

文章编号: 2095-2163(2022)10-0062-06

中图分类号: TP391.9

文献标志码: A

# 改进蚁群算法在物流机器人路径规划上的研究

于莲芝, 秦天

(上海理工大学 光电信息与计算机工程学院, 上海 200093)

**摘要:** 传统蚁群算法在进行路径规划时存在收敛速度较慢, 容易出现局部最优化, 并且在复杂的环境中算法效率大幅下降的问题, 提出一种优化的蚁群算法。该改进算法在路径搜寻的每一次迭代过程中, 信息素会按改进规则重新分布。改进的算法通过动态调整状态转移概率, 使算法避免停滞, 避免产生局部最优化问题。在解决蚂蚁死锁问题方面提出清除策略, 能够有效加快算法收敛速度, 提高路径规划算法的鲁棒性。从仿真结果可以看出, 相同环境下, 本文改进算法搜索的最优路径长度, 对比基本蚁群算法减少 8.53%, 对比已发表文献算法减少 5.60%, 搜索效率对比基本蚁群算法提升 65.38%, 对比已发表同篇文献中改进算法效率提升 53.63%。

**关键词:** 蚁群算法; 路径规划; 动态更新; 清除策略

## Research on improved ant colony algorithm in the logistics robot path planning

YU Lianzhi, QIN Tian

(School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China)

**【Abstract】** The traditional ant colony algorithm has the problems of slow convergence speed, being prone to local optimization and sharp decline in algorithm efficiency in complex environment. To deal with the problems, An optimized ant colony algorithm is proposed. In each iteration of path search, the pheromone could be redistributed according to the improved rules. The improved algorithm avoids stagnation and local optimization problems by dynamically adjusting the state transition probability. In solving the ant deadlock problem, a clearing strategy is proposed, which can effectively accelerate the convergence speed of the algorithm and improve the robustness of the path planning algorithm. The analysis of the simulation results could be concluded that under the same environment, the optimal path length of the improved algorithm in this paper is reduced by 8.53% compared with the basic ant colony algorithm, 5.60% compared with the algorithm in the published reference, and the search efficiency is increased by 65.38% compared with the basic ant colony algorithm, 53.63% compared with the improved algorithm in the same published reference.

**【Key words】** ant colony algorithm; path planning; dynamic update; clearance strategy

## 0 引言

随着科学技术的发展, 机器人现已广泛应用在仓储物流、现代化农业、智能制造工厂、智慧医疗等领域<sup>[1]</sup>。在此背景下, 移动机器人的路径规划研究即已成为时下学界的关注热点。路径规划是指规定移动机器人在具有障碍物的环境中从初始位置出发寻找一条无碰撞、安全到达目标位置的一条最优路径<sup>[2]</sup>。目前, 国内外已对移动机器人在路径规划的算法方向上进行了大量研究, 最为常见的路径有规划算法 Dijkstra<sup>[3]</sup>, A\* 算法<sup>[4]</sup>等。伴随着该项研究的快速发展, 现已衍生出了一系列的仿生智能算法, 诸如遗传算法<sup>[5]</sup>、粒子群算法<sup>[6]</sup>、蚁群算法<sup>[7]</sup>等。

本文即重点针对蚁群算法展开研究。初期, 是由学者 Dorigo 提出了最早的蚁群算法, 算法是通过模仿蚂蚁觅食行为的仿生研究模拟而来。传统的蚁群算法在进行路径规划时, 通常会出现如收敛速度慢、容易陷入局部最优化等问题。因而, 国内外的众多专家学者都对最早期的蚁群算法进行了研究改进。文献[8]通过建立信息素不均匀分布矩阵, 在目标点和蚁群的初始搜索点之间构建有利矩阵, 目标点和初始点之间的信息素大于其它区域的信息素浓度, 改变状态转移概率, 但是却对环境的要求较高, 目标点和初始点没有明确路径的环境会使得搜索问题变得更为复杂, 对收敛速度和路径缩短方面, 取得成效并不明显。文献[9]先是建立信息素不均匀分

