

# 自适应蚁群算法的移动机器人路径规划

封声飞<sup>1</sup>, 雷琦<sup>1</sup>, 吴文烈<sup>2</sup>, 宋豫川<sup>1</sup>

1. 重庆大学 机械传动国家重点实验室, 重庆 400030

2. 中船重工(重庆)西南装备研究院有限公司, 重庆 401122

**摘要:**针对传统蚁群算法在路径规划中存在收敛速度和寻优能力不平衡, 算法易陷入局部最优等问题, 提出一种自适应改进蚁群算法。为了提高算法收敛速度, 在栅格环境下, 根据最优路径的特点以及实际环境地图的基本参数, 对初始信息素进行差异化分配; 为了提高蚂蚁搜索效率, 在状态转移概率中引入转角启发信息并对路径启发信息进行改进; 重新制定信息素更新策略, 设定迭代阈值, 调整信息素挥发系数和信息素浓度, 使算法在迭代后期依然具有较强的搜索最优解能力; 采用分段三阶贝塞尔曲线对最优路径进行平滑处理以满足机器人实际运动要求。通过实验仿真与其他算法进行对比分析, 验证了改进算法的可行性、有效性和优越性。

**关键词:**移动机器人; 路径规划; 蚁群算法; 信息素差异化

**文献标志码:**A **中图分类号:**TP242.6 **doi:**10.3778/j.issn.1002-8331.1903-0401

封声飞, 雷琦, 吴文烈, 等. 自适应蚁群算法的移动机器人路径规划. 计算机工程与应用, 2019, 55(17): 35-43.

FENG Shengfei, LEI Qi, WU Wenlie, et al. Mobile robot path planning based on adaptive ant colony algorithm. Computer Engineering and Applications, 2019, 55(17): 35-43.

## Mobile Robot Path Planning Based on Adaptive Ant Colony Algorithm

FENG Shengfei<sup>1</sup>, LEI Qi<sup>1</sup>, WU Wenlie<sup>2</sup>, SONG Yuchuan<sup>1</sup>

1. State Key Laboratory of Mechanical Transmission, Chongqing University, Chongqing 400030, China

2. China Shipbuilding Heavy Industry(Chongqing) Southwest Equipment Research Institute Co. Ltd., Chongqing 401122, China

**Abstract:** As for original ant colony algorithm using in mobile robot path planning, there are problems such as the convergence speed and searching ability existing unbalance, falling into local optimum easily and so on. In this paper, an adaptive improved ant colony algorithm is proposed. Firstly, in order to improve convergence speed of the algorithm, in the grid environment the initial pheromone is differentiated distributed according to the characteristics of the optimal path and the basic parameters of the actual environment. Secondly, in order to raise the efficiency of ants search, the turn-angle heuristic information is introduced into the state transition probability and the path heuristic information is improved. Thirdly, it reformulates the strategy of pheromone renewal, sets the iteration threshold, and the pheromone volatilization coefficients and pheromone concentration are adjusted, so that at the later stage of iteration, the algorithm still has a strong ability to search for the optimal solution. Finally, the piecewise third-order B-spline curve is used to smooth the optimal path to meet the actual motion requirements of the robot. The feasibility, effectiveness and superiority of the improved algorithm are verified by comparing with other algorithms.

**Key words:** mobile robot; path planning; ant colony algorithm; pheromone differentiation

## 1 引言

移动机器人路径规划问题是机器人研究领域中的

一个热点, 也是机器人的控制基础, 要求机器人依据一定的标准(如时间最短、距离最短、能耗最低等), 在所

**基金项目:**国家自然科学基金(No.51205429); 国家工信部绿色制造系统集成项目(No.CCLS-JB-002); 国家工信部智能制造综合标准化与新模式应用项目。

**作者简介:**封声飞(1994—), 男, 硕士, 研究领域为移动机器人(AGV)路径规划与调度, E-mail: 284410986@qq.com; 雷琦(1976—), 女, 博士, 副教授, 研究领域为智能制造及其装备, 生产管理。

**收稿日期:**2019-03-26 **修回日期:**2019-05-21 **文章编号:**1002-8331(2019)17-0035-09

**网络出版:**2019-05-31, <http://kns.cnki.net/kcms/detail/11.2127.TP.20190530.1447.028.html>

©1994-2021 China Academic Journal Electronic Publishing House. All rights reserved. <http://www.cnki.net>

给定的工作环境中搜索出一条最优或者接近最优的安全路径。目前,很多传统算法<sup>[2-4]</sup>应用在移动机器人路径规划中。随着研究的深入,一些仿生智能优化算法<sup>[5-8]</sup>逐渐应用于路径规划研究中。仿生智能优化算法相比传统算法在解决复杂环境中的路径规划问题更有优势。

蚁群算法(Ant Colony Optimization, ACO)<sup>[9]</sup>是一种正反馈群智能优化算法,具有并行性、强鲁棒性、适应性、易与其他算法相结合等特点。传统蚁群算法最初应用于解决TSP问题<sup>[10]</sup>,后来逐渐应用于车间调度<sup>[11-12]</sup>和机器人路径规划问题<sup>[13]</sup>,但在求解路径时存在搜索效率低、参数较多、易出现算法停滞和陷入局部最优解等缺点。对此学者们对蚁群算法进行了改进。在信息素更新机制上,文献[14]在信息素更新时,使得最优路径上的信息素增加,最差路径上的信息素进行减少,在一定程度上提高了收敛速度,但在迭代后期各条路径上信息素浓度差异增加,算法易陷入局部最优;文献[15]采用随机值对信息素初始化,并对信息素进行交叉操作,提高了算法搜索能力,但降低了算法的收敛速度;在调整路径搜索策略上,文献[16]采用多策略进化机制并自适应调整进化选择概率,提高了算法的随机性,但当环境规模增加时,算法搜索最优解的能力明显降低;文献[17]提出了一种自适应搜索半径蚁群算法,根据环境的复杂程度自动改变寻优半径,进行最优局部最优点的获取,具有良好的动态规划能力,但路径全局性有待进一步改善,文献[18]利用细菌觅食算法来优化蚁群算法的参数,通过调整参数组合来优化算法整体性能,但只针对某些固定场景,不具备一般性。文献[19]利用改进人工势场算法所得路径进行信息素初始化,使得蚂蚁搜索具有方向性,但在迭代后期算法容易陷入停滞状态。由上述情况以及目前国内外对蚁群算法的研究现状可知,很少对算法的收敛速度和寻优能力进行协调平衡。

