后将这第 1 个位置上的元素插入到第 2 个元素后。为说明插入操作的原理, 同样以 8 个目标点为例, 选取一个路径如图 3(a), 随机从 18 产生不同的整数 2 和 5, 后将位置 5 的元素插入位置 2 后, 更新后的子代 S_{new} 如图 3(b) 所示。

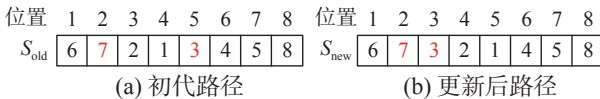


图 3 插入操作

Fig. 3 Insert operation

2.3.3 两条路径间的组合优化

启发式交叉算子以继承初代路径中的元素为基础重构可行解, 具有收敛速度快的特点^[23]。由于搜索方向固定, 当可行解与最优解的部分片段一致, 子代路径按照同一方向搜索会产生无效搜索并可能陷入局部最优。为克服这一缺陷, 本文提出一种改进搜索策略, 其主体思想是: 子代开始搜索方向变为双向搜索, 扩大搜索范围, 有利于提高算法的收敛速度。选取不同组的路径作为初代路径 $S_1 = \{s_{1,1}, s_{1,2}, \dots, s_{1,n}\}$ 和 $S_2 = \{s_{2,1}, s_{2,2}, \dots, s_{2,n}\}$, 利用改进的启发式交叉算子产生子代路径的基本步骤如下:

1) 随机在 $1 \sim n$ 中选取一个点作为当前点 s_j , 标记为 L, 将其作为子代路径的第一个位置;

2) 分别找出 s_j 在 S_1 和 S_2 的位置, 若 s_j 在初代路

径的首位, 则其上一位为最后一位; 若 s_j 在初代路径的最末位, 则其下一位为第一位。在 S_1 和 S_2 中分别找到相对于 s_j 前后位置的 $s_{1,j-1}$, $s_{1,j+1}$, $s_{2,j-1}$, $s_{2,j+1}$, 并计算他们的距离 $d_1(s_{1,j}, s_{1,j-1})$, $d_2(s_{1,j}, s_{1,j+1})$, $d_3(s_{2,j}, s_{2,j-1})$, $d_4(s_{2,j}, s_{2,j+1})$; 比较 d_1, d_2, d_3, d_4 大小并将距离最小值对应的位置元素加入到子代 S_{new} 中 s_j 的后一位中;

3) 删除在 S_1 和 S_2 中的 s_j , 更新 S_1 和 S_2 ;

4) 重复 2) 和 3), 直到 S_1 和 S_2 中元素只有两个, 非 L 元素为最后一位, 输出路径。

本文以 8 个目标点为例, 说明双向启发式算子的实现过程: 产生的初代个体如图 4(a) 所示; 随机选 3 作为 S_{new} 的第一位, 设目标点 3 到目标点 1 的距离最短, 经过一次 2) 的操作, 得到图 4(b); 后执行 3), 更新后的 S_1 和 S_2 如图 4(c) 所示; 重复 2)~3), 直至 S_1 和 S_2 元素中的元素都只有被标记为 L 的元素, 循环结束, 输出更新后的 S_{new} , 如图 4(d) 所示。

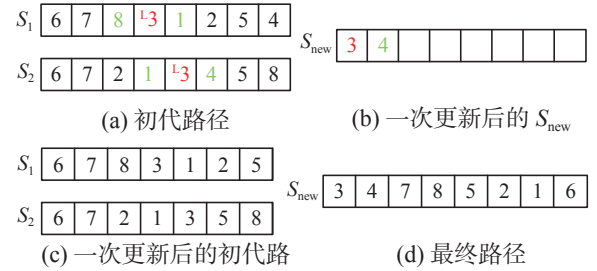


图 4 改进的启发式交叉操作

Fig. 4 Improved heuristic crossover operation

2.4 最短路径规划

本文在规划过程中考虑到移动机器人在障碍环境下的不可穿越障碍的移动约束, 采用 A* 算法^[24]规划出的最短避障距离替换无视障碍物的欧式距离, 使得移动机器人在有障碍环境下规划出的路线与实际最优路线一致。

