



DOI:10.12404/j.issn.1671-1815.2402048

引用格式:张浩,刘薇.基于改进蚁群算法的机器人路径规划[J].科学技术与工程,2025,25(3):1142-1149.

Zhang Hao, Liu Wei. Robot path planning based on improved ant colony algorithm [J]. Science Technology and Engineering, 2025, 25(3): 1142-1149.

基于改进蚁群算法的机器人路径规划

张浩¹, 刘薇^{1,2*}

(1. 天津工业大学机械工程学院, 天津 300387; 2. 天津市现代机电装备技术重点实验室, 天津 300387)

摘要 针对传统蚁群算法(ant colony algorithm, ACO)在移动机器人路径规划中存在的环境适应性差、拐点个多、计算复杂度高等问题,提出一种基于Sigmoid统计迭代的蚁群算法。首先,采用Sigmoid激活函数分布策略,增加起点到目标点路线上信息素的初始浓度,降低算法前期搜索的盲目性;其次,引入自适应因子动态调节启发函数,增加蚂蚁选择全局最优节点的期望程度,降低算法的收敛时间;最后,在每代蚁群中进行统计分析,提取每代蚂蚁路径最优、最差、平均三个特征参数,并根据迭代次数动态调整信息素更新函数。仿真结果表明,本文改进算法与蚁群系统、精英排序算法、传统蚁群算法相比,最优路径长度分别缩短2.7%、3.2%、5.4%,最优路径次数分别增加42%、53%、62%,最差路径长度分别缩短49%、62%、73%。研究显示,本文改进算法具有更强的全局寻优能力和较好的应用价值。

关键词 蚁群算法; 路径规划; 转移概率; 自适应调整

中图分类号 TP242; 文献标志码 A

Robot Path Planning Based on Improved Ant Colony Algorithm

ZHANG Hao¹, LIU Wei^{1,2*}

(1. School of Mechanical Engineering, Tiangong University, Tianjin 300387, China;

2. Tianjin Key Laboratory of Advanced Mechatronics Equipment Technology, Tianjin 300387, China)

[Abstract] The sigmoid iteration ACO(ant colony algorithm) was optimized for the problems of poor environmental adaptability, high number of inflection points and high computational complexity that exist in the traditional ACO(ant colony algorithm) in route planning. Firstly, the Sigmoid activation function distribution strategy was adopted to improve the initial pheromone spread through the position of the mesh nodes, and the initial concentration of the pheromone was assigned by the sigmoid, which reduced the blindness of the algorithm's pre-search. Secondly, the adaptive factor was introduced to dynamically regulate the heuristic function, which increased the degree of expectation of the ants in choosing the globally optimal node, and reduces the convergence time of the algorithm. Lastly, a statistical analysis was carried out in each generation of the ant, and the three characteristic parameters of ant path optimal, worst and average were extracted in each generation, and the pheromone updating function was dynamically adjusted according to the number of iterations to give full play to the parallelism characteristics of the algorithm. The results prove that the improved algorithm shortens the optimal path length by 2.7%, 3.2%, and 5.4%, reduces the average number of iterations by 42%, 53%, and 62%, and shortens the worst path length by 49%, 62%, and 73%, respectively, when compared with the ant colony system, the elite ranking algorithm, and the traditional ACO. The study prove that the optimized algorithm has stronger global optimality seeking ability and better application value.

[Keywords] ant colony algorithm; path planning; transfer probability; adaptive adjustment

随着人工智能领域的迅速发展,市场对移动机器人的需求不断增加,路径规划是移动机器人研究领域中的重要分支^[1]。良好的路径规划算法不仅能够适应动态变化的环境,还可有效降低能源消耗,提升运行效率^[2]。目前,移动机器人的路径规划问题已经成为国内外学者广泛关注和深入研究的领域。

路径规划技术主要分为传统算法和智能算法,传统算法包括Dijkstra算法、深度优先搜索算法、最小生成树(Kruskal算法)等;智能算法包括蚁群算法、布谷鸟搜索算法、模拟退火算法等^[3]。与传统算法相比,智能算法具有较强的通用性和可扩展性,可以应用于多个领域和场景。

蚁群算法被广泛应用于解决组合优化问题^[4]。

收稿日期:2024-03-22 修订日期:2024-06-05

基金项目:天津市自然科学基金(17JCYBJC19400)

第一作者:张浩(1999—),男,汉族,河北保定人,硕士研究生。研究方向:移动机器人。E-mail:1131766738@qq.com。

*通信作者:刘薇(1973—),女,汉族,天津人,博士,副教授。研究方向:机电一体化。E-mail:tguweiliu@sina.com。

该算法具备多个优点,如正反馈高、自适应强和启发式搜索等特征。然而,在实际应用中,随着问题规模的复杂多样,传统蚁群算法存在着诸多问题,难以满足实际需求。针对这些不足,国内外学者给出了相关改进算法。

