

doi:10.3969/j.issn.1673-9833.2024.04.004

基于改进蚁群算法的智能车路径优化

薛文嘉, 孙 晓, 解玉成, 陈培演, 陈元健, 田 甜

(湖南工业大学 机械工程学院, 湖南 株洲 412007)

摘 要: 用于自动泊车领域的 AGV 小车载质量大, 对移动轨迹的平滑性与行走距离有更高要求。针对传统蚁群算法易死锁、冗余节点多与转向幅度不可控等问题, 提出了一种改进蚁群算法。首先, 在算法正式开始迭代前使用地图补偿函数对地图进行优化, 降低死锁概率; 其次, 在对地图优化处理后, 对地图进行了信息素浓度初始化, 加快了算法收敛速度; 最后, 通过调整路径生成逻辑, 实现算法自适应调整步长, 提高了路径的平滑性, 减少转向摆动。仿真结果表明: 改进后的算法死锁现象减少, 收敛速度更快, 所生成的路径转向平滑, 冗余节点数与总路径长度降低。

关键词: 路径规划; 地图补偿函数; 自适应步长; 蚁群算法; 智能车

中图分类号: TP23

文献标志码: A

文章编号: 1673-9833(2024)04-0020-07

引文格式: 薛文嘉, 孙 晓, 解玉成, 等. 基于改进蚁群算法的智能车路径优化 [J]. 湖南工业大学学报, 2024, 38(4): 20-26.

Intelligent Vehicle Path Optimization Based on an Improved Ant Colony Algorithm

XUE Wenjia, SUN Xiao, XIE Yucheng, CHEN Peiyan, CHEN Yuanjian, TIAN Tian

(College of Mechanical Engineering, Hunan University Technology, Zhuzhou Hunan 412007, China)

Abstract: Due to the fact that AGVs used in the field of automatic parking are of great importance and have higher requirements for the smoothness of movement trajectory and walking distance, an improved ant colony algorithm has thus been proposed in view of such flaws as propensity to deadlock, redundancy of idle nodes, and uncontrollability of the steering amplitude found in traditional ant colony algorithms. Firstly, by using a map compensation function the map is to be optimized prior to the formal iteration of the algorithm, thus reducing the probability of deadlock. Secondly, the pheromone concentration of the map can be initialized after a map optimization, which helps to accelerate the convergence speed of the algorithm. Finally, by adjusting the path generation logic, the step size can be adaptively adjusted by the proposed algorithm, with the path smoothness improved and the steering swinging reduced. The simulation results show that the improved algorithm reduces deadlock occurrence with a faster convergence speed, a smoother generated path steering, a smaller number of idle nodes and a shorter total path length.

Keywords: path planning; map compensation function; adaptive step size; ant colony algorithm; AGV

收稿日期: 2023-05-12

基金项目: 湖南省重点领域研发计划基金资助项目 (2022GK2068)

作者简介: 薛文嘉, 男, 湖南工业大学硕士生, 主要研究方向为机器人控制算法, E-mail: 1478759225@qq.com

通信作者: 孙 晓, 男, 湖南工业大学教授, 硕士生导师, 主要研究方向为机电控制与计算机应用技术,

E-mail: sxbug@163.com

0 引言

随着 AGV (automated guided vehicle) 小车在智能泊车领域的广泛应用,行业对其平稳性和快捷性提出了更高的要求,而智能化的路径规划可以有效提高 AGV 的运行稳定性能和使用寿命、降低运输风险,因此其相关算法已经成为该领域新的研究热点^[1]。目前,AGV 的路径规划多采用标签引导,遇障碍就停止的驱动方式,灵活性较差^[2],在多车运行的区间内容易发生阻塞的情况;而采用对预存储环境信息进行处理后计算导航的驱动方式,也容易出现路径不够平滑,运输不安全的情况^[3]。在智能路径规划算法中,蚁群算法 (ant colony optimization, ACO) 由于具有信息正反馈和启发式搜索的优点^[4],因而在行业中的运用非常广泛,解决了路径规划中的很多难题,但是传统蚁群算法具有易死锁、冗余节点较多与路径质量较差等问题^[5]。针对以上问题,圣文顺等^[6]提出了最大最小蚂蚁系统,不再对每只蚂蚁都进行信息素更新。张志军等^[7]改进了状态转移方程与信息素更新方式,使得算法收敛速度有所提高。于飞等^[8]提出了信息素限定及自适应信息素挥发系数,使得蚁群对地图的探索有了更丰富的可能。