3 仿真结果及分析

为验证本文算法的优越性及实用性, 在 AMD Ryzen5-4600U with Radeon Graphics、Window11、64 位操作系统和 MATLAB 2020b 的环境下, 在相同地图环境下对蚁群 (ant colony optimization, ACO) 算法^[25]、传统 DBSO 算法^[26]、蚁群算法-粒子群 (ant colony optimization-improved particle swarm optimization, ACO-IPSO) 算法^[9]、以及本文算法进行对比仿真。为了衡量算法的性能, 文中使用了多个评价指标, 其含义如表 1 所示。本文算法参数设置如下: 最大迭代次数为 $G_{max} = 500$, 本文 DBSO 算法: 聚类组数 $K = 5$, 选择中心路径

的概率为 $p_c = 0.2$, 最小单条路径优化概率 $\beta = 0.1$ 以及最大单条路径优化概率 $\gamma = 0.8$ 。为避免算法的偶然性, 每组仿真分别进行 20 次, 其结果如表 2 和表 3 所示。

表 1 评价指标及含义

Table 1 Evaluation index and meanings

评价指标	含义
最优解	所有仿真中距离最短解
平均解	所有仿真中所求解的平均值
标准差	$\{[(\text{解}_1 - \text{平均解})^2 + (\text{解}_2 - \text{平均解})^2 + \dots + (\text{解}_N - \text{平均解})^2]/N\}^{0.5}$

表 2 简单障碍环境下各算法仿真结果

Table 2 Simulation results of each algorithm in simple obstacle environment

算法	每次仿真的总距离						
ACO	695	694	700	702	697	710	709
	691	702	702	709	703	703	689
	691	694	703	789	690	695	
ACO-IPSO	568	570	569	575	570	569	575
	585	575	580	570	580	589	580
	575	570	585	569	590	568	
传统DBSO	561	575	570	563	569	561	560
	560	570	559	570	560	559	561
	570	565	563	569	561	575	
本文算法	553	553	559	553	553	553	556
	553	556	554	554	553	553	554
	553	553	559	556	554	556	

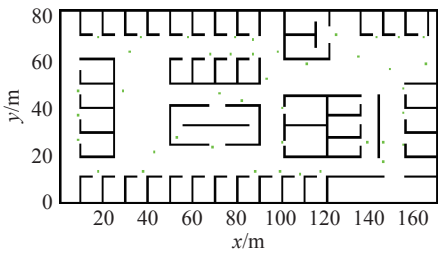
表 3 复杂障碍环境下各算法仿真结果

Table 3 Simulation results of each algorithm in complex obstacle environment

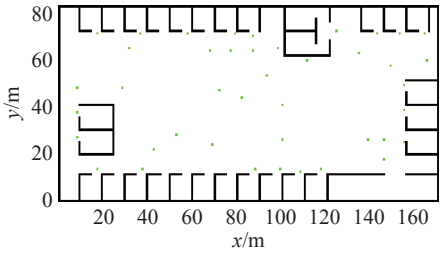
算法	每次仿真的总距离						
ACO	709	732	750	712	720	713	715
	713	750	731	732	718	750	713
	716	720	713	718	731	732	
ACO-IPSO	626	635	636	635	626	626	641
	636	635	635	627	626	633	640
	636	636	635	636	635	636	
传统DBSO	622	625	627	625	622	633	622
	625	622	622	627	637	627	625
	625	624	624	622	627	633	
本文算法	582	586	586	580	580	581	580
	580	580	583	582	581	580	586
	582	580	580	580	582	580	