文献[5]提出一种蚁群势场融合算法,该算法在信息素更新中引入势力场调节因子和斥力场函数,相较于传统蚁群算法,该方法规划的路径长度明显降低。文献[6]提出了一种局部信息素更新方法,通过降低已访问节点上的信息素含量来激励蚂蚁选择其他节点,从而增加整个群体的多样性,然而,该算法中局部信息素的更新是固定的,并未充分展现蚁群算法作为群智能算法所具备的进化智能特征。文献[7]提出了多种群蚁群算法,该算法通过将部分蚂蚁聚集在起点和终点位置进行协作,从而显著提升了算法的寻优能力和搜索效率。文献[8]在初始阶段采用贪婪策略生成一些次优解,以便在搜索初期就对不同路径上的信息素分布起到引导作用,加快了算法的收敛速度。文献[9]在传统蚁群算法的基础上引入负反馈平衡因子,通过引入不依赖于信息素搜索路径的扩展蚂蚁来改善算法,仿真结果显示平衡因子有效降低了算法后期陷入局部最优的概率。文献[10]提出 ACO&CM 算法,在蚁群算法(ant colony optimization, ACO)的框架中嵌入云模型(cloud model, CM)算法,利用 CM 的随机性降低已访问路径轨迹上信息素含量来提升算法的全局搜索能力。

上述改进算法大多致力于提高蚁群算法的寻优能力和搜索效率,但并没有考虑蚁群算法的进化智能,统计分析各代种群参数以提高收敛速度。为了解决传统蚁群算法存在的不足,将从三个方面对算法进行改进。首先,根据物理环境构建栅格地图,对初始信息素进行不均匀分配,并引入 Sigmoid 激活函数代替传统固定取值方式,以引导算法在初期进行更加有针对性的搜索;其次,对启发函数进行更新,引入自适应因子 u , 并增加目标点的引导,根据迭代次数动态调节启发函数;最后,分析每一代蚂蚁的三个特征参数:即每代蚂蚁路径长度的平均值、最差值、最优值,并在信息素更新函数中加以利用。根据每代蚂蚁结果自适应调整蚁群迭代过程,进而提高算法的全局寻优能力。

1 环境模型构建

1.1 栅格地图

建立环境地图的目的是为了协助机器人更好地理解其所处的环境,并做出相应的决策^[11]。

移动机器人实际的工作空间包括各种不同的地形、障碍物等。为了使机器人能够在物理环境中自主行动和执行任务,需要建立环境地图,并将其转化为一个抽象空间,使其能够从起点到达目标点。环境建模方法主要包括栅格地图法、点云地图法、可视化法和 Voronoi 图法^[12],其中栅格法具有数据结构简单、计算效率高和广泛适用性等特点。为了准确描述移动机器人路径规划过程中的运动状态,本文中采用栅格地图法进行环境建模,如图 1 所示。

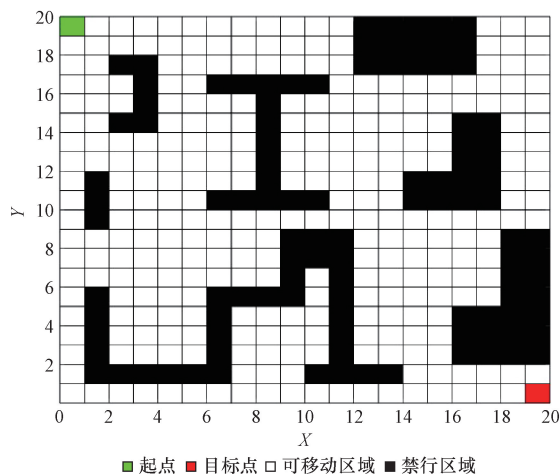


图 1 栅格环境模型

Fig. 1 Grid environment model

根据实际环境地图,对障碍物进行栅格化。在二维平面中,按照从上到下,从左到右的顺序为每个栅格标记一个序号 $i (i = 1, 2, 3, \dots)$, 栅格序号与坐标 (x_i, y_i) 的映射关系表达式为

$$\begin{cases} x_i = \begin{cases} a[\text{mod}(i, L) - 0.5], & \text{mod}(i, L) \neq 0 \\ N_x - 0.5a, & \text{其他} \end{cases} \\ y_i = N_y + 0.5a - \text{ceil}(i, L) \end{cases} \quad (1)$$

式(1)中: a 为栅格地图边长; L 为环境地图的总数; mod 为求余函数; ceil 为取整函数; N_x 为栅格地图行数; N_y 表示栅格地图列数。

1.2 步长搜索方式

相较于传统的四邻域搜索法(仅考虑上、下、左、右四个方向),八邻域搜索法能够扩大搜索空间,有助于移动机器人在搜索过程中避免陷入局部最优解,并进一步提升算法的搜索质量。因此,在本文中采用了八邻域搜索法进行路径规划。