综上所述,为了更好地解决这些问题,提高智能泊车中 AGV 小车运行的稳定性、路径的平滑性,降低小车运行距离,本研究提出了一种改进的蚁群算法。首先,通过地图补偿函数对地图中的障碍物进行有效地处理,使得带有死锁现象的障碍物大量减少,算法的收敛速度得到提高;其次,对地图优化处理后,对地图进行初始化信息素操作,使得算法前期收敛速度得到提高;最后,使用一种自适应补偿的路径更新规则,使得路径中冗余节点数量减少,路径质量得到提高。

1 地图建模

在路径规划前需要对环境进行建模,为 AGV 提供可读取的平面地图,后期配合室内定位算法,实现 AGV 的自主导航^[9]。常见的地图建模方法有栅格法^[10]、拓扑图法^[11]和几何图法^[12],其中,基于栅格的地图建模方式能降低建模的复杂度,减少计算量,是一种被广泛采用的地图建模方法,故本文采用栅格法对机器人的二维环境进行划分。

初始化环境信息后,将地图作为一个平面矩形,切分成 $m \times n$ 个小的单位正方形。广度优先对栅格进行编号,本次研究以 10×10 的栅格环境为例来进行说明,地图栅格化示意图见图 10。

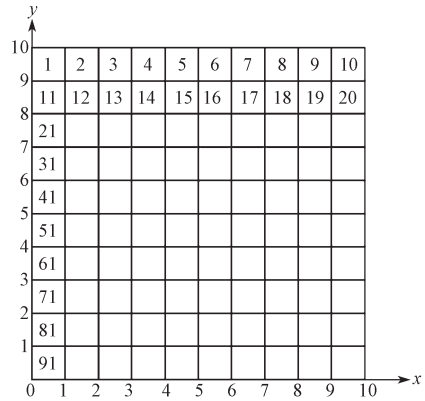


图1 地图栅格化示意图

Fig. 1 Map rasterization diagram

2 经典蚁群算法

2.1 经典蚁群算法描述

蚁群算法是模拟蚁群觅食行为衍生而来的智能算法。其原理为蚂蚁为了聚集群体,在走过的位置会留下信息素,一开始地图上没有信息素,蚁群盲目随机搜索;当第一只蚂蚁找到食物后,会沿着自己走过的路径返回,则其所行路径会覆盖两次信息素,该路径上信息素物质浓度更高;其他蚂蚁会向信息素浓度高的位置聚集,最终信息素浓度越来越高,所有蚂蚁会走在同一条路径上。蚁群算法有两个关键步骤,分别是状态转移规则与信息素更新。

2.2 状态转移规则

蚂蚁后续行动节点的选择依据,是根据信息素的浓度,状态转移规则如式(1)所示:

$$P_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\mu_{ij}(t)]^\beta}{\sum [\tau_{ij}(t)]^\alpha [\mu_{ij}(t)]^\beta}, & j \in allowed_k; \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

式中: P_{ij}^k 为蚂蚁 k 由栅格 i 选择栅格 j 的概率;其中 i 为蚂蚁所在的当前网格编号; j 为接下来可供选择的自由栅格编号; $\tau_{ij}(t)$ 为蚂蚁选择栅格 j 的启发信息; t 为目前迭代次数; α 为启发因子; β 为信息素因子; $\mu_{ij}(t)$ 为路径 i 与 j 间的信息素浓度; $allowed_k$ 为栅格 i 邻域的自由栅格。

2.3 信息素更新规则

在当次算法迭代中,将得到的路径节点存入禁忌表,直至当次迭代结束,将禁忌表清空并进行下一次迭代。在当次迭代结束后,禁忌表中存储所有蚂蚁的所行节点。并对之前迭代的路径降低信息素浓度,对新路径提高信息素浓度,规则如式(2)所示:

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \rho\Delta\tau_{ij}(t). \quad (2)$$

式中: ρ 为信息素挥发系数; $\Delta\tau_{ij}(t)$ 为蚂蚁在本次迭代过程中在路径 $\langle i, j \rangle$ 上的信息素增量, 按照式 (3) 进行计算:

$$\Delta\tau_{ij}(t) = \begin{cases} Q/L^k, & i, j \in L^k; \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

