

基于改进自适应蚁群算法的移动机器人路径规划

魏书鑫, 王群京, 李国丽, 许家紫, 文彦

引用本文

魏书鑫, 王群京, 李国丽, 许家紫, 文彦. [基于改进自适应蚁群算法的移动机器人路径规划](#)[J]. 计算机科学, 2024, 51(6A): 230500145-9.

WEI Shuxin, WANG Qunjing, LI Guoli, XU Jiazi, WEN Yan. [Path Planning for Mobile Robots Based on Modified Adaptive Ant Colony Optimization Algorithm](#) [J]. Computer Science, 2024, 51(6A): 230500145-9.

相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

[路径规划问题的多策略改进樽海鞘群算法研究](#)

Study on Multi-strategy Improved Salp Swarm Algorithm for Path Planning Problem
计算机科学, 2024, 51(6A): 230600083-9. <https://doi.org/10.11896/jsjcx.230600083>

[航母航空保障作业中异质群体的动态路径规划算法](#)

Dynamic Path Planning Algorithm for Heterogeneous Groups in Aircraft Carrier Aviation Support Operations
计算机科学, 2024, 51(3): 226-234. <https://doi.org/10.11896/jsjcx.221200119>

[面向绿色节能的智能网联电动车调度方法](#)

Intelligent Networked Electric Vehicles Scheduling Method for Green Energy Saving
计算机科学, 2023, 50(12): 285-293. <https://doi.org/10.11896/jsjcx.230100099>

[面向能耗优化和负载均衡的边缘服务器放置研究](#)

Edge Server Placement for Energy Consumption and Load Balancing
计算机科学, 2023, 50(6A): 220300088-5. <https://doi.org/10.11896/jsjcx.220300088>

[基于多元约束Petri网的水利测绘无人机路径规划](#)

Path Planning of Hydrographic Mapping UAV Based on Multi-constraint Petri Net
计算机科学, 2023, 50(6A): 220700079-7. <https://doi.org/10.11896/jsjcx.220700079>

基于改进自适应蚁群算法的移动机器人路径规划

魏书鑫^{1,2} 王群京^{1,2} 李国丽^{1,2} 许家紫^{1,2} 文彦^{1,3}

1. 安徽大学电气工程与自动化学院 合肥 230601

2. 安徽大学高节能电机及控制技术国家地方联合实验室 合肥 230601

3. 安徽大学互联网学院 合肥 230601

(z21301088@stu.ahu.edu.cn)

摘要 针对传统的蚁群算法(Ant Colony Optimization, ACO)存在收敛速度慢、效率低、容易陷入局部最优值等缺点,提出了一种新的ACO变体。首先引入了一种新的具有方向信息的启发式机制,在迭代过程中添加方向指导,进一步提高了算法的收敛速度。其次,提出了一种改进的启发式函数,以增强目标的目的性并减少路径的转弯次数。然后,引入了一种改进的状态转移概率规则,提高了搜索效率并增加了种群多样性。此外,提出了一种不均匀分布初始信息素浓度的新方法,以避免盲目搜索。形成的新的ACO变体称为改进的自适应蚁群优化算法(Modified Adaptive Ant Colony Optimization, MAACO)。为了验证所提出的MAACO的有效性,基于3种不同的空间环境模式,与现有其他7种算法进行了一系列实验。在所有的仿真实验中,所提出的MAACO生成了标准偏差为零的最短路径,并且在最小收敛生成内实现了最少的转弯次数;就3个实验而言,其与最佳现有结果相比,转弯次数平均减少了两次,平均减少比例为22.2%。实验结果证明了MAACO在减少路径长度、减少转弯次数和提高收敛速度方面的优点和其在路径规划中的实用性和高效性。

关键词: 蚁群算法;启发函数;转移概率;移动机器人;路径规划

中图分类号 TP242

Path Planning for Mobile Robots Based on Modified Adaptive Ant Colony Optimization Algorithm

WEI Shuxin^{1,2}, WANG Qunjing^{1,2}, LI Guoli^{1,2}, XU Jiazi^{1,2} and WEN Yan^{1,3}