综上所述,为了蚂蚁快速找到最优解,同时协调算法收敛速度和寻优能力,在保证算法快速收敛的同时依然具有较强的搜索能力,本文提出一种自适应蚁群算法(Adaptive Ant Colony Optimization, AACO)。首先,为提高算法的收敛速度,在迭代初期获得一个较小的全局解,对初始信息素浓度根据起始点和目标点位置进行差异化分配。其次,对状态转移概率、信息素更新策略进行自适应改进,在状态转移概率中引入转角启发信息并改进路径启发信息,提高算法效率和路径平滑性。再次,为了避免算法在迭代后期陷入停滞状态,当算法迭代次数达到设定的阈值时,重新调节信息素挥发系数和信息素浓度,避免算法陷入局部最优。最后,应用分段三阶Bezier曲线对最优路径进行平滑处理。

## 2 栅格法环境建模

移动机器人的工作环境为二维静态环境,假设障碍

物高度忽略不计且静止。由于栅格法简单有效,对障碍物的适应能力强,可大大减少环境建模的复杂性,因而本文采用栅格法划分工作环境,无障碍物的栅格为自由栅格,在仿真程序中用0表示,有障碍物的栅格为不可行栅格,在仿真程序中用1表示,为了防止机器人与障碍物碰撞,将障碍物适当膨化,采用序号法和二维直角坐标相结合来对栅格进行标识<sup>[20]</sup>。如图1所示,以10×10的栅格环境为例来说明。

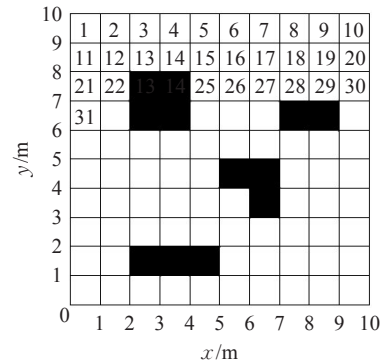


图1 栅格坐标与编号关系

如图1所示,空白栅格表示自由栅格,黑色栅格表示障碍物栅格,地图按从左到右,从上到下的顺序依次编号为1、2、...,每个栅格对应一个编号,坐标与栅格编号的关系如式(1):

$$\begin{cases} x_i = \text{mod}(N_i/N) - 0.5 \\ y_i = N - \text{ceil}(N_i/N) + 0.5 \end{cases} \quad (1)$$

式中,mod为取余运算,ceil表示向后取整数, $P_i$ 表示栅格编号, $P$ 表示每一列的栅格数量,所求坐标为每个栅格的中心点位置。

通过栅格法对机器人运行环境的建模划分,路径规划问题变成了根据起始点 $S$ 和目标点 $E$ ,利用算法在自由栅格集合中寻找有序的栅格子集,这些有序栅格中心连线即为算法规划的路径。

## 3 蚁群算法基本原理

蚁群算法模拟了自然界中蚂蚁的觅食行为,蚂蚁在寻找食物过程中会释放信息素,同时能够感知前面蚂蚁在路径上残留的信息素浓度来引导自己运动,因而逐渐形成一种正反馈现象:路径越短,残留信息素浓度越高;越多的蚂蚁经过该路径,该路径上的信息素浓度就越大,之后的蚂蚁选择这条路径的概率就越大,蚂蚁就是通过这种交流方式来找到食物<sup>[21]</sup>,蚁群算法中的两个关键部分就是路径选择概率和信息素更新,决定了蚁群算法的求解速度和解的质量。

### 3.1 转移概率

在静态已知环境中,在 $t$ 时刻蚂蚁 $m(m=1, 2, \dots, M)$ 在点 $i$ 向点 $j$ 移动是由转移概率 $p_{ij}^m(t)$ 决定,先根据式(2)计算点 $i$ 附近可行路径的 $p_{ij}^m(t)$ 值,再根据轮盘赌

法则选择下一段最佳路径。

$$p_{ij}^m(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta}{\sum_{j \in allowed} [\tau_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta}, & j \in allowed \\ 0, & j \notin allowed \end{cases} \quad (2)$$

式中,  $\tau_{ij}(t)$  为  $t$  时刻路径  $(i, j)$  上的信息素浓度,  $\eta_{ij}(t)$  为  $t$  时刻点  $i$  到点  $j$  的路径启发信息, 其值为  $1/d_{ij}$ ,  $d_{ij}$  为点  $i$  到点  $j$  的距离,  $\alpha$  为信息素浓度因子,  $\beta$  为距离启发信息强度因子,  $allowed$  为下一步可以选择的节点集合。

### 3.2 信息素更新

信息素的更新分为局部更新和全局更新, 本文选择全局更新, 即当所有蚂蚁完成一次循环搜索时再进行信息素更新<sup>[22]</sup>。随着时间的推移, 路径上的信息素会挥发一部分, 蚂蚁经过的路径会释放一定浓度的信息素, 在所有蚂蚁完成一次迭代后按照下式对信息素进行更新:

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \rho\Delta\tau_{ij}(t, t+1) \quad (3)$$

$$\Delta\tau_{ij}(t, t+1) = \sum_{m=1}^M \Delta\tau_{ij}^m(t, t+1) \quad (4)$$

式(3)中  $\rho$  为信息素挥发系数,  $1-\rho$  为信息素残留因子,  $\Delta\tau_{ij}(t, t+1)$  为本次循环中路径  $(i, j)$  上信息素的增量; 式(4)中  $M$  为蚂蚁总数,  $\Delta\tau_{ij}^m(t, t+1)$  为第  $m$  只蚂蚁在路径  $(i, j)$  上释放的信息素浓度, 对于其值, 文献[13]和文献[14]提出了3种信息素更新策略, 分别为: 蚁周模型(Ant-cycle)、蚁量模型(Ant-quantity)、蚁密模型(Ant-density), 为了提高蚁群的全局搜索能力, 本文采用Ant-cycle模型, 如式(5):

$$\Delta\tau_{ij}^m(t, t+1) = \begin{cases} Q/L_k, & \text{蚂蚁 } m \text{ 经过路径 } (i, j) \\ 0, & \text{其他} \end{cases} \quad (5)$$

式中,  $Q$  为常量, 表示信息素强度初值,  $L_k$  表示迭代次数为  $k$  时的全局最优解。

## 4 蚁群算法的自适应改进

算法陷入局部最优主要是由于非最优路径上的信息素浓度较大造成的。为了防止算法陷入局部最优, 本文对蚁群算法的改进采取以下措施: 在信息素初始化时, 在部分最优路径上信息素进行差化设置, 防止非最优路径上的信息素对蚂蚁产生误导。在信息素更新时, 选出路径较优的部分蚂蚁, 只针对这部分蚂蚁进行信息素更新, 避免了较差蚂蚁路径对下一代蚂蚁的影响。此外设定迭代阈值, 当迭代次数达到阈值时, 对路径上的信息素浓度和信息素挥发系数进行重新调整, 减小路径上的信息素浓度差异, 使陷入局部最优的蚂蚁可以快速跳出局部最优, 以下是具体改进方法。