2 传统蚁群算法原理

传统的蚁群算法主要涉及初始信息素分布、蚂

蚁状态转移和信息素更新等关键步骤^[13], 本文研究也主要从这三个方面进行改进。

2.1 初始信息素分布

蚁群算法中的信息素浓度是指蚂蚁在搜索过程中释放在路径上的一种信息素, 用于指导其他蚂蚁选择路径^[14]。在蚁群算法中, 初始信息素浓度的分布规则对算法的性能和收敛速度有着重要影响。传统蚁群算法遵循以下规则。

均匀分布: 初始时将信息素浓度在所有路径上均匀分布, 确保每条路径具有相同的初始信息素浓度, 帮助蚂蚁在开始阶段更均匀地探索搜索空间。

根据经验设定: 根据问题的特点和经验, 设定一个合适的初始信息素浓度值。不同问题可能需要不同的初始化策略, 需要结合具体问题进行调整^[15]。

2.2 状态转移规则

在栅格地图八邻域搜索法中, 由于已走路径放入禁忌表中, 蚂蚁在寻找最优路径时最多有 7 条路径可以选择。为确定下一步栅格节点的选择, 蚂蚁需要考虑栅格节点的信息素浓度和栅格节点之间的距离。信息素浓度代表了路径的优劣程度, 而栅格节点距离则影响了蚂蚁的移动成本。传统蚁群算法通常采用轮盘赌法的转移概率, 状态转移概率公式为

$$P_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}^\alpha(t) \eta_{ij}^\beta(t)}{\sum_{k \in A_k} \tau_{ij}^\alpha(t) \eta_{ij}^\beta(t)}, & j \in A_k \\ 0, & \text{其他} \end{cases} \quad (2)$$

$$\begin{cases} \eta_{ij}(t) = 1/d_{ij} \\ d_{ij} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \end{cases} \quad (3)$$

式(2)中: $\tau_{ij}(t)$ 表示蚂蚁 t 时刻在节点 i 到节点 j 路径上的信息素函数, 在搜索过程中会根据蚂蚁所走的路径质量释放信息素; $\eta_{ij}(t)$ 表示蚂蚁 t 时刻在节点 i 到节点 j 路径上的启发函数, 避免蚂蚁盲目或随机地选择路径; d_{ij} 为节点 i 到节点 j 的欧式距离; α 为信息素启发因子, 反映 $\tau_{ij}(t)$ 的重要性, 用于控制信息素更新的强度; β 为期望启发因子, 反映 $\eta_{ij}(t)$ 的重要性, 用于平衡算法的探索和利用能力; A_k 为栅格 j 的候选集合, 根据蚂蚁的当前位置进行动态更新。

2.3 信息素更新

蚂蚁在觅食路径上会留下信息素, 同时信息素具有一定的挥发性。每轮蚂蚁寻路结束后, 全局信息素的更新公式为

$$\begin{cases} \tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \sum_{k=1}^M \Delta\tau_{ij}^k \\ \Delta\tau_{ij}^k = Q/L_k \end{cases} \quad (4)$$

式(4)中: ρ 为信息素挥发系数, 用于控制信息素的衰减速率; $\Delta\tau_{ij}^k$ 为采用全局更新的蚁周模型; Q 为信息素强度系数; L_k 为蚂蚁 k 的路径长度; M 为每次迭代中参与搜索的蚂蚁总数, 合理选择蚂蚁总数可以增加算法的抗干扰能力。

3 蚁群算法的改进

3.1 Sigmoid 激活函数分布策略

在传统蚁群算法中, 栅格地图节点的初始信息素浓度相等, 这将导致蚂蚁缺乏引导作用, 可能会探索到一些错误路径, 从而降低搜索效率^[16]。为了解决此问题, 引入 Sigmoid 函数初始信息素更新方式, 通过设定非均匀的初始信息素浓度映射至特定区间, 有效地指导蚂蚁在搜索空间中更迅速地收敛至最优解。

Sigmoid 函数是一种常用的非线性激活函数, 如图 2 所示。该函数具有平滑性和适应性的特点, 能够根据具体问题调整信息素的分布策略, 使得蚂蚁在路径选择时能够更好地区分不同路径的优劣, 提高算法的多样性和收敛性。

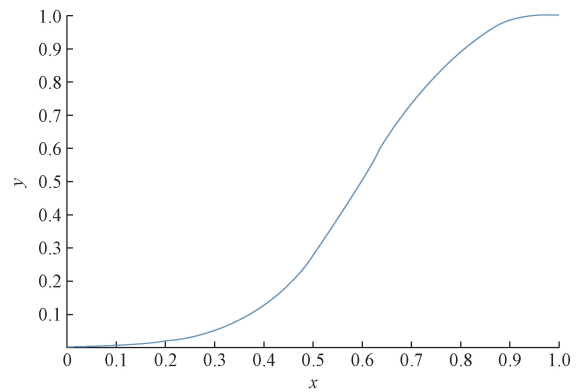


图 2 Sigmoid 函数

Fig. 2 Sigmoid function

Sigmoid 函数将初始信息素浓度从定值转化为与栅格节点位置有关的变量。引导蚂蚁在起始点选择路径时, 更具有多样性和随机性, 有利于探索搜索空间; 在接近目标点选择路径时, 充分发挥信息素的指导作用, 从而提高算法的全局最优解搜索能力。