**基金项目:** 国家重点研发计划项目(2018YFB1700902)。

**作者简介:** 于莲芝(1966-), 女, 博士, 副教授, 主要研究方向: 移动机器人技术与控制; 秦天(1996-), 男, 硕士研究生, 主要研究方向: 移动机器人技术与控制、图像处理。

**通讯作者:** 于莲芝 Email: yulianzhi001@163.com

**收稿日期:** 2022-03-09

布矩阵,在目标点和蚁群的初始搜索点间划分出优选区域,形成有利矩阵,将信息素在此区域按照新建立的数学模型重新分布,在前期搜索速度上得到了有效的提升,迭代时间也大大减少,但是建立的数学模型较为复杂,运算量十分可观,运行时间上较传统算法更长,特别是在复杂环境下的算法路径规划能力还有了明显下降。文献[10]采用新的启发函数,把当前所在位置和下一步位置间的距离  $d_{ij}$ 、和下一个将要移动到的位置与目标点之间的距离  $d_{je}$ ,两者和的平方的倒数作为算法的启发函数,这就提升了算法的效率。但却没有考虑到  $d_{ij} \ll d_{je}$ ,而这种情况却容易导致局部最优化问题。文献[11]提出信息素挥发因子自适应策略,在全局搜索能力上得到增强。但在处理局部最优化问题方面却未能给出有效解决方案,同时迭代次数较传统算法也未见到更大改进。文献[12]将传统蚁群算法和人工蜂群算法相结合,将2种算法的信息素浓度赋予不同的权重,得出更新策略,有效解决了局部最优化问题,加快了收敛速度,路径寻优过程中的多样性也得到了保证。但是在一些复杂环境中,容易出现死锁问题,进而容易使算法出现失败。文献[13]引入虚拟节点将搜索空间大大缩小,降低了迭代次数,提升了算法运行的收敛速度。但是设置的准换节点却降低路径寻优的多样性,而且还引入了新的拐点。这些则会直接导致机器人出现安全和功耗问题。

## 1 改进蚁群算法的路径规划方法研究

### 1.1 蚁群算法原理

蚁群算法通过信息素引导整个搜索过程,人工计算机模拟自然界中蚁群的觅食行为是该算法的核心内容。每只蚂蚁都会在路径上留下一种物质,将其称为信息素,当其他蚂蚁稍后经过时,就能够感知到这种物质,以此为向导,对方向选择做出引导<sup>[14]</sup>。一般来说,当每只蚂蚁离开巢穴时,大多都会选择一条通往目的地的路径。在每一个交叉节点上,将前一个蚂蚁释放的信息素作为标记来选择前行路径,最终得到最优路径。

### 1.2 节点状态转移概率

在人工蚁群算法中,当前研究区域蚂蚁数量设置为  $m$ ,当蚂蚁在2个不同的节点间的迁移过程中,两点间路径上的信息素总量对蚂蚁从一个节点到另外一个节点起着决定性的作用。文中,选择某一时间  $t$ ,使用  $p_{uv}^k(t)$  来表示蚂蚁  $k$  从节点  $u$  转移到下一个节点  $v$  概率。 $p_{uv}^k(t)$  由下式计算求出:

$$p_{uv}^k(t) = \frac{[T_{us}(t)]^\alpha \cdot [\eta_{us}(y)]^\beta}{\sum_{s \in \text{Called}_k} [T_{us}(t)]^\alpha \cdot [\eta_{us}(y)]^\beta} \quad (1)$$

其中,  $u, v$  表示当前节点和下一个节点;  $T_{uv}(t)$  表示信息素浓度;  $\eta_{uv}(t)$  表示启发式函数;  $\alpha$  是信息素因素;  $\beta$  是启发式因素。

### 1.3 启发式函数

研究中推得  $\eta_{uv}(t)$  的计算公式如下:

$$\eta_{uv}(t) = \frac{1}{d_{uv}} \quad (2)$$

其中,  $d_{uv}$  表示  $u, v$  节点之间的距离,用于评估路径长度对节点选择的影响程度。

在求解距离的方法中,比较典型的是求解算法。然而,在使用曼哈顿距离求解算法时,不能直接确定对角节点之间的距离。欧氏距离是计算2个节点间的线性距离,可用于对角线节点间的距离计算。因此,本研究选择欧几里得距离法来计算距离。如果2个节点间的距离越大,相应的启发式函数值越小;反之,如果2个节点间的距离越小,相应的启发式函数值就越大,那么从当前节点中选择一个节点的概率就越大。