1 School of Electrical Engineering and Automation, Anhui University, Hefei 230601, China

2 National Engineering Laboratory of Energy-Saving Motor & Control Technology, Anhui University, Hefei 230601, China

3 School of Internet Academy, Anhui University, Hefei 230601, China

Abstract For the traditional ACO has the disadvantages of slow convergence, low efficiency and easy to fall into local optimum, a new variant of ACO is proposed. Firstly, a new heuristic mechanism with directional information is introduced to add directional guidance in the iterative process, which further improves the convergence speed of the algorithm. Second, an improved heuristic function is proposed to enhance the purpose of the objective and reduce the number of turns in the path. Then, an improved state transfer probability rule is introduced to improve the search efficiency and increase the population diversity. In addition, a new method of unevenly distributing the initial pheromone concentration is proposed to avoid blind search. The new ACO variant is called the modified adaptive ant colony optimization algorithm(MAACO). To verify the effectiveness of the proposed MAACO, a series of experiments are conducted with seven other existing algorithms based on three different obstacle distribution environment patterns. In all simulation experiments, the proposed MAACO generates the shortest path with zero standard deviation and achieves the minimum number of turns within the minimum convergence generation. For the three experiments, the average reduction in the number of turns compared to the best available results is two, with a typical reduction of 22.2%. Experimental results demonstrate the advantages of MAACO in reducing path length, reducing the number of turns and increasing the convergence speed and its usefulness and efficiency in path planning.

Keywords Ant colony algorithm, Heuristic function, Transfer probability, Mobile robot, Path planning

1 引言

移动机器人路径规划的目的是在满足工程要求的空间环境中搜索从起点到目标点的最优无碰撞路径^[1]。通常,路径规划的步骤主要包括环境建模和路径搜索。详细地说,环境

建模用于将实际的物理空间转换为抽象的空间环境。应用路径搜索,在环境模型的基础上找到可行路径,进一步得到基于预定执行函数的最优解。路径规划是一个研究热点,在许多行业都受到了广泛的关注。有学者已经研究出了用于路径规划的方法和算法,例如神经网络算法^[2]、模拟退火算法^[3]、

基金项目:国家自然科学基金重点项目(51637001)

This work was supported by the Key Program of the National Natural Science Foundation of China(51637001).

通信作者:王群京(wangqunjingii@ahu.edu.cn)

Dijkstra 算法^[4]、A* 算法^[5]、人工势场法^[6]、快速探索随机树法^[7]、蚁群算法^[8]、Floyd 算法^[9]等。

随着仿生算法和人工智能的研究和发展,智能优化算法得到了迅速发展,如遗传算法(Genetic Algorithm, GA)、粒子群优化算法(Particle Swarm Optimization, PSO)和蚁群算法。遗传算法是一种通过模拟自然进化过程来寻找最优解的方法^[10]。为了解决自主水下航行器的路径规划问题,Chong 等提出了一种改进的自适应遗传算法(Improved Adaptive Genetic Algorithm, IAGA)^[11]。实验结果表明, IAGA 为高精度、长航程自主导航提供了有效的保证。此外,粒子群算法收敛速度快,设置参数少,在路径规划领域得到了广泛的应用。为了获得较高的最优飞行轨迹精度, Kathen 等提出了一种改进的 PSO 来求解高级安全飞行器的路径规划^[12]。ACO 算法是另一种广泛用于路径规划的进化算法,它是一种基于群体觅食行为的随机启发式搜索算法^[13]。通常,ACO 采用分布式计算、正反馈并行、启发式搜索系统,并具有正反馈、高鲁棒性和并行性的特点。ACO 首先用于解决旅行商问题(Traveling Salesman Problem, TSP)。随后,它被用于解决其他组合优化问题,例如路径规划、车辆路由问题、作业车间调度问题。然而,传统的 ACO 存在搜索效率低、收敛速度慢、容易陷入局部最优解等缺陷。因此,许多研究人员在搜索策略、启发式信息、信息素更新和路径选择规则方面对传统的 ACO 进行了改进。通过将精英策略与传统 ACO 相结合,提出了蚂蚁系统(Ant System, AS)以保留每一代中最好的个体^[14]。然而,太多的精英蚂蚁会导致搜索过早收敛,进一步导致早期停滞。随后, Fatemidokht 等提出了最大-最小蚂蚁系统(Max-Min Ant System, MMAS)来应对过早停滞现象^[15]。详细地说,MMAS 将信息素浓度保持在一个区间内。最小信息素浓度增加了探索更好解决方案的可能性,而最大信息素浓度确保了蚁群的启发式体验是基于最优解决方案的。