初始信息素 $\tau(0)$ 的表达式为

$$\tau(0) = S(x) = q_0 \left[1 - \frac{1}{1 + e^{-(x-d_{avg})}} \right] \quad (5)$$

$$d_{avg} = \sqrt{(x_s - x_e)^2 + (y_s - y_e)^2} / 2 \quad (6)$$

式中: q_0 为初始信息素浓度; x 为当前节点 i 到目标节点 j 的欧式距离; d_{avg} 为起始点 (x_s, y_s) 到目标点 (x_e, y_e) 的平均距离。

改进后的算法根据环境地图的规模设定不同的初始信息素浓度 q_0 。分析式(5)可知,当 x 较小时,蚂蚁在路径选择时更加依赖启发函数信息,以避免初始均等信息素对蚂蚁寻优造成的干扰。相反地,当 x 较大时,蚂蚁离目标点越近,此时适度增加目标点附近节点的信息素浓度,有助于提升算法的收敛速度,从而使蚂蚁更快地找到最优解。

3.2 改进距离自适应因子

传统蚁群算法中的启发函数使用栅格节点之间的欧式距离作为更新参数。然而,在八邻域搜索法中,当栅格长度设为1时,启发函数的取值结果仅限于1和 $\sqrt{2}/2$ 两种情况。

在搜索路径的早期阶段,蚂蚁的搜索策略主要依赖于启发函数,加强启发函数的主导作用可以有效地提升算法的收敛速度。然而,当转移概率相近时,会削弱启发函数的效果,从而降低蚂蚁选择路径的多样性,并对算法的全局搜索性能产生不利影响。因此,本文中对启发函数进行改进以增强其指导作用。在蚂蚁进行下一步栅格选择时,除了考虑当前节点到候选节点之间的欧式距离外,还将候选节点到目标点之间的距离也考虑在内,并将其加入到启发函数中,通过增加不同候选节点上启发函数之间的差异,将蚂蚁引导至较短的路径上,改进后的启发函数为

$$\eta_{ij}(t) = \frac{u}{(1-\zeta)d_{ij} + \zeta d_{js}} \quad (7)$$

$$u = 1 - \frac{N}{N_{\max}} \quad (8)$$

式中: d_{ij} 为当前节点 i 到候选节点 j 的欧式距离; d_{js} 为候选节点 j 到目标点 s 的欧式距离; ζ 为距离权重系数,用于调节目标点对启发函数的影响; u 为调节因子; N 为当前迭代次数; N_{\max} 为算法设定的最大迭代次数。

分析式(8)可知,在算法迭代的早期阶段,当 u 接近1时,增强启发函数能够加快算法收敛速度。而在算法迭代的后期阶段,当 u 接近0时,降低启发函数以避免局部最优,提升算法全局搜索能力。

3.3 改进信息素更新规则

传统蚁群算法解的构造主要有两个步骤,个体蚂蚁解的构造和群体间的信息交流^[17]。个体蚂蚁解的构造提高算法的多样性,信息交流帮助整个蚁群找到全局最优解。然而,在实际应用中,这种解法存在一些明显的缺陷。首先,传统蚁群算法在每次迭代中都使用相同的信息素更新方式^[18],无法根据问题的特性和搜索过程的变化进行动态调整。此

外,信息素的更新仅限于当前迭代的局部最优解,忽视了全局最优解的可能性,算法可能会过早地收敛,从而无法找到最优解。

为了更好地优化算法性能,将蚁群系统与精英排序算法相结合进行优化。在每次迭代过程中,不仅考虑当代最优路径,还充分利用本次迭代过程中的最差路径和平均路径进行信息交流。最差路径可提供搜索空间边界信息,而平均路径则可提供全局信息。通过充分利用群体的内部信息,更好地指导蚂蚁在搜索过程中寻找全局最优解。

局部信息素的更新策略表达式为

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \rho \sum_{\varphi=1}^{\sigma} \Delta\tau_{ij}^{\varphi}(t) \quad (9)$$

$$\Delta\tau_{ij}^{\varphi}(t) = \begin{cases} \frac{N}{N_{\max}}\sigma(L_{\min}^{-1} - L_{\max}^{-1}), & (i,j) \in D_s \\ \frac{N}{N_{\max}}\sigma(L^{-1} - L_{\text{ave}}^{-1}), & (i,j) \in D_j \end{cases} \quad (10)$$

式中: ρ 为可调参数; σ 为精英蚂蚁的个数; L_{\min} 为每一代中最佳路径对应的长度; L_{\max} 为每一代蚂蚁路径的平均长度; N 为当前迭代次数; N_{\max} 为最大迭代次数; D_s 为当代最优路径; D_j 为当代较优路径的集合; L 为每一代中较优路径对应的长度。