### 1.4 信息素浓度

信息素浓度由式(3)、式(4)进行计算:

$$T_{uv}(t+n) = (1-\rho) \cdot T_{uv}(t) + \Delta T_{uv}(t) \quad (3)$$

$$T_{uv}(t) = \sum_{k=1}^m \Delta T_{uv}^k(t) \quad (4)$$

式(3)是信息素刷新公式,  $\rho$  表示信息素的波动系数。式(4)中的  $T_{uv}(t)$  为前一时间  $t$  信息素浓度。式(3)中随着  $\rho$  的增加,蚂蚁的信息素挥发越快,直接影响算法的收敛速度;信息素挥发系数  $\rho$  越小,信息素挥发越慢,则会影响整个区域的搜索能力,容易陷入局部搜索。 $(1-\rho)$  表示信息素残留程度。

$\Delta T_{uv}(t)$  是信息素增加量,  $\Delta T_{uv}^k(t)$  是第  $k$  只蚂蚁信息素浓度增量。其值可由如下公式计算求出:

$$\Delta T_{uv}^k(t) = \begin{cases} \frac{Q}{L_k} & \text{第 } k \text{ 只蚂蚁到达点 } (u, v) \\ 0 & \text{第 } k \text{ 只蚂蚁不经过点 } (u, v) \end{cases} \quad (5)$$

其中,  $L_k$  表示蚂蚁行走的距离,即从起始位置到当前位置的距离;  $Q$  是信息素增强系数,在传统的蚁群算法中,  $Q$  表示一个常数。

### 1.5 环境建模

在典型的仓库模型中,包括的主要设备有:货架、传送带、隔离带和拣选台。为了对机器人路径在此环境下进行更好的规划,就要验证机器人改进算

法在较为复杂环境中的可行性,首先就要进行环境地图建模。由于该方法可以在网格环境下方便地进行建模,并具有节省空间的优点<sup>[14]</sup>。图1即显示了构建的光栅环境地图。

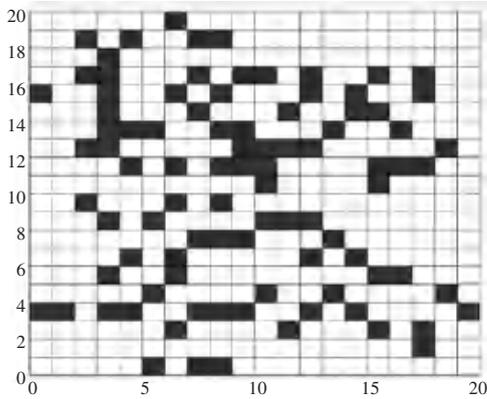


图1 栅格图

Fig. 1 Raster map

移动机器人在执行任务时,以20×20网格地图为例,将其工作区域划分为网格,参见图1。指定机器人在栅格的中心点上移动,栅格坐标由中心点表示。活动区域用白色标示,机器人可以通过;黑色为禁止区域,表示路径上有障碍物。当机器人移动到图形中的网格时,最后一个移动方向被移除,机器人可以向接近其当前位置的任何方向移动(障碍物的方向不能移动),即无法返回。栅格坐标由栅格编号表示,数学公式具体如下:

$$\begin{cases} \dot{x} = \text{mod}(n, M) - 0.5 \\ \dot{y} = M + 0.5 - \text{ceil}(\frac{n}{M}) \end{cases} \quad (6)$$

其中,  $M$  是行数,  $n$  是列数。

## 2 改进的蚁群算法及仿真分析

### 2.1 改进的蚁群算法

基本蚁群算法中,多数算法验证场景设置较为简单,为了解决算法适用性差、容易形成局部最优路径的问题,提出建立信息素浓度动态差异化分布矩阵,使得搜索速率加快,同时缩短搜索路径。改进的算法与传统的蚁群算法最大的区别就在于蚂蚁在搜寻路径时更新信息素的规则。

初始信息素浓度矩阵为  $A_{100 \times 100}$ ,改进后蚂蚁在每一次迭代寻找到目标后,对信息素进行一次动态更新,更新公式如下:

$$y = \psi \frac{1}{\sqrt{2\pi\sigma}} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right] \quad (7)$$

其中,  $\psi$  为平衡系数,  $\sigma$  为方差。蚂蚁在行进

过程中对路径进行不断地优化,由于每次蚂蚁释放的信息素都是按照标准正态分布,距离此蚂蚁越近的栅格获得的信息素浓度就越高,在拐点处按照公式(7)会产生信息素的堆积效应,即在拐弯处,在弧内的信息素浓度堆积会越来越多,而弧外相对于弧内的信息素浓度则会低很多。蚂蚁选择下一个目的地的概率与信息素浓度成正比,如图2所示,会在有弧度的地方进行自动的矫正,因此按照式(7)进行信息素更新,有利于减少发生局部最优化的可能性,缩短搜索路径。