式中: Q 为信息素强度; L^k 为第 k 只蚂蚁在当次迭代中所行轨迹长度。

3 改进的蚁群算法

传统的蚁群算法由于是一种启发式搜索算法, 具有信息正反馈的特点, 在路径寻优中得到了广泛的应用。但是该算法也存在着局部最优、收敛速度慢与路径质量不高等问题, 为了提高蚁群算法的搜索效率和改进路径质量, 本文对传统蚁群算法做了以下改进。

3.1 地图补偿函数

针对传统蚁群算法易陷入死锁与收敛速度慢的问题, 本文提出了地图补偿函数。该函数的原理为在地图建模完成后, 先对地图中的环境信息进行预处理。且先建立死锁障碍库, 将死锁障碍库中的地图模型与实际地图中的障碍物进行匹配, 若判定结果为障碍物, 符合库中待修正地图模型, 则对障碍物进行修正; 否则, 对障碍物不做处理。

地图补偿函数的公式为

$$f(A) = \begin{cases} 1, & A=1; \\ 0, & \text{else.} \end{cases} \quad (4)$$

式中: A 为待判定障碍物与障碍库的比对结果, 若匹配则赋值 1, 否则赋值 0。

栅格进行障碍化处理后如图 2 所示。

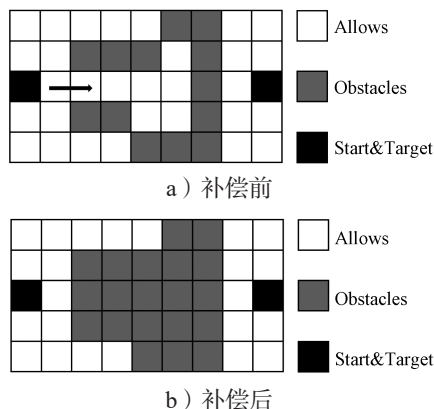


图 2 地图补偿函数示意图

Fig. 2 Schematic diagram of the map compensation function

原地图在凹形障碍物中存在大量的死锁栅格, 通过地图补偿函数处理后, 使得发生可能性大大降低。

3.2 初始化信息素浓度

传统蚁群算法在地图上的初始信息素浓度分布

相同, 此类均匀分布会使蚁群搜索随机性强、盲目性高, 使得算法收敛速度慢。为了防止死锁现象的发生以及加快算法收敛速度, 算法迭代前在地图上进行不均匀浓度信息素释放。由于地图建模时设置左上角为起始点, 右下角为目标点, 而张志军等^[7]发现在地图不同位置转移概率不同, 选择从左上至右下的分布方式更加有利于减少搜索路径长度。在实验中设立了 $(0.5p, 1.5p)$, $(0.6p, 1.4p)$, $(0.7p, 1.3p)$, $(0.8p, 1.2p)$ 4 组不同信息素浓度的对照组实验。在其余条件不变的情况下, 只修改左上与右下初始信息素的浓度, 左下到右上的栅格信息素浓度不变, 独立运行 30 次, 取收敛时间平均值。实验结果如表 1 所示。

表 1 信息素参数设置的影响

(左上 p , 右下 p)	$(0.5p, 1.5p)$	$(0.6p, 1.4p)$	$(0.7p, 1.3p)$	$(0.8p, 1.2p)$
收敛时间 /s	5.021 3	4.823 7	4.972 7	4.890 3

由表 1 可知: 在初始信息素设立为 $(0.6p, 1.4p)$ 时, 算法收敛速度最快。因此本文对地图作出如式 (5) 的信息素浓度初始化。图 3 所示为信息素浓度初始化示意图。

$$p_{i,j} = \begin{cases} 0.6p, & j=i-1, j=i-n, j=i-n-1; \\ p, & j=i-n+1, j=i-n-1; \\ 1.4p, & j=i+1, j=i+n, j=i+n+1. \end{cases} \quad (5)$$

式中: $p_{i,j}$ 为蚂蚁从 i 号栅格到 j 号栅格的初始信息素浓度; n 为所依据地图信息建立栅格地图模型的列数。

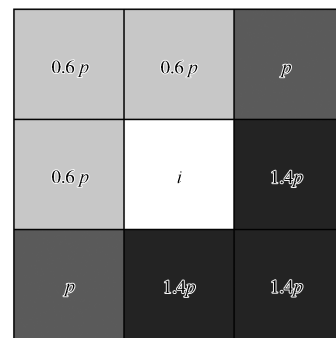


图 3 初始信息素浓度分布示意图

Fig. 3 Schematic diagram of the initial pheromone concentration distribution

为防止错误的引导, 将障碍物栅格的信息素进行剔除。

3.3 全局自适应步长

在传统蚁群算法中, 蚁群常用的搜索模式为四方向搜索和八方向搜索, 八方向搜索由于其易实现、效率高的特点, 在蚁群算法与路径规划等问题上得到了广泛的应用。设定每个栅格的尺寸为 $1\text{ m} \times 1\text{ m}$, 则 AGV 每次实际移动 1 m 或 $\sqrt{2}\text{ m}$, 该搜索方式存