在全局信息素的更新策略中,引入 ACS 信息素更新策略,根据式(11)进行更新。即

$$\tau_{ij}(t+1) = (1-\alpha)\tau_{ij}(t) + \alpha \sum_{k=1}^{\sigma} \Delta\tau_{ij}^k(t) \quad (11)$$

式(11)中: α 为全局信息素挥发系数; $\Delta\tau_{ij}^k(t)$ 为全局最优蚂蚁对应路径上的信息素更新量,表达式为

$$\Delta\tau_{ij}^k(t) = \begin{cases} \frac{1}{L_c}, & (i,j) \in D_c \\ 0, & \text{其他} \end{cases} \quad (12)$$

式(12)中: L_c 为从路径规划起始至当前搜索到的全局最优路径长度; D_c 为全局最优路径的集合。改进后的算法将根据路径综合信息进行排序。由于排名靠前的蚂蚁更有可能找到更短、更优的路径,所以会对其走过的路径轨迹更新较多的信息素。

3.4 改进蚁群算法的流程

本文改进算法作为一种启发式搜索算法,由于其并行性和分布式计算的特点,在处理大规模、非线性复杂的问题时,可在相对可接受的时间内生成一个较高质量的解决方案。改进蚁群算法的流程如下(图3)。

(1)通过实际环境构建栅格环境地图,设置起点与目标点,并进行参数初始化赋值。

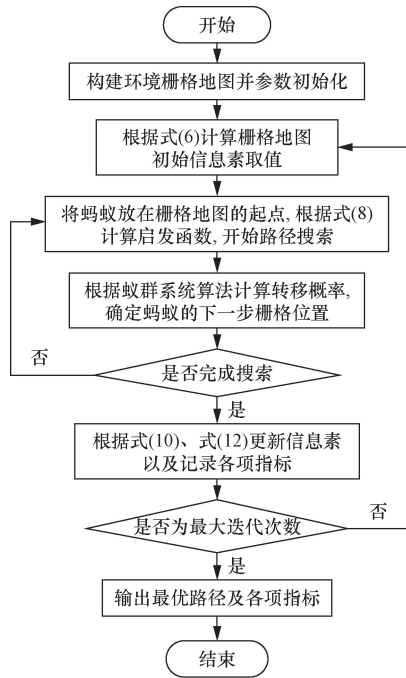


图3 算法流程图
Fig. 3 Algorithm flowchart

- (2) 根据式(5)计算栅格地图初始信息素初始值。
- (3) 将蚂蚁放在栅格地图的起点, 根据式(7)计算启发函数, 开始进行路径搜索。
- (4) 根据蚁群系统算法计算转移概率, 确定蚂蚁的下一步栅格位置。
- (5) 判断蚂蚁是否到达设定的目标点, 若到达, 继续执行(6); 否则返回至(3), 进入下一次循环。
- (6) 根据式(9)、式(11)更新信息素以及记录各项指标。
- (7) 判断是否完成迭代, 若完成, 则输出最优路径, 否则继续执行(3)。

4 实验结果与仿真

为消除栅格地图环境对算法的影响, 分别构建了 20×20 和 30×30 大小的障碍物栅格地图, 并在 Matlab2022a 仿真平台上进行实验。

将本文算法与蚁群系统、精英蚁群算法、传统蚁群算法以及文献[12]方法进行比较, 共进行 50 次仿真实验, 取最优结果进行分析。

4.1 算法参数设置

由于算法参数设置的不确定性, 至今仍未找到最优参数的数学解析方法。因此, 本文中采用控制变量法, 在参数变化范围内设定多种不同的参数组合, 并通过实验结果对比验证最优参数。

在 20×20 和 30×30 的栅格地图下, 每次仅改变单个参数取值, 最终取 50 次仿真结果的平均值, 5 种算法的相关参数设置如表 1 所示。

表 1 相关参数设置

Table 1 Values of relevant parameters

实验参数	参数值
蚂蚁总数 m	50
全局信息素 s	0.2
局部信息素 h	0.2
轮盘赌阈值 o	0.8
启发函数因子 a	5
精英蚂蚁个数 e	5
信息素初始值 q	5
最大迭代次数 k	250
信息素挥发因子 β	1

4.2 20×20 栅格环境试验对比

为了验证算法的有效性, 选择 20×20 的简单栅格环境进行仿真, 并进行 5 种算法的对比实验。图 4 和图 5 为 5 种算法的路径轨迹图和收敛曲线图。

由图 4、图 5 对比分析可知, 在 20×20 栅格环境下的实验中, 当蚂蚁移动到工型障碍物下端时, 其

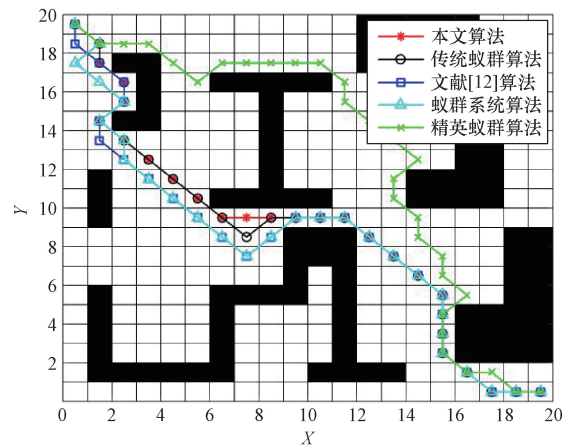


图 4 5 种算法路径图 (20×20)