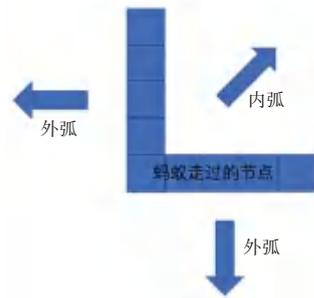


图2 信息素更新示意图

Fig. 2 Pheromone update diagram

### 2.2 死锁问题的处理

蚁群在进行路径寻优的过程中,从当前点转移到下一个栅格时,以当前栅格为中心,除自身所在位置的栅格外,其它栅格不可选,此时蚂蚁就进入死锁状态,会给算法准确性和效率带来很大影响,于是提出了在每次寻找到路径后来重新规划信息素分布的策略。只需要得到一个可以到达目的地的路径,紧接着会自动调整信息素的分布,所以不需要在每次迭代过程中每只蚂蚁都必须到达目的地才会进行下一次迭代,在蚂蚁寻找路径的过程中将陷入死锁状态的蚂蚁直接清除,并将形成死锁的栅格加入禁忌表,这一策略可以有效提高算法的效率。

假设蚂蚁  $i$  在时刻  $t$  进入某个节点,如果按照搜索条件,下一个可选节点集为空,即判断蚂蚁  $i$  进入死锁状态。将此时处于死锁状态的蚂蚁进行直接清除处理,并将当前形成死锁的节点加入禁忌集合 ( $K_n$ ),再将当前蚂蚁爬过节点的信息素清空,将蚂蚁直接清除的策略有效解决了传统蚁群算法中蚂蚁陷入死锁状态的缺点,并且可以显著提高算法效率。

## 3 改进蚁群算法的仿真

### 3.1 改进蚁群算法的算法步骤

**步骤1** 设置初始化参数。在算法中将参数设置为初始值。

**步骤 2** 构建一个环境模型。绘制一个网格地图并将其转换为邻接矩阵。根据起始点和结束点构造信息素矩阵。初始化起始点、爬行路径长度和禁忌列表。

**步骤 3** 搜索路径。将  $M$  只蚂蚁放在起始点, 根据信息素和状态转移概率搜索蚂蚁下一个将要到达的节点, 直至寻找到设置的目标点。当某只蚂蚁陷入死锁陷阱, 则根据清除策略加以处理。

**步骤 4** 更新信息素。根据算法改进的信息素更新策略, 将每个栅格上的信息素按照正态分布进行更新。

**步骤 5** 循环迭代。根据预先设定的迭代次数, 判断在算法运行中是否到达预先设定的迭代次数, 如果达到, 则将当前寻优得到的最短路径输出, 否则算法步骤回调, 继续执行步骤 3, 直至到达最大迭代次数, 输出结果。

综上所述可知, 算法的运算处理流程如图 3 所示。

### 3.2 仿真与分析

基于前文分析, 在  $20 \times 20$  网格环境下进行实验仿真。利用 Matlab 进行栅格建模, 算法参数设置见表 1。实验过程中, 算法改进信息素分布变化动态如图 4 所示。图 4 演示了信息素浓度在拐点的地方进行自我优化调整的过程。对传统的蚁群算法寻优算法、文献[11]提出的蚁群算法的路径规划方法和本文提出的改进蚁群算法进行迭代次数、迭代