### 4.1 初始信息素差异化分配原则

对于传统蚁群算法初始信息素浓度均匀分配所导致的算法迭代初期搜索盲目性强、收敛速度慢等问题, 在之前的研究中, 学者们利用其他算法提前进行路径规划得到一条粗略路径, 在信息素初始化时, 在此条路径与其他路径上的信息素进行差异化分配<sup>[23-24]</sup>。蚂蚁根据路径上信息素浓度的差异来寻找最优路径, 提高了算法的收敛速度。该方法具有强启发性, 强启发性表示具有一条从起始点到终点的完整路径信息对蚂蚁进行引导, 由于该条路径的信息素启发信息不一定是最优的, 会对蚂蚁产生误导, 使算法陷入局部影响解的质量。

为了保证差异化的信息素在最优路径上, 减少错误的启发信息对蚂蚁的误导, 提高蚂蚁寻优能力。本文提出一种对蚂蚁具有弱启发性的初始信息素差异化规则, 弱启发性表示对部分路段进行信息素差异化, 这部份路径是可确定为最优路径的一部分。由于最优路径多集中在起始点  $S$  和目标点  $E$  的连线  $L$  附近区域, 同时受环境中障碍物的影响。在  $t$  时刻, 蚁群算法在选择下一点  $j$  时, 根据点  $j$  到直线  $L$  的距离以及地图尺寸和环境中障碍物占比, 对该区域的初始信息素进行差异化增量, 使得初始信息素浓度的差异化更加符合实际求解问题。以上表达用数学公式表示如下:

$$\tau_{ij}^0(t) = \begin{cases} \tau_{ij}^0(t) + \lambda \cdot [1 - (dist/dm)], & dist < dm \\ \tau_{ij}^0(t), & dist \geq dm \end{cases} \quad (6)$$

$$dm = (\xi \cdot l) / 4 \quad (7)$$

式(6)中  $\tau_{ij}^0(t)$  表示  $t$  时刻点  $i$  到点  $j$  的初始信息素浓度,  $dist$  表示点  $j$  到起始点  $S$  和目标点  $E$  连线  $L$  的距离,  $dm$  表示实际地图环境综合评价,  $\lambda$  是与地图尺寸有关的一个常数, 其值在  $0.1/\omega \pm 0.03$  的范围内,  $\omega$  为地图尺寸中长或宽的最大值; 式(7)中  $\xi$  表示地图中障碍物占比, 为障碍物栅格与栅格总数的比值,  $l$  表示地图中任意两点间的最长距离, 矩形地图为对角线长度。

### 4.2 转移概率的改进

在机器人路径规划的过程中应该尽量避免不必要的路径转折以提高运动的平稳性, 本文在计算路径转移概率时引入转角启发信息, 减少路径中大转角和转角次数。改进后的路径转移概率如下:

$$p_{ij}^m(t) = \begin{cases} \frac{[\tau'_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta [\omega_{ij}(t)]^\sigma}{\sum_{j \in allowed} [\tau'_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta [\omega_{ij}(t)]^\sigma}, & j \in allowed \\ 0, & j \notin allowed \end{cases} \quad (8)$$

式(8)中  $\omega_{ij}(t)$  表示  $t$  时刻点  $i$  到点  $j$  的转角启发函数,  $\sigma$  表示转角启发信息因子, 其值为正整数。如图2所示, 蚂蚁在栅格中行走转角  $\theta$  只能是  $0^\circ$ 、 $45^\circ$ 、 $90^\circ$ 、 $135^\circ$  四种情况, 根据角度大小对机器人行走的影响程度设定对应的影响权值  $a$ , 如式(9), 转角启发函数  $\omega_{ij}(t)$  与  $a$



成反比,如式(10):

$$a = \begin{cases} 1, \theta = 0^\circ \\ 2, \theta = 135^\circ \\ 3, \theta = 90^\circ \\ 4, \theta = 45^\circ \end{cases} \quad (9)$$

$$\omega_{ij}(t) = 1/a \quad (10)$$

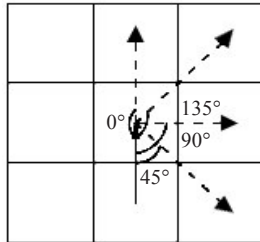


图2 路径转折角度

传统蚁群算法的启发函数启发性不够强,受A\*算法的启发<sup>[25]</sup>,本文采用A\*算法中的估计代价函数作为蚁群算法的启发函数,即 $t$ 时刻点 $i$ 到点 $j$ 的启发信息为 $j$ 点到目标点 $E$ 的欧氏距离,如下式:

$$\eta_{ij}(t) = \sqrt{(x_j - x_E)^2 + (y_j - y_E)^2} \quad (11)$$

式(11)中 $(x_j, y_j)$ 、 $(x_E, y_E)$ 分别表示点 $j$ 和目标点 $E$ 的坐标。

#### 4.3 改进信息素更新规则

在传统蚁群算法中,对最差蚂蚁进行信息素更新过后,会对下一代产生误导。为了协调收敛速度和算法全局搜索能力,解决解的多样性和信息素更新之间的矛盾,本文采用部分蚂蚁更新信息素的方法,即在所有蚂蚁完成一次迭代后,对搜索到路径的蚂蚁根据路径长度从小到大排序。选择排序在前面部分的蚂蚁进行信息素更新,被选择更新的每只蚂蚁在每条路径上信息素增量 $\Delta\tau_{ij}^m(t, t+1)$ 相同,对后面部分的蚂蚁不进行信息素更新,如式(12):

$$\Delta\tau_{ij}^m(t, t+1) = \begin{cases} \frac{Q}{l_{\text{gbest}}} \cdot (l_{\text{worst}} - l_{\text{best}}) \cdot (1 - \epsilon), & \text{rank} \leq \kappa \\ 0, & \text{rank} > \kappa \end{cases} \quad (12)$$

$$\kappa = \mu \cdot M \quad (13)$$

式(12)中 $l_{\text{gbest}}$ 表示当前全局最优解,  $l_{\text{worst}}$ 表示当前迭代产生的局部最差值,  $l_{\text{best}}$ 表示当前迭代产生的最优值,  $\epsilon$ 表示蚁群进化率,  $\epsilon = k/K_{\text{max}}$ ,  $n$ 为当前迭代次数,  $K_{\text{max}}$ 为最大迭代次数;  $\text{rank}$ 为排序后的蚂蚁序号,式(13)中 $\kappa$ 表示需要更新信息素的蚂蚁数量。  $\mu$ 表示要更新信息素蚂蚁的比例,  $\mu \in (0, 1)$ ,  $M$ 表示蚂蚁的总数。