此外,通过调整信息素的初始值并使用人工势场法设置迭代阈值, Wang 等提高了 ACO 避免陷入局部最优的能力^[16]。为了解决机器人路径规划问题, Zhu 等结合 A* 算法引入了一种改进的蚁群算法^[17]。HUI 等将狼群分配引入蚁群算法的启发式函数中,解决了移动机器人的路径规划问题^[18]。Miao 等提出了一种改进的 ACO,该方法改进状态转移概率、信息素更新规则和启发式信息^[19],实验结果表明,其能够平衡全局搜索能力和收敛速度。通过结合移动服务机器人全局路径规划的模糊控制, Tao 等提出了一种可以提高收敛速度的 ACO 变体^[20]。为了弥补 ACO 求解速度慢的缺点, HSU 等提出了一种改进的精英蚁群算法来求解最优路径规划^[21]。Li 等对 ACO 的精英势场进行了改进,提出了一种三维路径规划算法^[22],改进后的 ACO 在求解三维机器人路径规划时具有收敛速度快、搜索精度高、效率高等优点。尽管上述研究已经取得了很高的算法性能,并为实际情况提供了高质量的解决方案,但在路径规划问题的有效性和效率方面,仍存在改进空间。在这项研究中,提出了 4 种新颖的机制,旨在弥补传统 ACO 的缺陷。主要贡献概括如下:

(1)提出了一种带方向信息的启发式机制,使算法具有方向智能,能够降低计算复杂度和计算时间。

(2)将规划路径的起点、目标点、当前点和转弯时间信息嵌入到改进的启发式函数中。启发式函数的改进便于调整从

当前点到目标点的搜索速度,并平衡启发式信息的强度。

(3)引入了一种自适应伪随机传递规则,以提高搜索效率和群体多样性。此外,提出了一种自适应调整机制来改进状态转移概率规则。

(4)提出了一种新的信息素分布方法,通过预先选择可选区域来提高路径规划的引导能力。不均匀分布包括初始信息素的不均匀分布和信息素浓度更新后的限制。

2 ACO 的数学模型

传统的 ACO 是一种仿生算法,其灵感来自自然界中蚂蚁的觅食行为。在觅食过程中,蚂蚁在寻找食物的同时会在其路径上释放信息素,其他蚂蚁可以探测到同一物种发出的信息素。通常,信息素会随时间蒸发,在较短的路径上剩余的信息素浓度更高。因此,蚁群可以选择一条最佳路径,并倾向于向有高信息素的方向移动。传统 ACO 的基本流程如下文所示。

2.1 初始化参数设置

首先,需要初始化 ACO 的参数。这些参数包括最大迭代次数 K 、蚂蚁数量 M 、启发式因子 α 、预期启发式因子 β 、信息素蒸发因子 ρ 和信息素强度值 Q 。此外,启发式信息矩阵中,信息素的值也应同样进行初始化。

2.2 迭代路径搜索

如何选择最优路径是 ACO 的重要问题之一。在传统的 ACO 中,采用轮盘选择方法。这种思想是,每个个体被选择的概率与其适应度成正比。在传统的 ACO 中,蚂蚁从第 i 点到第 j 点的第 k 个蚂蚁的概率选择规则定义为:

$$p_{ij}^m = \begin{cases} \frac{[\tau(i,j)]^\alpha * [\eta(i,j)]^\beta}{\sum_{s \in J_m(i)} [\tau(i,s)]^\alpha * [\eta(i,s)]^\beta}, & (i,j) \in J_m \\ 0, & \text{else} \end{cases} \quad (1)$$