时间、稳定性方面的对比, 实验对比结果如图 5、图 6 所示。3 种蚁群算法仿真结果对比见表 2。

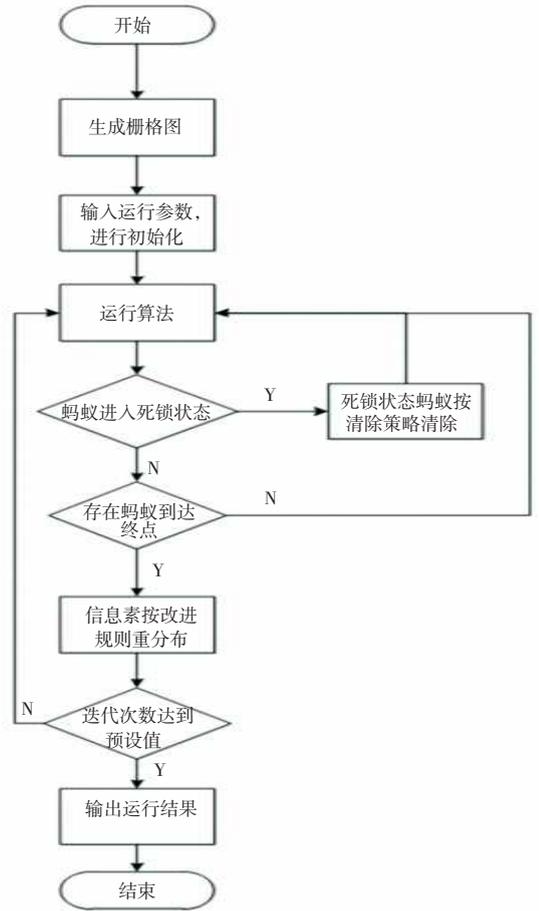


图 3 算法流程图

Fig. 3 The flow chart of the algorithm

表 1 参数设置

Tab. 1 Parameters setting

参数设置	传统蚁群算法	文献[11]算法	改进算法
设置的蚂蚁数量 $M$	50	50	50
信息素启发因子 $\alpha$	1	1	1
启发因子 $\beta$	7	7	7
平衡因子 $\psi$	-	-	300
位置参数 $\mu$	-	-	0
尺度参数 $\sigma$	-	-	1
全局信息素挥发因子 $\rho$	0.3	0.3	$y = \psi \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$
最大迭代次数	100	100	100
信息素增强系数 $Q$	1	1	1