随着路径上信息素的积累,蚂蚁倾向于选择信息素浓度最大的路径,此时算法不再出现新解,为了增强算法在迭代后期的寻优能力,扩大解搜索空间,需要对算法进行扰动。根据算法陷入局部最优的最大迭代次数设定迭代阈值 $R$ ,本文规定算法连续20次未出现新的

最优解表明算法陷入局部最优状态,算法陷入局部最优的迭代次数与问题的规模和环境的复杂程度有关,为了保证该策略的通用性,  $R$ 的取值应该尽量大一些。当算法迭代次数达到 $R$ 时,减小各条路径上信息素差异,增大信息素挥发系数具体见式(14),使得算法在后期依然具有较强的全局搜索能力。

$$\tau_{ij}^R = 2 \cdot \tau_{ij}^0, \quad \rho' = 1.5\rho, \quad \text{if}(N=R) \quad (14)$$

式中,  $\tau_{ij}^R$ 表示当迭代次数为 $R$ 时的信息素浓度,  $\tau_{ij}^0$ 表示初始信息素浓度。

#### 4.4 分段三阶Bezier曲线平滑路径

在栅格地图中路径规划的结果是由栅格中心坐标连接而成的一条折线,折线在转弯处存在尖峰,不利于机器人的行走。这也是栅格法进行路径规划的一个缺陷,为了弥补栅格法的不足,让算法得到的路径对实际机器人运动更有引导作用,需要对尖峰进行平滑处理。

Bezier曲线是能够描述复杂情况的曲线,根据折线上的关键点,通过Bezier公式逼近该折线,从而得到一条与折线走势相同的平滑曲线。如图3所示,其中折线上的点称为控制点,连接的多边形称为控制多边形<sup>[26]</sup>。

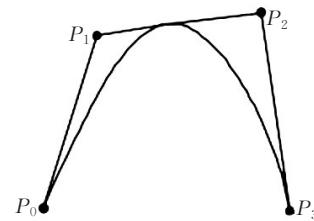


图3 Bezier平滑曲线

在空间 $n+1$ 点的位置 $p_i(i=0, 1, 2, \dots, n)$ ,则 $n$ 次Bezier曲线描述为:

$$B(T) = \sum_{i=0}^n C_n^i T^i (1-T)^{n-i} P_i, \quad T \in [0, 1] \quad (15)$$

式中,  $P_i$ 代表Bezier曲线的控制点,  $T$ 表示Bezier曲线控制参数,本文采用三阶Bezier曲线平滑路径,公式表示为:

$$B(T) = P_0(1-T)^3 + 3P_1T(1-T)^2 + 3P_2T^2(1-T) + P_3T^3 \quad (16)$$

由于在机器人路径中的折点数 $N > 4$ ,因而需要对路径折线分段采用贝塞尔曲线平滑处理,然后再拼接起来,拼接处应该满足连续性,前一点与后一点在拼接处的曲率相等<sup>[27]</sup>。任意点处的曲率为:

$$K(T) = \frac{x'(T)B'(T) - B'(T)x''(T)}{(x^2(T) + B^2(T))^{3/2}} \quad (17)$$

#### 4.5 改进蚁群算法执行步骤

步骤1 基本参数初始化,建立栅格地图,初始化参数:信息素浓度因子 $\alpha$ ,启发信息强度因子 $\beta$ ,转角权重因子 $\delta$ ,信息素初始强度值 $Q$ ,信息素挥发系数 $\rho$ ,需要更新信息素的蚂蚁比例 $\mu$ ,最大迭代次数 $K_{\text{max}}$ ,蚂蚁

总数  $M$ 。

**步骤2** 初始信息素不均匀分布,首先对信息素均匀分配得到  $\tau_{ij}^0(t)$ ,再根据式(6)和(7)对地图中任意可行点进行信息素差异化分布得到  $\tau'_{ij}(t)$ 。

**步骤3** 蚂蚁路径选择,将  $M$  只蚂蚁放在起始点  $S$  处,并把  $S$  加入禁忌表中,根据式(8)至(11)计算每只蚂蚁的状态转移概率,按照轮盘赌法则选择下一点可行点  $j$ 。

**步骤4** 判断点  $j$  是否为终点坐标  $E$ ,若是,则执行步骤5,否则将  $j$  加入禁忌表,再次执行步骤3,直到找到目标点为止。

**步骤5** 信息素全局更新,当所有的蚂蚁完成一次循环迭代后,对所有完成路径搜索的蚂蚁按照路径长度由小到大排序,按式(13)计算需要更新信息素的蚂蚁数量,再按照式(12)对所选蚂蚁进行信息素更新。

**步骤6** 算法停滞处理,当迭代次数  $k$  达到所设定迭代阈值  $R$  时,根据式(14)对路径上初始信息素浓度和信息素挥发系数进行调整。

**步骤7** 算法全局最优结果,判断迭代次数是否达到最大迭代次数  $K_{\max}$ ,若达到,则输出全局最优结果,输出路径长度,否则循环执行步骤(3)到(7)。

**步骤8** 路径平滑处理,对步骤7所得到的全局最优路径采用分段三阶Bezier曲线平滑处理,最后输出处理后的路径轨迹,算法结束。

## 5 实验仿真与分析

为了验证改进蚁群算法的可行性、稳定性和优越性,在MATLAB 2014a上进行仿真实验,运行环境为:Win10(64 bit)操作系统,Core™i3-8100处理器,8 GB运行内存。本文从以下几个方面进行实验验证:(1)优化引入参数。(2)通过在不同的栅格地图尺寸和环境下,应用本文改进蚁群算法与传统蚁群算法、文献[15]和文献[16]进行对比分析验证改进算法的优越性。(3)改进算法在迭代中解的多样性进行分析。(4)应用三阶分段Bezier曲线平滑处理改进蚁群算法搜索的最优路径。(5)算法实用性分析。

### 5.1 优化引入参数

在本次实验中根据经验对蚁群算法的基本参数初始化为:信息素浓度因子  $\alpha=1$ ,距离启发信息强度因子  $\beta=5$ ,信息素强度  $Q=3$ ,信息素挥发系数  $\rho=0.3$ ,最大迭代次数  $K_{\max}=100$ ,蚂蚁总数  $M=50$ ,现对本文引入的参数:转角启发信息强度因子  $\delta$ ,迭代阈值  $R$ ,更新信息素的蚂蚁比例  $\mu$  进行优化组合,首先设定各个参数范围:  $\mu \in \{0.3, 0.4, 0.5, 0.6, 0.7\}$ ,  $\delta \in \{1, 2, 3, 4, 5\}$ ,  $R \in \{30, 40, 50, 60, 70\}$ 。参数默认值设为  $\mu=0.3$ ,  $\delta=1$ ,  $R=30$ 。每次实验只改变一个参数,其他参数取默认