在 80×170 的二维栅格地图内对各算法进行测试, 设置如图 5 所示的简单障碍环境地图和复

杂障碍环境地图。



(a) 简单障碍



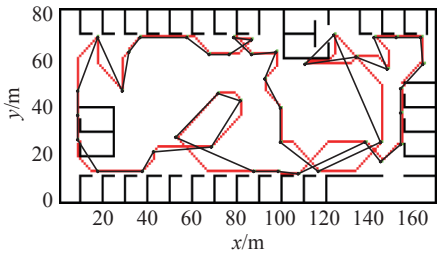
(b) 复杂障碍

图 5 环境地图

Fig. 5 Environmental map

3.1 简单障碍环境下各算法对比

为验证本文算法在简单障碍环境下多目标路径规划的性能, 在环境中选取 42 个目标点进行规划, 算法中的距离评估依据为目标点间的欧式距离, 规划结果如图 6 和图 7 所示。图 6 中, 黑色区域为障碍, 绿色区域为目标点, 白色区域为可行区域, 红色区域为机器人实际行驶路径, 黑线连成的回路为机器人的理想行驶路径。图 6 为各算法在 20 次仿真中最好的规划结果, 其中图 6(a) 描述了传统 ACO 的规划结果, 可以看出路径产生大量交叉, 路径总距离较长; 图 6(b) 和 (c) 分别是传统 DBSO 算法和 ACO-IPSO 算法, 图中可以看出在深蓝色虚线框内 S1, S2, S3 的区域, 由于陷入局部最优导致路径交叉, 路径总距离变长。图 6(d) 为本文算法规划出的结果, 规划出的路径无交叉, 且路径总距离相较其他算法较短, 同时经验证也是最短路径。由图 7 可以看出, 在相同条件下, 本文提出的 DBSO 算法在全局最优以及收敛速度方面上优于其他算法其中蓝色曲线为 ACO, 黑色为 DBSO, 粉色为传统 ACO-IPSO, 绿色为本文算法。



(a) ACO 算法

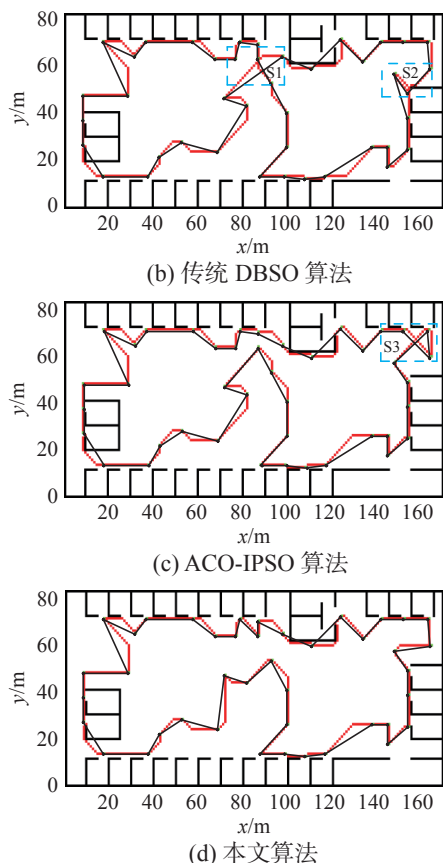


图 6 简单障碍环境下各算法路径规划结果

Fig. 6 Path planning results of each algorithm in simple obstacle environment

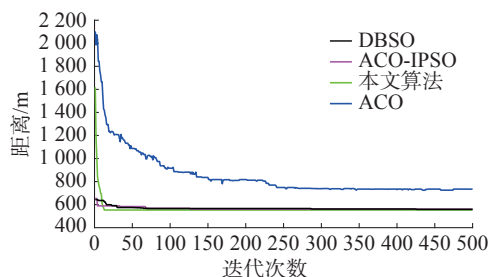


图 7 各算法在 500 代内的收敛曲线

Fig. 7 Convergence curves of each algorithm within 500 generations

3.2 复杂障碍环境下各算法对比

为进一步验证本文算法在复杂障碍环境下多目标路径规划的优势, 采用与简单障碍环境相同的目标点进行仿真测试。与简单障碍环境不同, 在复杂障碍环境下本文所提算法中采用两目标点之间的最短避障距离作为距离评估依据, 规划结果如下图 8 所示。图 8 为各算法在复杂障碍环境下 20 次仿真中最好的规划结果, 从图 8(a)、(b) 和 (c) 可以看出, 在相应深蓝虚线框 S4, S5, S6 位置处, 由于移动机器人的移动规则限制, 采用传统的欧氏距离作为评估依据, 在规划时算法会认为两点直线最近的点为最优解, 进而导致了与实际不符合规划顺序。由图 8(d) 可知, 由于采用了