Fig. 4 5 kinds of algorithm path graph (20×20)

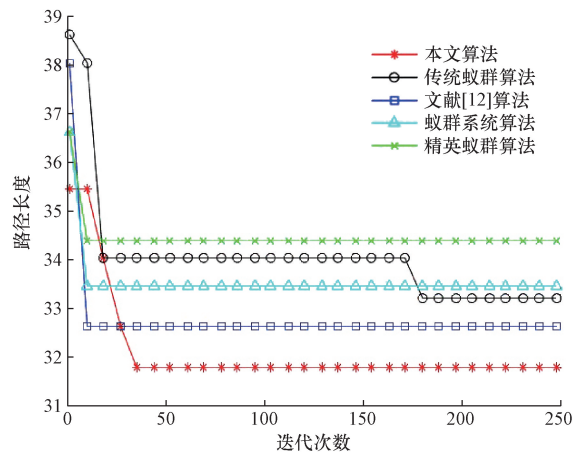


图 5 5 种算法迭代次数图 (20×20)

Fig. 5 5 kinds of algorithm iteration frequency graph (20×20)

他算法的路径图出现路径倒退现象,并且存在较多折线,陷入U形陷阱中未能逃离局部最优解。而本文算法通过引入种群信息交流来调节信息素更新因子,在后期迭代中充分利用了信息素更新机制,逐渐找到有效的路径,并呈现出较为平滑的轨迹,成功逃离了局部最优解。虽然蚁群系统和文献[12]算法具有较少的迭代次数,在迭代初期(10代)即可搜索到全局最优路径,但两者的最优路径长度解质量较差,稍逊于本文算法。

为了进一步比较各个算法的性能,在20×20栅格环境下记录每次算法的实验参数,将不同算法运行50次并取平均值。各个算法的综合性能对比如表2所示。

表2 20×20栅格地图算法性能对比

实验参数	本文算法	蚁群系统	传统蚁群	精英蚁群	文献[12]算法
最优次数	45	21	17	26	22
最优长度	31.871	33.495	33.343	34.343	32.716
最差长度	33.627	35.213	36.041	37.843	36.355
收敛次数	31	13	173	16	13
长度方差	0.523	10.375	15.237	23.843	5.367

从表2数据可知,在20×20栅格环境下的实验中,本文算法在最优次数、最优路径长度以及路径长度方差等参数上表现明显优于蚁群系统和精英蚁群算法;与文献[12]相比,本文算法的最优路径长度降低2.8%、最差路径长度降低8.1%、最优次数增加54%。虽然算法的收敛次数仅优于传统蚁群算法,但本文算法的最优路径长度为全局最优,通过牺牲算法的收敛速度来提升全局最优解,进一步增强了算法鲁棒性,实现综合考虑下的最优路径。

4.3 30×30栅格环境试验对比

第4.2节给出本文算法在简单环境下的有效性,为了验证算法在复杂环境下的可行性,构建了30×30的复杂栅格环境进行仿真,并进行5种算法的对比实验。图6和图7为5种算法的路径轨迹图及收敛曲线图。

通过对比分析图6和图7可知,随着栅格地图规模的增大,障碍物分布更加密集,解空间的扩大显著提高了算法寻找最优解的复杂度。

由于信息素均匀分布策略,传统蚁群算法和蚁群系统在迭代初期生成的路径存在交错现象,且难以达到最优解。虽然文献[12]算法最快找到局部最优解,但算法易出现停滞行为,鲁棒性较弱。

本文中采用Sigmoid激活函数在栅格地图上分布初始信息素,并在启发函数中引入自适应因子 u 和目标点位置。通过采用Sigmoid函数策略,有效

指导蚂蚁的前期搜索,避免算法陷入局部最优解,而启发函数的改进则使蚂蚁始终关注目标点,在复杂环境下进行搜索时能够减少搜索时间、使路径更平滑且转弯次数较少。

根据图7可知,在达到最优解之前本文算法能够保持相对稳定的搜索过程,即在平均迭代次数范围内本文算法能够获得较优的解决方案。相比之下,其他算法由于多样性不足,容易陷入停滞状态且无法找到最优解。