在着步长固定的缺陷。固定步长过大,会导致路径多样性减少,甚至在狭窄地形无法迭代;固定步长过小,则会导致算法收敛时间过长。目前已有的自适应蚁群算法在探索能力方面有一定程度提高,但起始步长仍需手动设置,且最终路径常产生冗余节点。图4为传统蚁群算法搜索模式示意图。

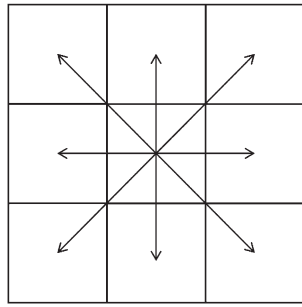


图4 传统蚁群算法搜索模式示意图

Fig. 4 Schematic diagram of the search mode of the traditional ant colony algorithm

本文采用全局自适应步长机制(见图5),最终所形成路径不再局限于四方向和八方向搜索,且路径可自由选择。算法具体执行过程为:算法在迭代指定次数形成最佳路径之后,如图5中 n 虚线所示,在此路径基础上继续优化,对路径节点依次编号,将 n 号节点与 $n+2$ 号节点直接相连,判断两个节点间有无障碍物。若 n 号节点与 $n+2$ 号节点间没有障碍物,则舍去第 $n+1$ 号节点,否则继续迭代至下一个栅格,直至所有 n 号节点与 $n+2$ 号节点间连线都存在障碍物。