最短避障距离作为评估依据, 避免了因为距离评估问题造成的误判而增加多余的行驶距离。

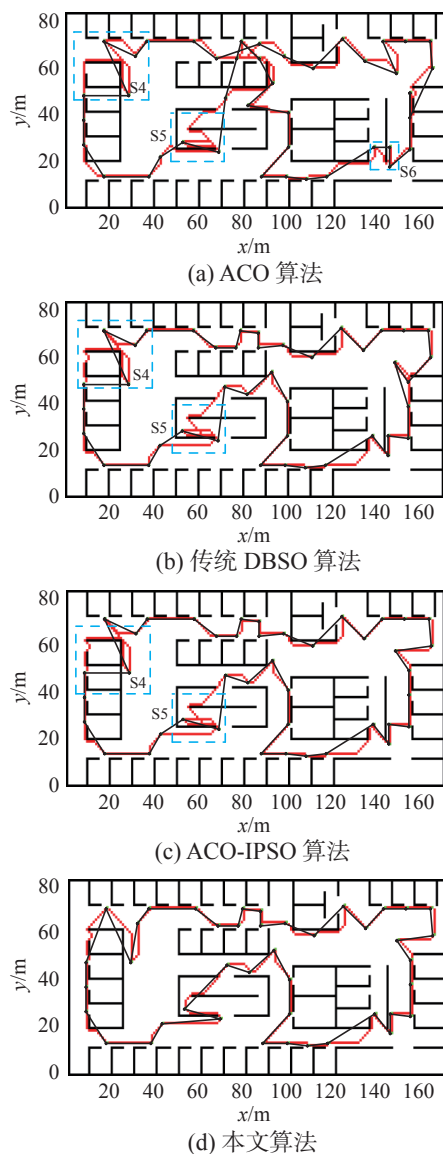


图 8 复杂障碍环境下各算法路径规划结果

Fig. 8 Path planning results of each algorithm in complex obstacle environment

由表 4 和表 5 可知, 当目标点数为 42 时, 在简单障碍环境下 20 次仿真中, 本文算法所求的距离最优解与 ACO, ACO-IPSO 和传统 DBSO 相比, 分别减少了 19.73%、2.64%、1.07%; 而在复杂障碍环境下, 本文算法所求距离最优解与 ACO, 传统 DBSO 和 ACO-IPSO 相比, 分别减少了 18.19%、7.34%、6.75%。且无论在简单障碍还是复杂障碍环境下, 所提算法得到的平均解和标准差均优于其他算法。由图 9 可以看出, 在相同环境下, 本文提出的 DBSO 算法在全局最优以及收敛速度方面上优于其他算法, 其中蓝色曲线为 ACO, 黑色为 DBSO, 粉色为传统 ACO-IPSO, 绿色为本文算法。此外, 分析标准差数据可知, 多次试验中本文算法计算出的距离变化幅度较小, 因而较其他

算法,所提方法具有较好的鲁棒性。综上,可以看出在不同环境下本文所提算法均可以规划出令人满意的效果。

表 4 简单障碍环境下的数据对比

Table 4 Data comparison in simple obstacle environment

算法	最优解	平均解	标准差
ACO	689	698.4	6.65
传统DBSO	568	575.6	7.05
ACO-IPSO	559	565.1	5.25
本文算法	553	554.4	1.91

表 5 复杂障碍环境下的数据对比

Table 5 Data comparison in complex obstacle environment

算法	最优解	平均解	标准差
ACO	709	724.4	13.1
传统DBSO	626	633.8	4.82
ACO-IPSO	622	625.8	4.07
本文算法	580	581.5	2.09

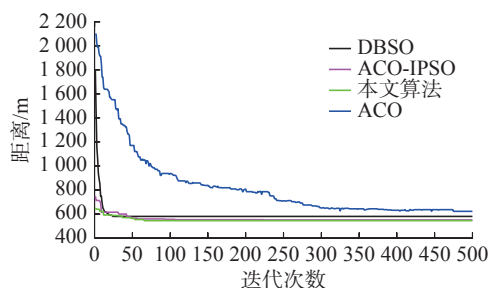


图 9 各算法在 500 代内的收敛曲线图

Fig. 9 Convergence curves of each algorithm within 500 generations

4 结束语

本文针对移动机器人多目标点路径规划问题提出了一种有效的解决方案。考虑传统 DBSO 算法在处理多目标点路径规划问题上的局限性,提出了改进的启发式路径优化策略,进而增加了算法的收敛速度。通过设计与迭代次数相关的适应度函数,降低了陷入局部最优的概率。将欧氏距离更换为更加实际的最短避障距离,提高了算法的实用性。仿真结果表明,本文所提启发式自适应 DBSO 算法可以缩短移动机器人往返目标点之间的路程,从而有效解决移动机器人多目标点路径规划问题。将本文提出的新方法应用到动态环境下的移动机器人的多目标点路径规划是下一步的研究内容。