值,为了让选择的参数值环境适应性更强,本次实验在  $50 \times 50$  复杂环境中(如图4,绿色栅格为起点,红色栅格为终点)进行且对每个参数的仿真进行10次实验取均值。由表1参数优化结果可知:  $\mu$  在0.6附近最优,  $\delta$  在3附近最优,  $R$  在50附近最优,综上本文引入参数的取值为  $\mu=0.6$ ,  $\delta=3$ ,  $R=50$ 。

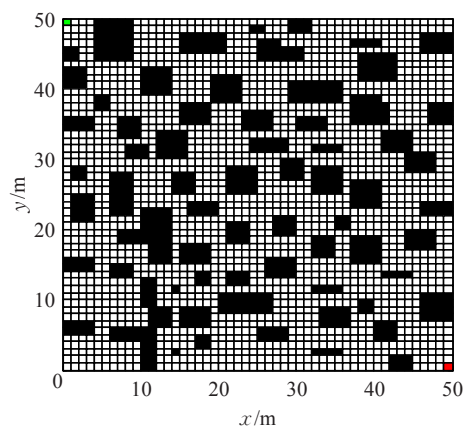


图4  $50 \times 50$  复杂环境

表1 本文引入参数优化结果

$\mu$	0.3	0.4	0.5	0.6	0.7
路径长度	81.498	80.569	78.912	77.740	79.154
$\delta$	1	2	3	4	5
路径长度	81.498	81.983	80.569	82.2254	81.740
$R$	30	40	50	60	70
路径长度	81.498	81.397	73.740	83.397	81.255

### 5.2 算法对比分析

蚁群算法是一种随机搜索算法,判断算法的优越性和稳定性可以依据算法运行一次得到最优解的可能性大小来说明。在本实验环节分别采用两种尺寸和障碍复杂程度都不同的栅格环境进行实验,在每种地图环境中对传统蚁群算法、文献[15]、文献[16]和本文改进的蚁群算法各运行20次进行分析。

#### 5.2.1 $20 \times 20$ 简单环境下的实验仿真

在本次实验中,四种算法在该地图中搜索出的路径结果如图5,算法收敛曲线比较如图6。图中绿色方格表示路径起始点  $S$ ,红色栅格表示路径目标点  $E$ 。

由图5和图6知,传统蚁群算法陷入了局部最优,主要原因是在路径搜索过程中,算法迭代初期信息素均匀分配,信息素的正反馈作用不强,路径启发信息  $\eta_{ij}(t)$  为当前点到下一点的距离,路径选择倾向于选择  $\eta_{ij}(t)$  值小的路段,启发信息方向性较弱,因而路径转折点较多、路径较长。随着算法的迭代,信息素在该路径上的积累,后代蚂蚁倾向于选择此路径,因而无法跳出局部最优,文献[15]和文献[16]以及本文改进的算法都对初始信息素浓度采用了不同方法进行不均匀分配。算法在迭代初期就有良好的正反馈,使得算法在初期就具有一