由图5结果可知,全局自适应步长算法形成的最终路径拐点显著减少,且路径长度缩短。

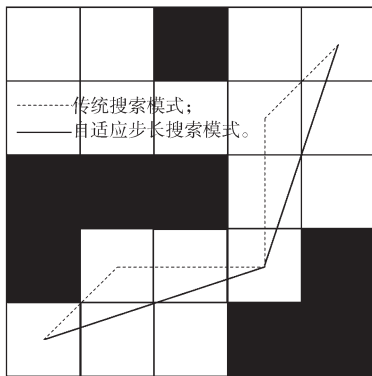


图5 全局自适应步长搜索路径示意图

Fig. 5 Global adaptive step size search path schematic diagram

3.4 算法流程

改进的蚁群算法流程如图6所示。

1) 地图生成。已知环境中的障碍物栅格地图,设置栅格序号。

2) 环境建模与参数初始化。对每个可行节点进行环境信息收集,对各项参数进行设定。

3) 地图栅格化与地图优化。使用地图补偿函数对地图中障碍物进行优化处理。

4) 信息素浓度初始化。

5) 判断蚁群 k 是否到达终点,若到达则进行下

一次迭代, $k=k+1$,对路径上的信息素进行更新;若未达到,则继续更新路径。

6) 判断是否达到所设定最大迭代次数。如果是则进行下一步;否则继续迭代,直至达到指定最大迭代次数。

7) 使用全局自适应步长调节机制对路径进行修正。

8) 输出最终结果。

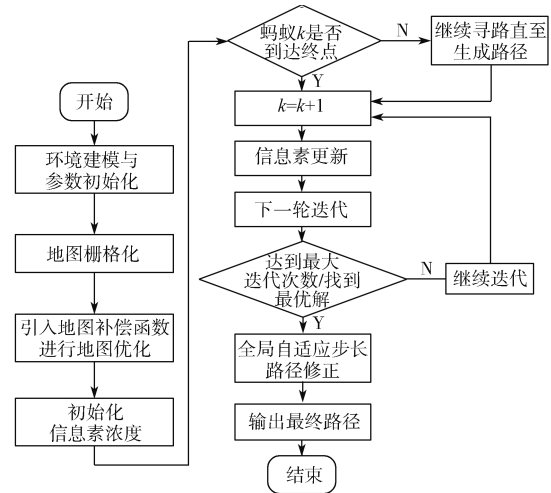


图6 WJACO 算法流程图

Fig. 6 Flow chart of WJACO algorithm

4 仿真结果与分析

为了验证本文算法的有效性与实用性,本研究在 Matlab R2018b 环境下,根据实际地图信息搭建栅格地图,并进行算法仿真及结果比较分析。使用传统蚁群算法 ACO、部分改进后的传统蚁群算法 WJACOTest 和改进蚁群算法 WJACO 在二维静态网格下进行移动机器人路径规划对比实验,设二维栅格地图的单位长度为 1 m。蚁群的初始节点坐标为 (0.5, 19.5),目标节点为 (19.5, 0.5)。本文算法的参数设置如下: $M=100$, $\alpha=1$, $\beta=3$, $\chi=1.2$, $N_{\max}=50$, $\rho=0.8$, $\lambda_1=11.5$, $\lambda_2=5$, $\lambda_3=2$, 以及信息素强度 $Q=100$,产生 100 个个体放入地图中。

4.1 地图补偿函数仿真验证

为了验证地图补偿函数对本文算法的作用,在传统蚁群算法的基础上只加入本文提出的地图补偿函数,并与传统蚁群算法以及 WJACO 进行对比实验,主要是对比以下 3 个方面:一是死锁蚂蚁数量,二是寻找最优路径迭代次数,三是最优路径长度。在 20×20 栅格地图下分别采用 3 种算法进行路径规划仿真试验,仿真次数为 10 次,试验结果如图 7 所示。其中 WJACOTest 为在传统蚁群算法的基础上只加入本文提出的地图补偿函数的算法。

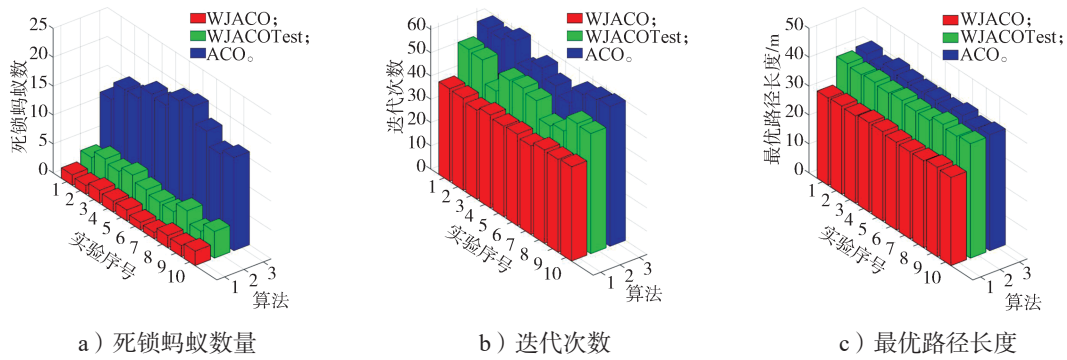


图7 3种算法迭代效果对比图

Fig. 7 Comparison of iteration effects of the three algorithms

由图 7a 可知：本文算法 WJACO 与 WJACOTest 找到的死锁蚂蚁数量平均分别为 1.46 和 4.99，传统蚁群算法 10 次运行后死锁蚂蚁数量为 17.76，WJACOTest 相比于传统蚁群算法死锁蚂蚁数量约减少了 71.9%；由图 7b 可知：本文算法 WJACO 与 WJACOTest 寻找最优路径迭代次数分别为 40.14 和 53.29，传统蚁群算法的平均最优路径迭代次数为 57.06。WJACOTest 相比于传统蚁群算法的最优路径迭代次数约减少了 6.6%；由图 7c 可知：本文算法 WJACO 与 WJACOTest 的最优路径长度分别为 37.15 和 45.12，传统蚁群算法的平均最优路径长度为 47.42。WJACOTest 相比于传统蚁群算法的最优路径长度提升效果不明显。综上所述，采用地图补偿函数对地图进行处理可减少死锁蚂蚁数量，提高算法收敛速度，能够让算法更快地趋于稳定状态。

4.2 信息素初始化仿真验证

为了验证信息素初始化在提高算法收敛速度方面的作用，在传统蚁群算法的基础上只加入本文信息素初始化函数，在 20×20 的栅格地图中进行 10 次仿真试验，并且与本文算法 WJACO 以及传统蚁群算法进行试验对比分析，试验结果如表 2 所示，其中 WJACOTest 为在基本蚁群算法的基础上只加入本文信息素初始化的算法。