参考文献:

- [1] LIN Wei, DELGADO-FRIAS J G, GAUSE D C, et al. Hybrid Newton-raphson genetic algorithm for the traveling salesman problem[J]. *Cybernetics and systems*, 1995, 26(4): 387–412.
- [2] XIANG Xiaoshu, TIAN Ye, ZHANG Xingyi, et al. A pairwise proximity learning-based ant colony algorithm for dynamic vehicle routing problems[J]. *IEEE transactions on intelligent transportation systems*, 2022, 23(6): 5275–5286.
- [3] XUE Yang, SUN Jianqiao. Solving the path planning problem in mobile robotics with the multi-objective evolutionary algorithm[J]. *Applied sciences*, 2018, 8(9): 1425.
- [4] PARADISO R, ROBERTI R, LAGANÁ D, et al. An exact solution framework for multitrip vehicle-routing problems with time windows[J]. *Operations research*, 2020, 68(1): 180–198.
- [5] CHEN Zhihuan, WU Huaiyu, CHEN Yang, et al. Patrol robot path planning in nuclear power plant using an interval multi-objective particle swarm optimization algorithm[J]. *Applied soft computing*, 2022, 116: 108192.
- [6] 刘长石, 申立智, 盛虎宜, 等. 考虑交通拥堵规避的低碳时变车辆路径问题研究 [J]. *控制与决策*, 2020, 35(10): 2486–2496.
- [7] LIU Changshi, SHEN Lizhi, SHENG Huyi, et al. Research on low-carbon time-dependent vehicle routing problem with traffic congestion avoidance approaches[J]. *Control and decision*, 2020, 35(10): 2486–2496.
- [8] 赵畅, 刘允刚, 陈琳, 等. 面向元启发式算法的多无人机路径规划现状与展望 [J]. *控制与决策*, 2022, 37(5): 1102–1115.
- [9] ZHAO Chang, LIU Yungang, CHEN Lin, et al. Research and development trend of multi-UAV path planning based on metaheuristic algorithm[J]. *Control and decision*, 2022, 37(5): 1102–1115.
- [10] 胡立坤, 王帅军, 吕智林, 等. 移动机器人多目标搜寻的 D^{*}-蚁群融合算法 [J]. *小型微型计算机系统*, 2020, 41(3): 471–476.
- [11] HU Likun, WANG Shuaijun, LYU Zhilin, et al. Multi-objective search based on d^{*}-ant colony fusion algorithm[J]. *Journal of Chinese computer systems*, 2020, 41(3): 471–476.
- [12] 蒲兴成, 李俊杰, 吴慧超, 等. 基于改进粒子群算法的移动机器人多目标点路径规划 [J]. *智能系统学报*, 2017, 12(3): 301–309.
- [13] PU Xingcheng, LI Junjie, WU Huichao, et al. Mobile robot multi-goal path planning using improved particle swarm optimization[J]. *CAAI transactions on intelligent systems*, 2017, 12(3): 301–309.
- [14] SHI Yuhui. Brain Storm Optimization Algorithm[C]//International Conference in Swarm Intelligence. Berlin: Springer, 2011: 303–309.