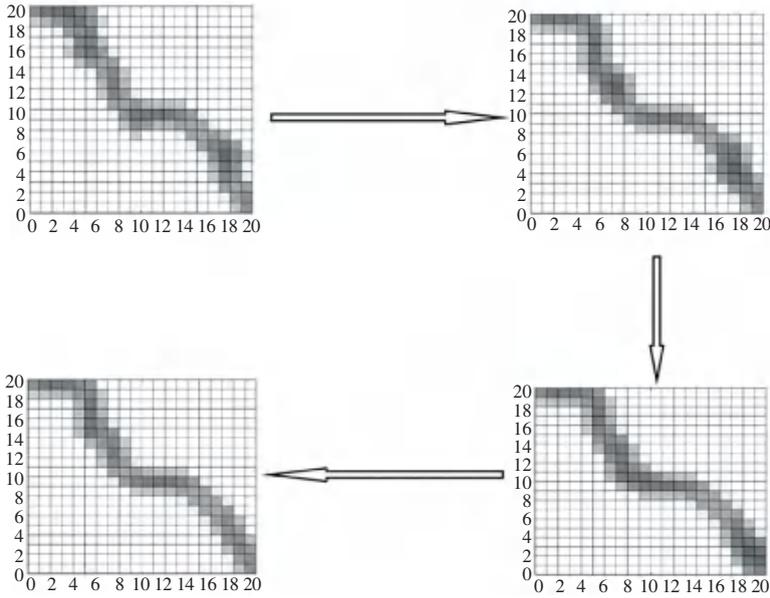


图4 信息素动态分布变化图

Fig. 4 Pheromone dynamic distribution change map

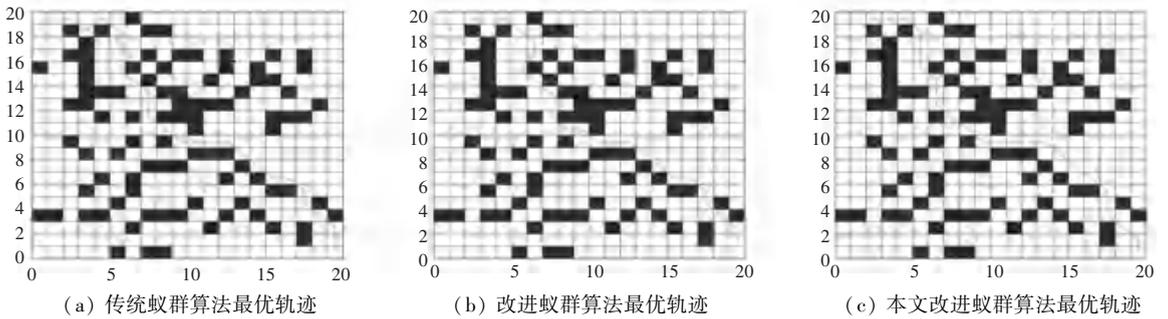


图5 3种蚁群算法最短路径规划图

Fig. 5 Shortest path planning diagram based on three ant colony algorithms

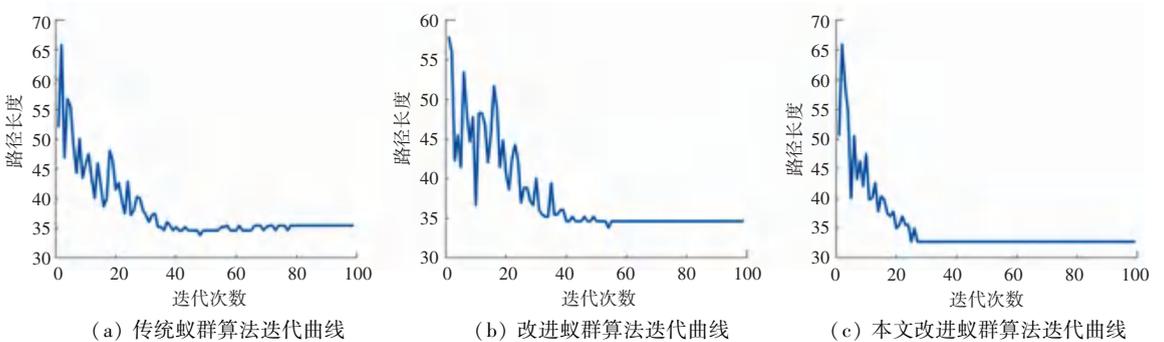


图6 3种蚁群算法路径规划迭代收敛曲线

Fig. 6 Iterative convergence curves of three ant colony algorithms for path planning

表2 3种蚁群算法仿真结果对比

Tab. 2 The simulation results comparison of three ant colony algorithms

算法种类	最短路径长度/cm	迭代耗时/s	最佳迭代次数
传统蚁群算法	35.813 2	187.6	78
文献[11]对比算法	34.701 4	177.6	57
本文算法	32.756 3	145.2	27

在较为复杂、有不规则障碍物的实验环境中对算法的有效性和适用性进行测试。基本蚁群算法、文献[11]蚁群算法和本文提出改进蚁群算法的路径寻优轨迹与迭代收敛曲线见图4、图5。分析可知,传统蚁群算法在收敛速度上较慢,在不规则障碍物处容易发生死锁,搜索结果路径较长,算法收敛性较差。文献[11]的算法也有类似的不足,容易陷入

凹状障碍物,不能够进行自动优化,进而形成局部最优优化。由图 6 可知,传统蚁群算法在路径寻优过程的前期不能够快速地寻找到目标点的方向,收敛速度较慢。文献[11]较基本蚁群算法在收敛速度上有所提高,但对于路径选择也与基本蚁群算法一样未臻优化。由表 2 可以看到,本文改进算法的搜索最优轨迹长度与传统蚁群算法相比减少 8.53%,较文献[11]算法减少 5.60%,搜索效率、即迭代次数减少率较基本蚁群算法减少 65.38%,较文献[11]中改进算法减少 53.63%。根据实验与仿真的结果,在设定的复杂环境中,本文提出的改进的蚁群算法能够有效提升路径的寻优效果。

#### 4 结束语

针对物流机器人在路径寻优过程中收敛速度慢的问题,提出一种基于蚁群算法的改进算法。在路径转移概率中引入信息素高斯分布,可以动态地调整状态转移概率,从而避免了算法中的停滞现象,改进了算法中信息素的更新策略。同时,对搜索过程中出现的死锁问题提出了清除策略。通过仿真结果可以看到,改进算法的迭代次数明显减少,路径长度缩短,搜索路径更为光滑。有效地提高了物流机器人路径规划的速度和性能。

由于本文是基于栅格图进行仿真研究的,在较小的场地环境中效果较好。随着场地的增大,本文算法对全局性最优化问题的解决效果变差。下一步研究将针对这个问题,不断进行完善。