在30×30复杂栅格环境下,各个算法的综合性能对比如表3所示。

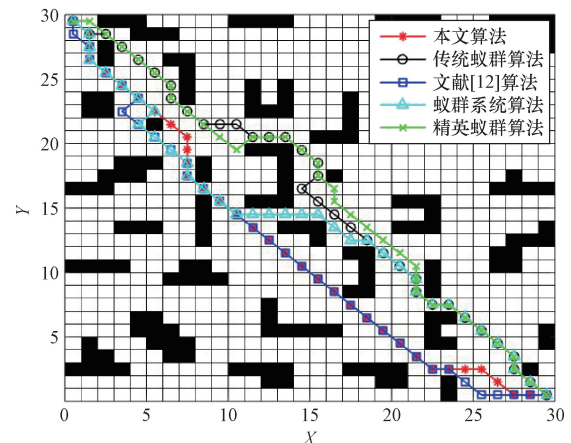


图6 5种算法路径图(30×30)

Fig. 6 5 kinds of algorithm path graph (30×30)

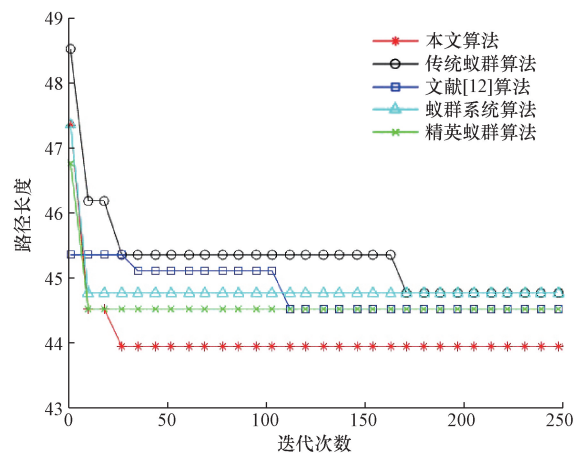


图7 5种算法迭代次数图(30×30)

Fig. 7 5 kinds of algorithm iteration frequency graph (30×30)

表3 30×30栅格地图算法性能对比

实验参数	本文算法	蚁群系统	传统蚁群	精英蚁群	文献[12]算法
最优次数	41	19	16	25	26
最优长度	43.915	44.893	44.893	44.776	44.776
最差长度	45.357	47.318	48.357	46.627	46.017
收敛次数	25	13	174	13	103
长度方差	4.672	13.572	17.621	24.637	8.661

从表3数据可知,在 30×30 复杂栅格环境下的实验中,本文算法的最优路径长度相较于文献[12]算法和蚁群系统分别降低2.7%、3.2%。在迭代次数方面,本文算法平均在第25代时就可以达到收敛,比传统蚁群算法提高85%,比文献[12]算法提高72%,而在长度方差方面,本文算法相较于蚁群系统、传统蚁群算法、精英蚁群算法分别降低65%、73%、81%;在最优次数和最差路径长度方面,本文算法也明显优于其他算法。因此可以得出结论,在复杂障碍物的环境下,本文算法依然能够稳定地得到最优解,可为移动机器人的路径规划问题提供可靠的解决方案。

5 结论

针对传统蚁群算法在路径规划问题中的局限性,基于蚁群系统和排序蚁群算法,在初始信息素分布、启发函数以及信息素更新方面进行了改进,提出了一种优化的蚁群算法。

(1)在初始信息素分布上采用Sigmoid激活函数分布策略,以实现非均匀分布并降低盲目性;改进启发式函数,并引入自适应因子和候选节点到目标点的距离信息,从而增加路径搜索的目标导向性。

(2)通过将蚁群系统和精英排序算法相结合进行优化,结合蚁群群体内部的信息交流,提取每一代蚂蚁路径最优、最差、平均三个特征参数。根据迭代次数动态调节信息素更新规则,以平衡收敛速度与跳出局部最优解之间的关系。

(3)从最优次数,最优路径长度,最差路径长度,平均收敛次数,路径长度方差等指标综合对比其他优化算法,经验证,本文提出的算法在最优路径长度平均降低5%、收敛速度平均增加68%、路径解集方差平均降低73%、具备较高的路径规划有效性和稳定性。