表2 3种算法独立运行 30 次的平均结果

Table 2 Average results of 30 independent runs of the three algorithms

算法名称	路径长度 /m	收敛时间 /s	迭代次数
ACO	73.964 4	4.535 5	50.07
WJACOTest	70.431 5	3.985 2	45.29
WJACO	43.351 7	3.657 1	40.14

由表 2 可知：ACO 与 WJACOTest 找到的平均路径长度分别为 73.964 4 和 70.431 5，在路径长度上减少了 4.78%；而 ACO 与 WJACOTest 的收敛时间分别为 4.535 5 s 与 3.985 2 s，WJACOTest 相比于传统蚁群算法寻得路径时间减少了 12.13%；ACO 与 WJACOTest 的迭代次数分别为 50.07 与 45.29，WJACOTest 相比于传统蚁群算法迭代次数减少了 9.54%。综上所述，采用信息素初始化对算法进行优化可以明显减少算法收敛时间，提高算法的收敛速度，能够让算法更快地趋于稳定状态。

4.3 自适应步长函数仿真验证

为了验证自适应步长函数在减少路径围余节点与提高路径质量方面的作用，在传统蚁群算法基础上只加入本文全局自适应步长逻辑，在 20×20 栅格地图中进行 10 次仿真试验，并与本文算法 WJACO 及传统蚁群算法进行对比，试验结果如图 8 所示。

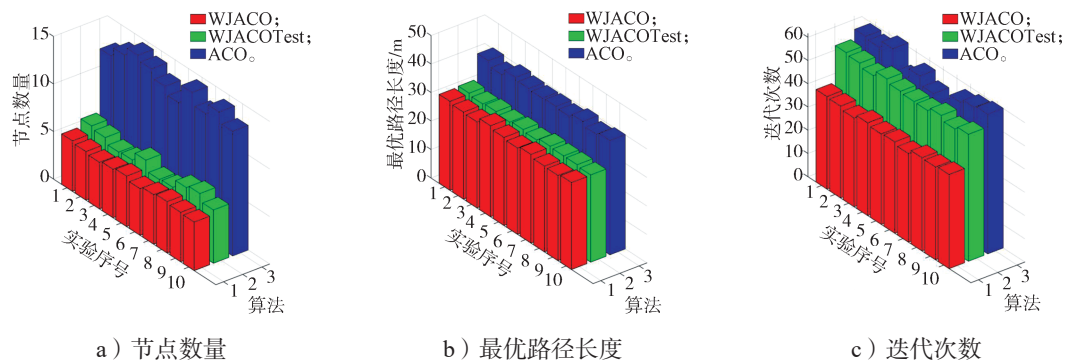


图8 3种算法效果对比图

Fig. 8 Comparison of the effects of the three algorithms

图8中, WJACOTest 为在基本蚁群算法的基础上只加入本文全局自适应步长函数的算法。

由图8a可知: 本文算法 WJACO 与 WJACOTest 找到的平均节点数量分别为 6.1 和 6.3, 传统蚁群算法运行 10 次所得到的平均节点数量为 14.7。WJACOTest 相比于传统蚁群算法平均节点数量减少了 57.14%; 由图8b可知: 本研究的 WJACO 算法与 WJACOTest 算法平均最优路径长度分别为 37.762 5 和 38.963 1, 传统蚁群算法运行 10 次, 平均最优路径长度为 51.398 5, WJACOTest 相比于传统蚁群算法的平均最优路径长度减少了 24.19%。由图8c可知: 本文 WJACO 算法与 WJACOTest 算法的路径寻优平均迭代次数分别为 39.23 和 57.62, 传统蚁群算法的路径寻优平均迭代次数为 58.26, WJACOTest 相比于传统蚁群算法的路径寻优平均迭代次数几乎相同。综上所述, 采用全局自适应步长的方式来选择下

一个可行节点, 能够有效减少拐点数量与路径长度。

4.4 仿真环境下的路径规划试验

为了对比在复杂环境下 WJACO 与传统蚁群算法、精英蚁群算法的差异, 搭建仿真环境为 20×20 的二维静态栅格地图。分别采用 3 种算法进行路径规划仿真试验, 所得结果如图9~10所示。图9为传统蚁群算法、精英蚁群算法与本文算法路径仿真试验图, 图10为传统蚁群算法、精英蚁群算法与本文算法收敛曲线对比图。由图9a~c可知: 在 20×20 静态栅格地图下, 就最优路径而言, WJACO 算法搜索到的最优路径长度最短, 路径转折节点最少; 由图10所示的 3 种算法最优路径收敛曲线可以看出: WJACO 算法的收敛速度比传统蚁群算法和精英蚁群算法的收敛速度更快, 收敛曲线更加平稳, 验证了本研究算法在复杂路况中的实用性。