#### 参考文献

- [1] DAI Xiaolin, LONG Shuai, ZHANG Zhiwen, et al. Mobile robot path planning based on ant colony algorithm with A \* heuristic method[J]. *Frontiers in Neurorobotics*, 2019, 13:15.
- [2] BAI Tian, FAN Zhen, LIU Meiqin, et al. Multiple waypoint path planning for a home mobile robot[C]//2018 Ninth International Conference on Intelligent Control and Informati on Processing (ICICIP). Wanzhou, China: IEEE, 2018: 53-58.
- [3] 汤红杰,王鼎,皇攀凌,等. 优化 Dijkstra 算法在工厂内物流 AGV 路径规划的研究[J]. *机械设计与制造*, 2018(S1):117-120.
- [4] 张益辉,王长宁,孙玲. 基于 A \* 算法的机器人路径规划与避障研究[J]. *微型电脑应用*, 2020, 36(02):120-123.
- [5] 卢月品,赵阳,孟跃强,等. 基于改进遗传算法的狭窄空间 路径规划[J]. *计算机应用研究*, 2015, 32(02):413-418.
- [6] 韩明,刘教民,吴朔媚,等. 粒子群优化的移动机器人路径 规划算法[J]. *计算机应用*, 2017, 37(08):2258-2263.
- [7] 江明,王飞,葛愿,等. 基于改进蚁群算法的移动机器人路 径规划研究[J]. *仪器仪表学报*, 2019 40(02):113-121.
- [8] 曹新亮,王智文,冯晶,等. 基于改进蚁群算法的机器人全局 路径规划研究[J]. *计算机工程与科学*, 2020, 42(03):564-570.
- [9] 赵静,汤云峰,蒋国平,等. 基于改进蚁群算法的移动机器人 路径规划[J]. *南京邮电大学学报(自然科学版)*, 2019, 39(06): 73-78.
- [10] 牛龙辉,季野彪. 基于改进蚁群算法的机器人路径规划问题 研究[J]. *微处理机*, 2020, 41(01):37-40.
- [11] 李顺东,游晓明,刘升. 结合 ABC 算法动态分级的双蚁态蚁群 算法[J]. *计算机工程与应用*, 2020, 56(12):37-46.
- [12] 张丽珍,何龙,吴迪,等. 改进型蚁群算法在路径规划中的研究 [J]. *制造业自动化*, 2020, 42(02):55-59.
- [13] 万晓凤,胡伟,方武义,等. 基于改进蚁群算法的机器人路 径规划研究[J]. *计算机工程与应用*, 2014, 50(18):63-66.
- [14] 王影,梁凯,于攀,等. 基于栅格模型与改进蚁群算法的机器人 路径规划研究[J]. *内蒙古科技大学学报*, 2019, 38(04):349-354.
- [15] 潘慧萍,李宝安,张乐,等. 基于多特征融合政府工作报告关 键词提取研究[J/OL]. *数据分析与知识发现*:1-15[2022-02-22]. <http://kns.cnki.net/kcms/detail/10.1478.g2.20220221.0905.002.html>.
- [16] LAMPLE G, BALLESTEROS M, SUBRAMANIAN S, et al. Neural architectures for named entity recognition [C]//Annual Conference of the North American Chapter of the Association for Computational Linguistics; Human Language Technologies(HLT-NAACL). San Diego: ACL, 2016:260-270.
- [17] YAO Yushi, HUANG Zheng. Bi-directional LSTM recurrent neural network for chinese word segmentation[C]// International Conference on Neural Information Processing (ICONIP). Kyoto, Japan: dblp, 2016:345-353.
- [18] LEJUN G, ZHIFEI Z, SHIQI C. Clinical named entity recognition from Chinese electronic medical records based on deep learning pretraining[J]. *Journal of healthcare engineering*, 2020(2):1-8.
- [19] GANCHEV M K, WEISS D. State-of-the-art chinese word segmentation with bi-lstms [C]//Conference on Empirical Methods in Natural Language Processin (EMNLP). Brussels, Belgium: ACL, 2018:4902-4908.
- [20] SUN Zhiqing, SHEN Gehui, DENG Zhihong. A gap-based framework for chinese word segmentation via very deep convolutional networks[J]. *CoRR*, abs/1712.0950, 2017.
- [21] QIU Xipeng, PEI Hengzhi, YAN Hang, et al. A concise model for multicriteria Chinese word segmentation with Transformer encoder [C]// Conference on Empirical Methods in Natural Language Processing (EMNLP). ACL, 2020: 2887-2897.

(上接第 61 页)