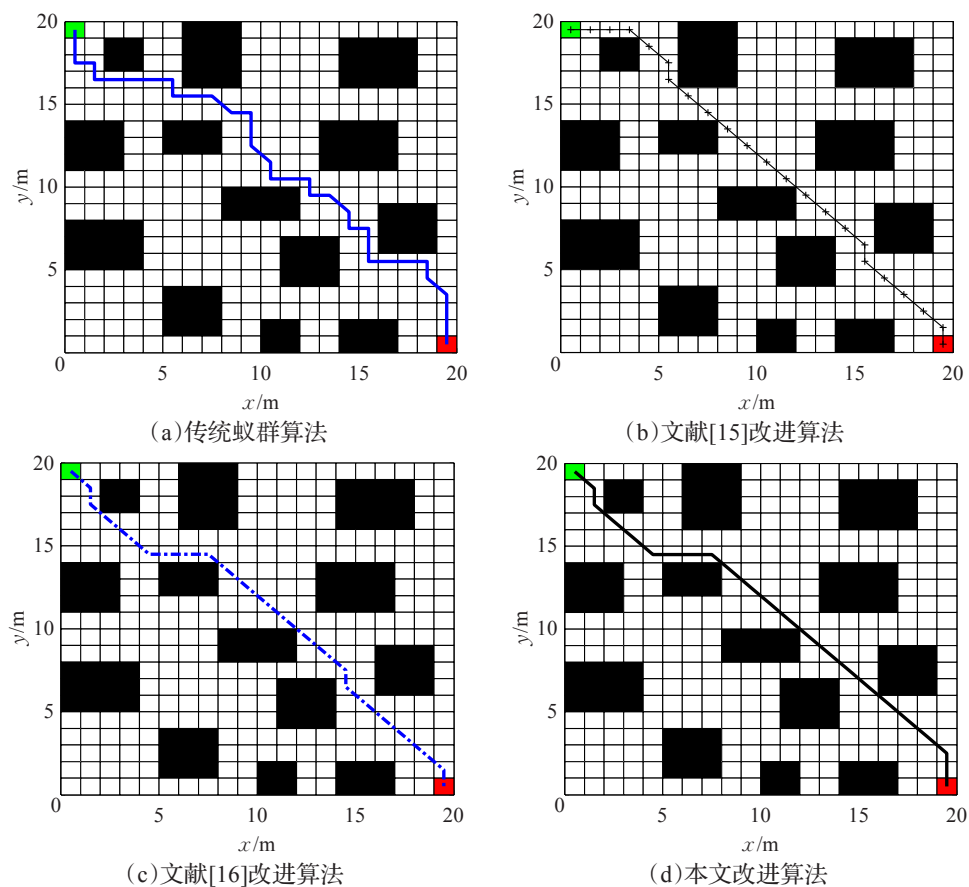


图5 20×20环境下四种算法的规划路径结果

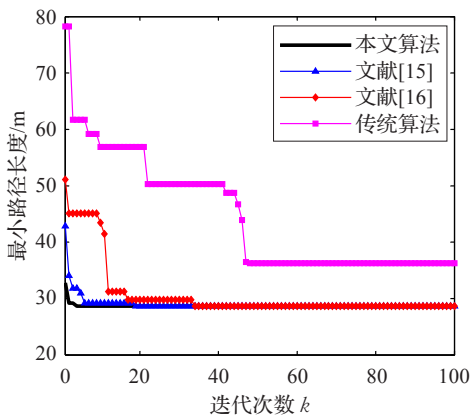


图6 20×20环境下四种算法收敛曲线

个较小的初值,提高了算法的收敛速度,同时能够获得算法最优解。

由表2可知,文献[15]和文献[16]以及本文改进的算法都能够获得最优解,在求解质量上差别不大,但本文改进算法的收敛速度明显比文献[15]和文献[16]收敛速

度快,平均收敛代数5,根据迭代标准差可以看出收敛速度比其他三种算法稳定。由此可得,在20×20的简单环境中本文改进的蚁群算法能够保证快速收敛的同时寻找到最优解。

5.2.2 30×30复杂环境下的实验仿真

为了验证在复杂环境中改进算法的适应性和有效性。本次采用30×30复杂环境地图进行实验,四种算法的实验结果如图7所示。

由图7和图8知,由于地图尺寸和环境复杂度的增加,四种算法的迭代次数都有所增加,但传统的蚁群算法依然陷入了局部最优。文献[15]在信息素更新时引入当前最优最差解和信息素交叉操作。在算法能搜索到最优解的同时提高了算法的收敛速度,得到最优解时的迭代次数为31。在文献[16]中,由于选择进化概率随着迭代自适应变化,在迭代后期算法随机性增强,收敛速度慢,收敛到最优结果的迭代次数为82。本文改进的算法在迭代初期得到较小的初始值,说明该算法在路径

表2 20×20环境中四种算法仿真结果

地图环境	算法	路径长度				迭代次数			
		最优解	最大值	平均值	标准差	最小值	最大值	平均值	标准差
20×20简单环境	传统ACO	35.071 1	38.000 0	36.813 3	1.140	9	50	25	11.351
	文献[15]	28.627 4	29.799 0	28.766 8	0.316	3	23	11	5.733
	文献[16]	28.627 4	28.627 4	28.627 4	0	20	55	31	9.366
	本文算法	28.627 4	28.627 4	28.627 4	0	3	7	5	1.337

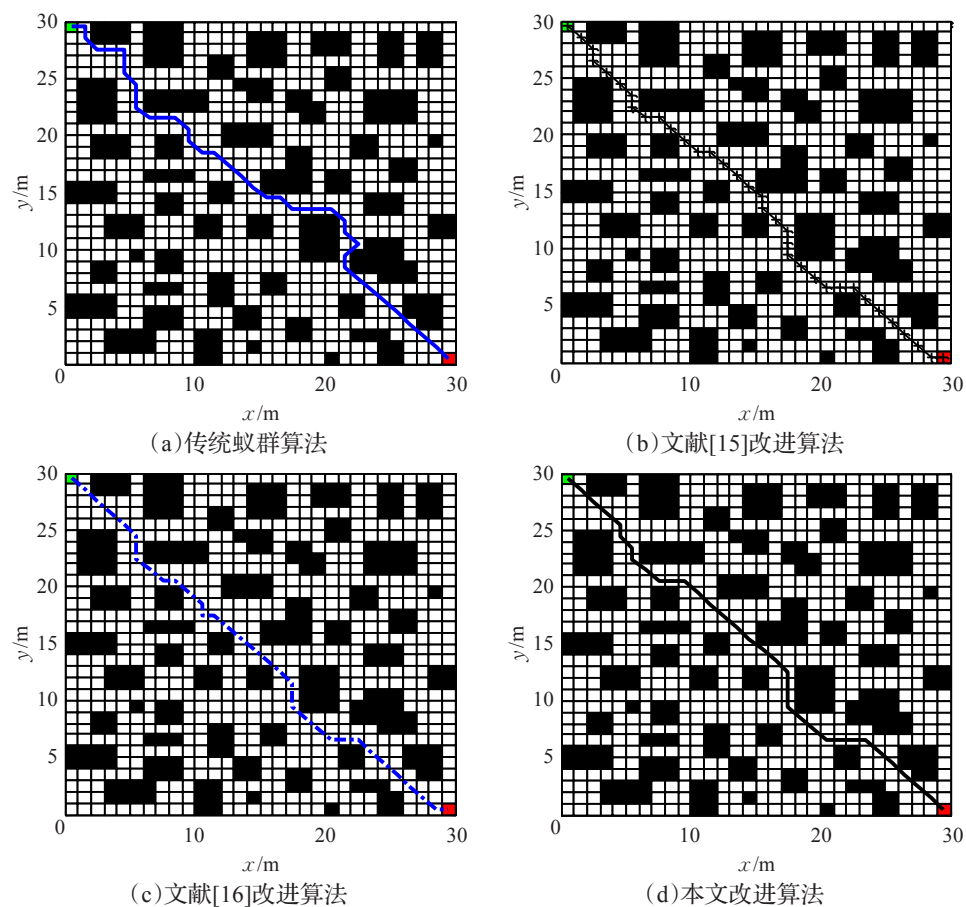


图7 30×30环境下四种算法的路径规划

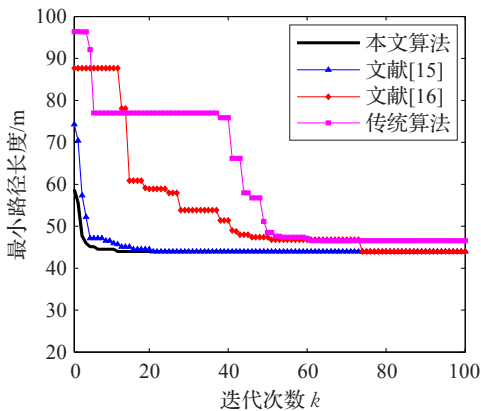


图8 30×30环境下四种算法的收敛曲线

搜索时具有良好的方向性,算法快速收敛,得到最优解的迭代次数为8。

由表3知,在本次实验中,本文改进算法所得路径长度平均值和迭代次数平均值均比其他三种算法小,说明本文改进算法的收敛速度比其他三种算法快,算法搜

索最优解的能力比其他三种算法强;由路径长度标准差和迭代次数标准差可得,本文改进算法的稳定性更好。由此可得,在30×30复杂环境中本文改进的算法依然表现出良好的鲁棒性和优越性。

综上所述,通过以上两种不同环境下对本文改进的蚁群算法、传统蚁群算法、文献[15]改进算法和文献[16]改进的算法进行仿真实验,从算法的收敛速度和搜索最优解的能力以及算法迭代初期所得结果大小和最优解的质量等方面对算法的性能进行分析。验证了本文改进算法在路径搜索时具有较强的方向性、稳定性,算法能快速找到最优解。当环境发生变化时本文改进蚁群算法具有较强的环境适应性,算法具有一般性。