其中 J_m 表示下一次迭代中蚂蚁 m 的可选网格集合, p_{ij}^m 表示从第 i 点到第 j 点的每个可选路径的转移概率, α 是信息素重要性因子, β 是启发式函数的重要性因子, $\tau(i,j)$ 表示信息素浓度在 (i,j) 点上的值, $\eta(i,j)$ 表示启发式信息的可见性,为启发式值,其计算公式由式(2)给出。

$$\eta(i,j) = \frac{1}{d_{ij}} \quad (2)$$

其中, d_{ij} 是从第 i 点到第 j 点的欧几里得距离。

2.3 信息素更新

在每个蚂蚁构建从起点到目标点的路径之后,路径中包含的每个边缘上的信息素浓度将根据路径的总长度进行更新。然后,在所有蚂蚁进行迭代搜索之后,信息素浓度将在全部范围内更新。信息素浓度的更新规则见式(3)和式(4):

$$\tau_{t+1}^m(i,j) = (1-\rho) * \tau_t^m(i,j) + \sum_{m=1}^M \Delta\tau_t^m(i,j) \quad (3)$$

$$\Delta\tau_t^m(i,j) = \begin{cases} Q/L_m, & \text{如果蚂蚁 } m \text{ 从节点 } i \text{ 上行进到节点 } j \\ 0, & \text{其他} \end{cases} \quad (4)$$

其中, ρ 表示数值介于 0 和 1 之间的全局信息素蒸发因子; $1-\rho$ 为信息素残差系数; $\tau_{t+1}^m(i,j)$ 为当前迭代中从第 i 点到第 j 点的信息素浓度; $\Delta\tau_t^m(i,j)$ 表示第 m 个蚂蚁在当前迭代中从第 i 点到第 j 点的路径上的信息素浓度变化; Q 是一个常数,表示信息素强度; L_m 表示当前迭代中第 m 个蚂蚁的总长度。

3 改进的自适应 ACO(MAACO)

本节提出了 4 种机制来克服传统 ACO 的缺点。首先,提出了一种具有方向信息的启发式机制,以通过形成方向智能来降低计算复杂度。然后,提出了一种改进的启发式函数,以提高算法的目的性,有效避免搜索的盲目性。随后,引入了状态转移概率规则,以提高搜索效率并增加群多样性。最后,提出了一种分布不均匀的初始信息素,以增强搜索过程中的指导能力。这 4 种机制,形成了一种新的 ACO 变体,称为改进的自适应 ACO(MAACO)。MAACO 流程图如图 1 所示。