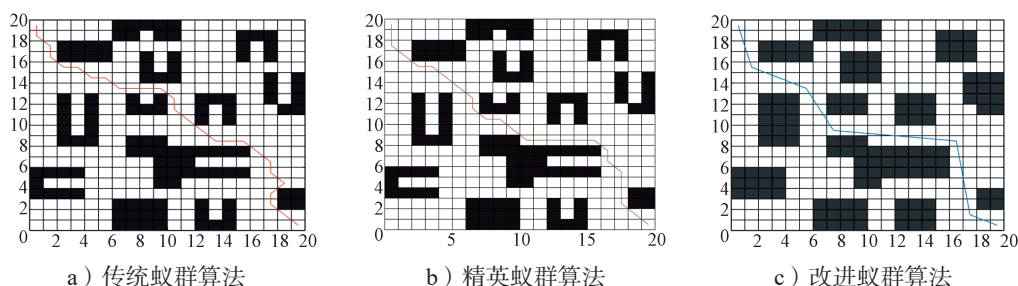


图9 不同算法生成路径对比图

Fig. 9 Comparison of generated paths by different algorithms

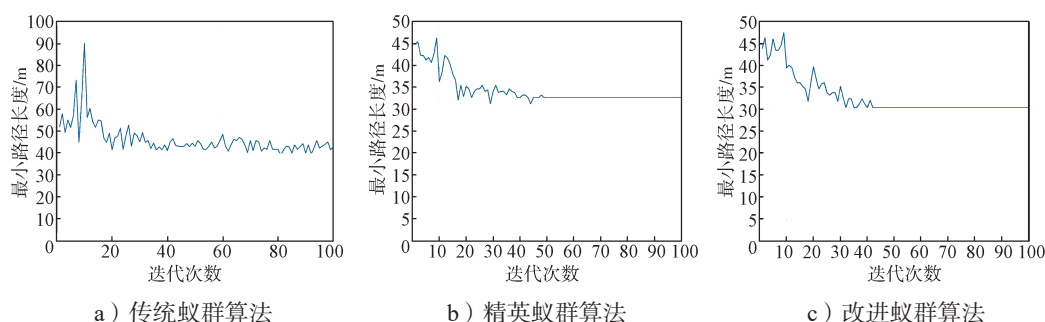


图10 不同算法收敛曲线对比图

Fig. 10 Comparison of convergence curves of different algorithms

5 结语

本文提出了一种改进蚁群算法, 针对传统蚁群算法中出现的一些问题, 提出了相应的改进措施。首先, 针对蚁群算法前期死锁蚂蚁数量多, 易陷入局部最优解的问题, 提出了地图补偿函数, 在算法运行前对地图中的障碍物进行了有效处理, 使得死锁现象减少, 算法收敛速度提高; 其次, 针对前期蚁群算法收敛速度慢的问题, 提出了对地图中进行信息素初始化的方法, 有效提高了算法的收敛速度; 最后, 引入了

全局自适应步长的自我调节机制, 使得算法生成路径中拐点数量和路径长度明显下降。根据实际环境, 使用 Matlab 软件对地图进行栅格地图建模, 并进行仿真实验, 实验结果表明: 改进后的算法收敛速度和路径质量得到提高, 对实际工程具有借鉴意义。

参考文献:

- [1] 周枫林, 陈腾飞, 孙 晓, 等. 基于三维空间曲线的四足机器人轨迹规划 [J]. 湖南工业大学学报, 2023,

- 37(2): 50–55.
ZHOU Fenglin, CHEN Tengfei, SUN Xiao, et al. Trajectory Planning of Quadruped Robots Based on Three-Dimensional Space Curves[J]. Journal of Hunan University of Technology, 2023, 37(2): 50–55.
- [2] 薛晨慷, 李 光, 易 静, 等. 基于 Bezier 曲线的四足机器人 Trot 步态优化 [J]. 湖南工业大学学报, 2022, 36(5): 42–49.
XUE Chenkang, LI Guang, YI Jing, et al. Trot Gait Optimization of Quadruped Robots Based on Bezier Curves[J]. Journal of Hunan University of Technology, 2022, 36(5): 42–49.
- [3] 钱 平, 顾才东, 鲜学丰, 等. 基于改进蚁群算法的水下机器人路径规划研究 [J]. 制造业自动化, 2022, 44(12): 181–184, 208.
QIAN Ping, GU Caidong, XIAN Xuefeng, et al. Path Planning of Underwater Robot Based on Improved Ant Colony Algorithm[J]. Manufacturing Automation, 2022, 44(12): 181–184, 208.
- [4] 宋 宇, 张 浩, 程 超. 基于改进蚁群算法的物流机器人路径规划 [J]. 现代制造工程, 2022(11): 35–40, 47.
SONG Yu, ZHANG Hao, CHENG Chao. Path Planning of Logistics Robot Based on Improved Ant Colony Algorithm[J]. Modern Manufacturing Engineering, 2022(11): 35–40, 47.
- [5] 吴晓雯, 郑巧仙, 徐鑫强. 改进蚁群算法求解多目标单边装配线平衡问题 [J]. 计算机科学, 2022, 49 (增刊 2): 164–168.
WU Xiaowen, ZHENG Qiaoxian, XU Xinqiang. Improved Ant Colony Algorithm for Solving Multi-Objective Unilateral Assembly Line Balancing Problem[J]. Computer Science, 2022, 49(S2): 164–168.
- [6] 圣文顺, 徐爱萍, 徐刘晶. 基于蚁群算法与遗传算法的 TSP 路径规划仿真 [J]. 计算机仿真, 2022, 39(12): 398–402, 412.
SHENG Wenshun, XU Aiping, XU Liujing. Simulation of Traveling Salesman Path Planning Based on Ant Colony Algorithm and Genetic Algorithm[J]. Computer Simulation, 2022, 39(12): 398–402, 412.
- [7] 张志军, 董学平, 甘 敏. 基于优化蚁群算法的 AGV 路径规划研究 [J]. 合肥工业大学学报 (自然科学版), 2022, 45(7): 914–919, 924.
ZHANG Zhijun, DONG Xueping, GAN Min. Research on Path Planning for AGV Based on Improved Ant Colony Algorithm[J]. Journal of Hefei University of Technology (Natural Science), 2022, 45(7): 914–919, 924.
- [8] 于 飞, 卢朝霞. 基于改进蚁群算法的四足巡检机器人全局路径规划方法 [J]. 制造业自动化, 2022, 44(7): 154–157.
YU Fei, LU Zhaoxia. Global Path Planning Method of Quadruped Inspection Robot Based on Improved Ant Colony Algorithm[J]. Manufacturing Automation, 2022, 44(7): 154–157.
- [9] 赵天亮, 张小俊, 张明路, 等. 基于改进融合蚁群算法的机器人路径规划方法研究 [J]. 制造业自动化, 2022, 44(5): 185–190.
ZHAO Tianliang, ZHANG Xiaojun, ZHANG Minglu, et al. Research on Robot Path Planning Method Based on Improved Fusion Ant Colony Algorithm[J]. Manufacturing Automation, 2022, 44(5): 185–190.
- [10] 蒲兴成, 宋欣琳. 分组教学蚁群算法改进及其在机器人路径规划中应用 [J]. 智能系统学报, 2022, 17(4): 764–771.
PU Xingcheng, SONG Xinlin. Improvement of Ant Colony Algorithm in Group Teaching and Its Application in Robot Path Planning[J]. CAAI Transactions on Intelligent Systems, 2022, 17(4): 764–771.
- [11] 张松灿, 普杰信, 司彦娜, 等. 蚁群算法在移动机器人路径规划中的应用综述 [J]. 计算机工程与应用, 2020, 56(8): 10–19.
ZHANG Songcan, PU Jiexin, SI Yanna, et al. Survey on Application of Ant Colony Algorithm in Path Planning of Mobile Robot[J]. Computer Engineering and Applications, 2020, 56(8): 10–19.
- [12] 谢楚政, 李 光, 于权伟, 等. 四足机器人 Trot 步态规划与仿真分析 [J]. 湖南工业大学学报, 2021, 35(5): 32–40.
XIE Chuzheng, LI Guang, YU Quanwei, et al. Trot Gait Planning and Simulation Analysis of Quadruped Robots[J]. Journal of Hunan University of Technology, 2021, 35(5): 32–40.

(责任编辑: 姜利民)