5.3 解的多样性分析

为了进一步验证算法搜索最优解的能力,本文对算法解的多样性进行分析,解的多样性指在一次迭代中,所有蚂蚁搜索到不同解的个数,反映了算法搜索最优解

表3 30×30环境中四种算法仿真结果

地图环境	算法	路径长度				迭代次数			
		最优解	最大值	平均值	标准差	最小值	最大值	平均值	标准差
30×30复杂环境	传统ACO	47.355 3	66.828 4	53.924 0	5.327 5	16	48	31	11.189 0
	文献[15]	43.355 3	48.183 8	44.850 1	1.507 0	12	87	55	21.106 8
	文献[16]	44.769 6	51.355 3	47.257 4	1.716 5	36	96	66	17.703 0
	本文算法	43.355 3	46.769 6	44.436 1	0.971 0	4	15	9	2.849 0



的能力。解的多样性也反映了每次迭代每只蚂蚁的局部寻优能力,值越大表明蚂蚁之间寻找到不同解的概率越大,从而找到全局最优解的可能性越大。因而蚂蚁的局部寻优能力与算法的全局寻优能力成正相关。本文通过每代所有蚂蚁搜索到解的标准差  $D(k)$  作为解的多样性函数来分析算法寻优能力,  $D(k)$  计算如下式:

$$D(k) = \sqrt{\frac{\sum_{m=1}^M (L_m^k - \bar{L}_k)^2}{M-1}} \quad (18)$$

式中,  $L_m^k$  表示第  $k$  代第  $m$  只蚂蚁搜索到的路径长度,  $\bar{L}_k$  表示第  $k$  代所有蚂蚁搜索到路径的平均值。本次实验以图7所示环境为例,所得解的多样性曲线如图9所示。

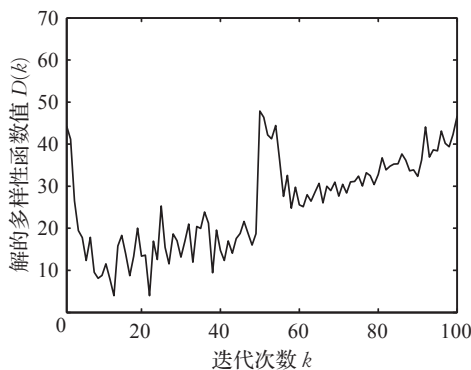


图9 30×30环境中解的多样性曲线

由图9知,随着算法迭代的进行,解的多样性逐渐降低,接着在一个区间来回波动,但  $D(k)$  值始终大于0,说明算法未出现停滞状态,每一代的解都具有较好的多样性,蚂蚁的局部寻优能力较强。当迭代次数达到所设定阈值50时,由于对信息素浓度和信息素挥发系数进行了调整,解的多样性增加,蚂蚁的局部寻优能力和算法全局搜索能力都增强。解的多样性整体呈现出上升趋势,说明本文改进的算法解具有较好的多样性,具有较强的全局搜索能力。

#### 5.4 最优路径平滑处理

在本实验环节中主要是对最优路径中的转折点进行平滑处理,实验首先在30×30复杂地图中应用改进的算法得到最优路径,然后应用分段三阶贝塞尔曲线进行路径平滑处理。蚁群算法规划的路径和用分段三阶贝塞尔曲线平滑后的路径对比如图10所示。

从图10可以看出,最优路径采用分段三阶Bezier曲线进行平滑处理后所得到的路径在转折点处更加平滑,更加接近机器人实际运动轨迹,平滑的路径提高了机器人运动的平稳性。

#### 5.5 算法实用性分析

本文所提出的自适应蚁群算法主要根据蚂蚁在实际寻路特点对搜索机制和信息素更新方式进行更加合理的优化,根据文献[14]对算法复杂度的定义,算法复杂度包括时间复杂度和空间复杂度,在本文算法中,设

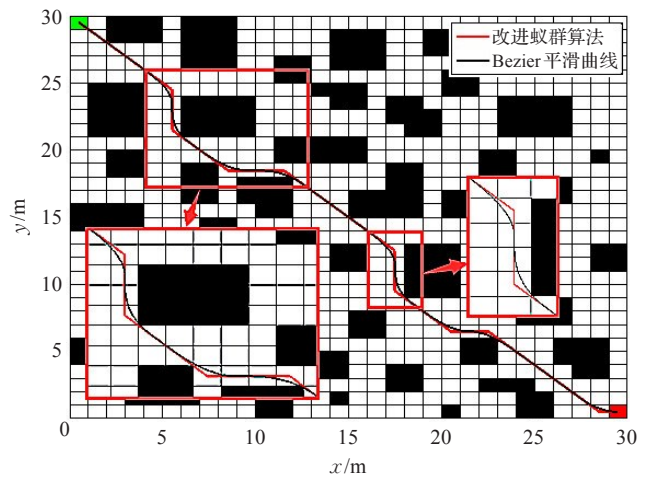


图10 最优路径Bezier曲线平滑处理

$n$  为求解问题的规模,  $m$  表示蚂蚁数量,  $K_{\max}$  表示最大迭代次数,则本文算法的时间复杂度表示为:

$$T(n) = O(n^2 \cdot m \cdot K_{\max}) \quad (19)$$

同理,根据文献[14]对算法空间复杂度定义为算法执行时间内所占用的存储单元,本文算法空间复杂度主要包两部分:一是对问题的描述,所需存储空间不超过数兆;二是实现算法所需要的数据,主要包括地图数据。空间复杂度计算公式为:

$$S(n) = O(n^2) + O(m \cdot n) \quad (20)$$

由式(19)和式(20)得本文算法的时间复杂度和空间复杂度与传统蚁群算法相同。但是本文算法更加具有优越性。在求解问题规模较大的场景以及后期的动态路径中,本文算法的适应性更强,算法普适性更好。

本文自适应蚁群算法主要应用在全局静态环境中,在动态实时路径规划中,当环境中有动态障碍物或者出现突发情况导致机器人不能按照预先规划的路径运动时,只需通过传感器得到运动障碍物在一定时间内的运动范围和突发情况所在地图中的位置,并将该位置的自由栅格标记为障碍物栅格,然后机器人以当前点为新起点,即可用本文自适应蚁群算法重新规划一条到目标点的避开动态障碍物的最优路径。在局部未知环境中,在搜索半径内的局部区域中找到局部最优目标点,再调用本文改进蚁群算法规划一条局部区域内的起点到局部目标点的局部最优路径,再以上一次的局部目标点作为本次寻找路径的起点,更新局部搜索区域,重复寻找局部目标点并调用本文算法进行路径规划直到找到全局目标点为止,所规划的所有局部最优路径连起来即为未知环境中的最优路径。