- [11] 戚远航, 黄子峻, 曾楚祥, 等. 头脑风暴优化算法求解带转角能耗多无人机路径规划问题 [J]. 计算机应用研究, 2022, 39(1): 177–182.
QI Yuanhang, HUANG Zijun, ZENG Chuxiang, et al. Brain storm optimization algorithm for multi-UAV path planning with angular energy consumption[J]. Application research of computers, 2022, 39(1): 177–182.
- [12] 朱代先, 王明博, 刘树林, 等. 基于头脑风暴算法的 Fast-SLAM2.0 算法 [J]. 计算机应用研究, 2021, 38(12): 3629–3633.
ZHU Daixian, WANG Mingbo, LIU Shulin, et al. Fast-SLAM 2.0 algorithm based on brain storm optimization[J]. Application research of computers, 2021, 38(12): 3629–3633.
- [13] YANG Jian, ZHAO Donghui, XIANG Xinhao, et al. Robotic brain storm optimization: a multi-target collaborative searching paradigm for swarm robotics[M]//Lecture Notes in Computer Science. Cham: Springer International Publishing, 2021: 155–167.
- [14] 狄卫民, 杜慧莉, 张鹏阁. 考虑动态拥堵的多车型绿色车辆路径问题优化 [J]. 计算机工程与设计, 2021, 42(9): 2614–2620.
DI Weimin, DU Huili, ZHANG Pengge. Optimization of multi-vehicle green vehicle routing problem considering dynamic congestion[J]. Computer engineering and design, 2021, 42(9): 2614–2620.
- [15] 刘伯运, 赵帅, 赵强强, 等. 水面垃圾清理机器人 [J]. 兵工自动化, 2022, 41(2): 92–96.
LIU Boyun, ZHAO Shuai, ZHAO Qiangqiang, et al. Water surface garbage cleaning robot[J]. Ordnance industry automation, 2022, 41(2): 92–96.
- [16] 周立辉, 张永生, 孙勇, 等. 智能变电站巡检机器人研制及应用 [J]. 电力系统自动化, 2011, 35(19): 85–88, 96.
ZHOU Lihui, ZHANG Yongsheng, SUN Yong, et al. Development and application of equipment inspection robot for smart substations[J]. Automation of electric power systems, 2011, 35(19): 85–88, 96.
- [17] CHENG Shi, SHI Yuhui, QIN Quande, et al. Maintaining population diversity in brain storm optimization algorithm[C]//2014 IEEE Congress on Evolutionary Computation. Beijing: IEEE, 2014: 3230–3237.
- [18] 杨玉婷, 史玉回, 夏顺仁. 基于讨论机制的头脑风暴优化算法 [J]. 浙江大学学报(工学版), 2013, 47(10): 1705–1711, 1746.
YANG Yuting, SHI Yuhui, XIA Shunren. Discussion mechanism based brain storm optimization algorithm[J]. Journal of Zhejiang university (engineering science edition), 2013, 47(10): 1705–1711, 1746.
- [19] NAZARAHARI M. Multi-objective multi-robot path planning in continuous environment using an enhanced genetic algorithm[J]. Expert systems with applications, 2019, 115: 106–120.
- [20] 吴晓涛, 孙增圻. 用遗传算法进行路径规划 [J]. 清华大学学报(自然科学版), 1995, 35(5): 14–19.
WU Xiaotao, SUN Zengqi. Using genetic algorithm for path planning[J]. Journal of Tsinghua University (science and technology edition), 1995, 35(5): 14–19.
- [21] CHICCO G. Overview and performance assessment of the clustering methods for electrical load pattern grouping[J]. Energy, 2012, 42(1): 68–80.
- [22] PONNAMBALAM S G, JAGANNATHAN H, KATARIA M, et al. A TSP-GA multi-objective algorithm for flow-shop scheduling[J]. The international journal of advanced manufacturing technology, 2004, 23(11): 909–915.
- [23] DEEP K. A new crossover operator for real coded genetic algorithms[J]. Applied mathematics and computation, 2007, 188(1): 895–911.
- [24] DUCHOŃ F, BABINEC A, KAJAN M, et al. Path planning with modified a star algorithm for a mobile robot[J]. Procedia engineering, 2014, 96: 59–69.
- [25] 刘晓莹, 蔡自兴, 余伶俐, 等. 一种正交混沌蚁群算法在群机器人任务规划中的应用研究 [J]. 小型微型计算机系统, 2010, 31(1): 164–168.
LIU Xiaoying, CAI Zixing, YU Lingli, et al. An orthogonal-cluster chaos ant colony algorithm based on swarm-robots system mission planning application research[J]. Journal of Chinese computer systems, 2010, 31(1): 164–168.
- [26] WU Changyou, FU Xisong. An agglomerative greedy brain storm optimization algorithm for solving the TSP[J]. IEEE access, 2016, 8: 201606–201621.

作者简介:



陈强, 副教授, 博士生导师, 主要研究方向为伺服系统智能控制、移动机器人调度与控制。获授权发明专利 60 余项, 发表学术论文 100 余篇, 出版英文专著 1 部。



马健, 硕士研究生, 主要研究方向为机器人多目标路径规划。



杨繁, 硕士研究生, 主要研究方向为移动机器人调度与多目标路径规划。