本文研究了已知环境中静态障碍物的路径规划问题。未来,将结合未知环境中的动态障碍物来进行进一步的改进,从而解决更为复杂的路径规划问题。

参 考 文 献

- [1] 杨松,洪涛,朱良宽. 改进蚁群算法的森林防火移动机器人路径规划[J]. 森林工程, 2024, 40(1): 152-159.
Yang Song, Hong Tao, Zhu Liangkuan. Path planning of forest fire prevention mobile robot with improved ant colony algorithm[J]. Forest Engineering, 2019, 40(1): 152-159.
- [2] 杨桂华,符士宾,刘志毅,等. 基于改进蚁群算法的室内移动机器人路径规划[J]. 科学技术与工程, 2019, 19(19): 175-179.
Yang Guihua, Fu Shibin, Liu Zhiyi, et al. Indoor mobile robot path planning based on improved ant colony algorithm[J]. Science Technology and Engineering, 2019, 19(19): 175-179.
- [3] 钱程. 基于边缘计算的动环监管自适应平台研究[D]. 哈尔滨: 哈尔滨理工大学, 2022.
Qian Cheng. Research on adaptive platform of dynamic environment regulation based on edge computing[D]. Harbin: Harbin University of Science and Technology, 2022.
- [4] 刘双双,黄宜庆. 多策略蚁群算法在机器人路径规划中的应用[J]. 计算机工程与应用, 2022, 58(6): 278-286.
Liu Shuangshuang, Huang Yiqing. Application of multi-strategy ant colony algorithm in robot path planning[J]. Computer Engineering and Applications, 2022, 58(6): 278-286.
- [5] 杨宏飞. 基于改进蚁群和人工势场法的无人机三维航迹规划方法[D]. 保定: 河北大学, 2023.
Yang Hongfei. UAV 3D flight path planning method based on improved ant colony and artificial potential field method[D]. Baoding: Hebei University, 2023.
- [6] 邵琪,时维国. 基于改进蚁群算法的机器人路径规划研究[J]. 现代制造工程, 2023(6): 46-51.
Shao Qi, Shi Weiguo. Research on robot path planning based on improved ant colony algorithm[J]. Modern Manufacture Engineering, 2023(6): 46-51.
- [7] 袁汪凰,游晓明,刘升. 动态学习机制的双种群蚁群算法[J]. 计算机科学与探索, 2019, 13(7): 1239-1250.
Yuan Wanghuang, You Xiaoming, Liu Sheng. Two-population ant colony algorithm based on dynamic learning mechanism[J]. Journal of Computer Science and Exploration, 2019, 13(7): 1239-1250.
- [8] 陈颖杰,高茂庭. 基于信息素初始分配和动态更新的蚁群算法[J]. 计算机工程与应用, 2022, 58(2): 95-101.
Chen Yingjie, Gao Maoting. Ant colony algorithm based on initial pheromone allocation and dynamic updating[J]. Computer Engineering and Applications, 2022, 58(2): 95-101.
- [9] Ye K, Zhang C S, Ning J X, et al. Ant-colony algorithm with a strengthened negative-feedback mechanism for constraint-satisfaction problems[J]. Information Sciences, 2017, 406/407: 29-41.
- [10] Zhang X X, Shen X, Yu Z Q. A novel hybrid ant colony optimization for a multicast routing problem[J]. Algorithms, 2019, 12(1): 18.
- [11] 刘羽翀,邓平. 基于改进蚁群算法的路径规划算法研究[J]. 信息技术与信息化, 2023(10): 83-88.
Liu Yuchong, Deng Ping. Research on path planning algorithm based on improved ant colony algorithm[J]. Information Technology and Informatization, 2023(10): 83-88.
- [12] Bullnheimer B, Hartl R F, Strauss C. A new rank based version of the ant system—a computational study[J]. Central European Journal of Operations Research, 1999, 7(1): 25-38.
- [13] 刘璐,沈小伟,葛超,等. 基于改进蚁群算法的植保无人机路径规划[J]. 计算机仿真, 2024, 41(1): 39-43.
Liu Lu, Shen Xiaowei, Ge Chao, et al. Path planning for plant protection UAV based on improved ant colony algorithm[J]. Computer Simulation, 2024, 41(1): 39-43.
- [14] 郝兆明,安平娟,李红岩,等. 增强目标启发信息蚁群算法的移动机器人路径规划[J]. 科学技术与工程, 2023, 23(22): 9585-9591.
Hao Zhaoming, An Pingjuan, Li Hongyan, et al. Path planning

- for mobile robots with enhanced goal-inspired information ant colony algorithm [J]. *Science Technology and Engineering*, 2023, 23(22): 9585-9591.
- [15] Dorigo M, Birattari M, Stutzle T. Ant colony optimization [J]. *IEEE Computational Intelligence Magazine*, 2006, 1(4): 28-39.
- [16] 许明乐, 游晓明, 刘升. 基于统计分析的自适应蚁群算法及应用[J]. *计算机应用与软件*, 2017, 34(7): 204-211.
Xu Mingle, You Xiaoming, Liu Sheng. Adaptive ant colony algorithm and its application based on statistical analysis [J]. *Computer Applications and Software*, 2017, 34(7): 204-211.
- [17] 聂晓雪. 基于链路状态的 SDN 数据中心流量调度算法研究 [D]. 呼和浩特: 内蒙古工业大学, 2021.
Nie Xiaoxue. Research on SDN data center traffic scheduling algorithm based on link state [D]. Hohhot: Inner Mongolia University of Technology, 2021.
- [18] 张真诚. 机器人路径规划的改进粒子群-蚁群算法[J]. *电子测量技术*, 2021, 44(8): 65-69.
Zhang Zhencheng. Improved particle swarm and ant colony algorithm for robot path planning [J]. *Electronic Measurement Technology*, 2021, 44(8): 65-69.