## 6 结论

在全局静态环境中,本文对蚁群传统蚁群算法应用于机器人路径规划中存在的不足,在栅格地图中对蚁群算法进行了自适应改进。

(1)根据最优路径的特点以及实际工作环境的基本



信息(地图尺寸、障碍物占比)对初始信息素进行差异化分配,使得算法在迭代初期信息素对路径选择具有弱启发性,保证信息素对路径选择具有启发性的同时避免算法陷入局部最优。

(2)在状态转移概率中引入角度启发信息,对减少路径上的转折次数提高平滑性作用显著;采用A\*算法中的估价代价函数作为路径启发信息,使得算法搜索路径方向性更强。

(3)改进信息素更新规则,信息素增量根据当前全局最优、当代最优、当代最差以及算法进化率来自适应确定,并只对一部分较优蚂蚁进行信息素更新。避免了最差蚂蚁对后代蚂蚁的影响,提高了算法搜索效率;当算法达到迭代阈值时,对路径上信息素浓度,信息素挥发系数进行调整,增强算法全局搜索能力。

(4)对最优路径采用分段三阶贝塞尔曲线平滑处理,使得路径轨迹平滑。

最后,通过实验和其他算法(传统蚁群算法、文献[15]和文献[16]改进算法)进行对比分析验证了本文改进算法在保证解的质量的同时收敛速度更快,搜索最优解的能力更强,能够保证快速收敛的同时依然具有较强的全局搜索能力,通过算法多样性分析以及采用贝塞尔曲线对最优路径的平滑处理后得到的曲线进行观察分析,充分说明了本文改进算法具有较好的适应性、鲁棒性和优越性。最后对算法的实用性进行了分析,阐述了本文算法应用在动态实时路径规划中的主要过程。下一步将对本文算法在动态路径规划中的应用展开深入研究。

## 参考文献:

- [1] 朱大奇,颜明重. 移动机器人路径规划技术综述[J]. 控制与决策, 2010, 25(7): 961-967.
- [2] Han J, Seo Y. Mobile robot path planning with surrounding point set and path improvement[J]. Applied Soft Computing, 2017, 57: 35-47.
- [3] 赵晓,王铮,黄程侃,等. 基于改进A\*算法的移动机器人路径规划[J]. 机器人, 2018, 40(6): 903-910.
- [4] 张希闻,肖本贤. 基于人工势场法的移动机器人动态路径规划[J]. 山东农业大学学报(自然科学版), 2018, 49(6): 937-940.
- [5] 王娟,吴宪祥,郭宝龙. 基于改进粒子群优化算法的移动机器人路径规划[J]. 计算机工程与应用, 2012, 48(15): 240-244.
- [6] 刘二辉,姚锡凡,蓝宏宇,等. 基于改进遗传算法的自动导引小车路径规划及其实现平台[J]. 计算机集成制造系统, 2017, 23(3): 465-472.
- [7] 李丽娜,郭永强,张晓东,等. 萤火虫算法结合人工势场法的机器人路径规划[J]. 计算机工程与应用, 2018, 54(20): 104-109.
- [8] 梁晓丹,蔺娜,陈瀚宁. 基于细菌觅食行为的移动机器人动态路径规划[J]. 仪器仪表学报, 2016, 37(6): 1316-1324.
- [9] Dorigo M, Maniezzo V, Colomi A. Ant system: optimization by a colony of cooperating agents[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 1996, 1: 29-41.
- [10] Dorigo M, Gambardella L M. Ant colonies for the travelling salesman problem[J]. Biosystems, 1997, 43(2): 73-81.
- [11] Blum C, Sampels M. An ant colony optimization algorithm for shop scheduling problems[J]. Journal of Mathematical Modelling and Algorithms, 2004, 3: 285-308.
- [12] Jiang Y, Xu Z, Xu X, et al. A schedule optimization model on multirunway based on ant colony algorithm[J]. Mathematical Problems in Engineering, 2014: 1-11.
- [13] Tan G Z, He H, Sloman A. Ant colony system algorithm for real-time globally optimal path planning of mobile robots[J]. Acta Automatica Sinica, 2007, 3: 279-285.
- [14] 王晓燕,杨乐,张宇,等. 基于改进势场蚁群算法的机器人路径规划[J]. 控制与决策, 2018, 33(10): 1775-1781.
- [15] Zhang Y, Wang F L, Fu F K, et al. Multi-AGV path planning for indoor factory by using prioritized planning and improved ant algorithm[J]. Journal of Engineering and Technological Sciences, 2018, 50: 534-547.
- [16] 黄辰,费继友,刘洋,等. 基于动态反馈A\*蚁群算法的平滑路径规划方法[J]. 农业机械学报, 2017, 48(4): 34-40.
- [17] 刘新宇,谭力铭,杨春曦,等. 未知环境下的蚁群-聚类自适应动态路径规划[J]. 计算机科学与探索, 2019, 13(5): 846-857.
- [18] Li P, Zhu H. Parameter selection for ant colony algorithm based on bacterial foraging algorithm[J]. Mathematical Problems in Engineering, 2016: 1-12.
- [19] 张强,陈兵奎,刘小雍,等. 基于改进势场蚁群算法的移动机器人最优路径规划[J]. 农业机械学报, 2019, 50(5): 23-42.
- [20] Akka K, Khaber F. Mobile robot path planning using an improved ant colony optimization[J]. International Journal of Advanced Robotic Systems, 2018, 15(3).
- [21] 朱颢东,孙振,吴迪,等. 基于改进蚁群算法的移动机器人路径规划[J]. 重庆邮电大学学报(自然科学版), 2016, 28(6): 849-855.
- [22] 徐梁. 轮式移动机器人路径规划研究[D]. 成都: 西南交通大学, 2018.
- [23] 邓高峰,张雪萍,刘彦萍. 一种障碍环境下机器人路径规划的蚁群粒子群算法[J]. 控制理论与应用, 2009, 26(8): 879-883.
- [24] 王芳,李昆鹏,袁明新. 一种人工势场导向的蚁群路径规划算法[J]. 计算机科学, 2014, 41(Z11): 47-50.
- [25] 王维,裴东,冯璋. 改进A\*算法的移动机器人最短路径规划[J]. 计算机应用, 2018, 38(5): 1523-1526.
- [26] 余伶俐,龙子威,周开军. 基于贝塞尔曲线的机器人非时间轨迹跟踪方法[J]. 仪器仪表学报, 2016, 37(7): 1564-1572.
- [27] 刘学问,陶钧,徐海巍. 基于三阶贝塞尔曲线的AGV轨迹规划研究[J]. 工业控制计算机, 2018, 31(1): 113-114.