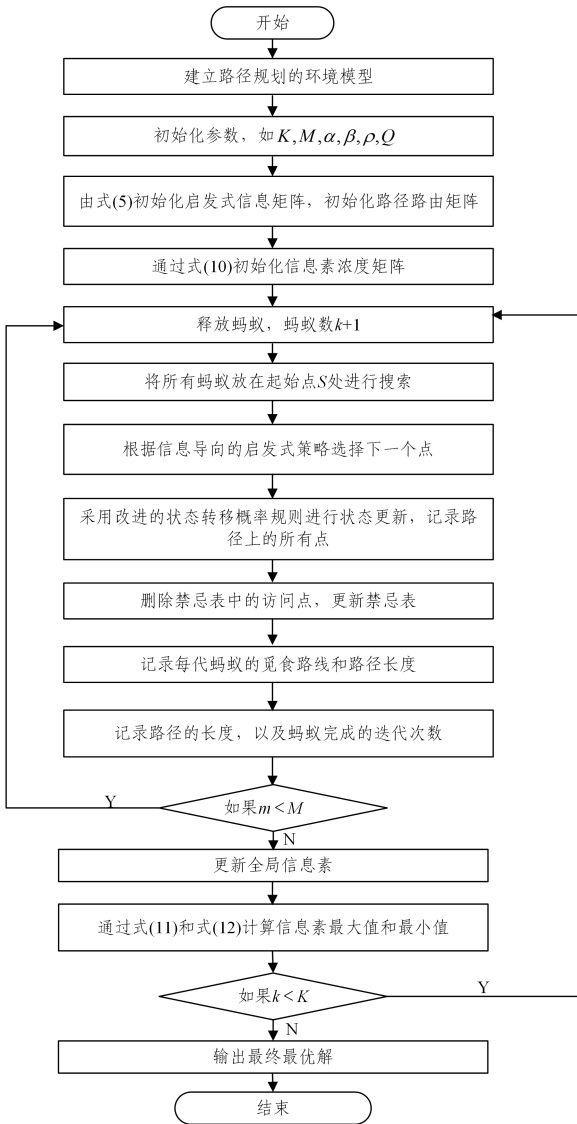


图 1 MAACO 的流程图

Fig. 1 Flow chart of MAACO

3.1 具有定向信息的启发式机制

具有当前点的方向信息的启发式机制中,如果在传统 ACO 中当前点周围没有障碍物,则可以选择 8 个可选点作为下一个点。如图 2 所示,当前点由 P 表示,其 8 个可选点由 $P1, P2, P3, P4, P5, P6, P7$ 和 $P8$ 表示。根据传统 ACO 的规则,需要这 8 个可选点来计算相应的转移概率。这意味着进行了一些不必要的计算,计算时间将变得 longer,同时还会导致路径冗余,并增加搜索路径时的转弯次数。

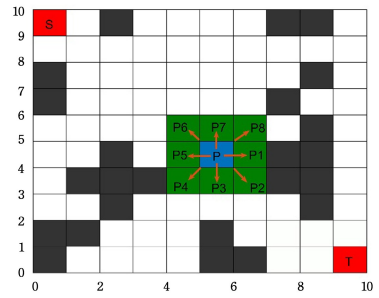


图 2 传统 ACO 中当前点的 8 个可选点

Fig. 2 Eight selectable points of the current point in traditional ACO

因此,需要从起点到目标点给出方向指导,并在选择下一个点时删除一些不必要的操作。基于此,提出了一种新的带有定向信息的启发式机制,以克服传统 ACO 的缺点。根据起始点和目标点的位置,形成方向信息,并且将当前点的 8 个可选点简化为 3 个可选点。起始点 S 的坐标为 (x_s, y_s) ,目标点 T 的坐标为 (x_T, y_T) ,如图 3 所示,起始点 S 与目标点 T 的方位信息由 S 至 T 的箭头表示。显然,如果将起点视为坐标系的原点,则从 S 到 T 的箭头方向指向第四象限。因此,为了保持有效的搜索过程,希望当前点的下一个选定也与箭头方向一起。

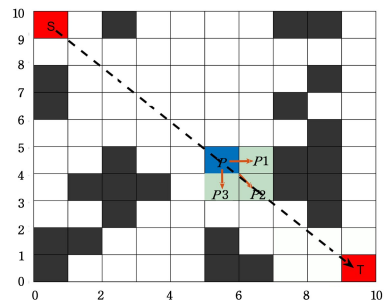


图 3 具有方向信息的新型启发式机制的示意图

Fig. 3 Schematic diagram of a novel heuristic mechanism with directional information

假设当前点 p 的坐标为 (x_c, y_c) ,下一个点 p_n 的坐标为 (x_d, y_d) 。根据具有方向信息的新型启发式机制,从 p 到 p_n 的向量应指向第四象限。这意味着 $x_{d-c} = x_d - x_c$ 的值不应小于 0,并且 $y_{d-c} = y_d - y_c$ 的值不应大于 0。基于此规则,上述 8 个点中的 5 个可选点 ($P4, P5, P6, P7, P8$) 不能满足要求,仅保留 3 个可选点,如图 3 所示。如果从 S 到 T 的箭头方向指向第一象限,则 x_{d-c} 和 y_{d-c} 的值均应大于等于 0。还有另外两种情况:如果箭头指向第二象限,则 x_{d-c} 的值不应大于 0,并且 y_{d-c} 的值不应小于 0;如果箭头指向第三象限,则 x_{d-c} 和 y_{d-c} 的值应不大于 0。

可选点数量的减少使得算法具有方向智能,有利于降低计算复杂度,减少计算时间。另一方面,所提出的具有方向信息的启发式机制具有避免盲目搜索和路径冗余的能力。

3.2 改进的启发式函数

在传统 ACO 中,启发式函数 $\eta(i, j) = 1/d_{ij}$ 的值仅与当前点和下一个可选点之间的距离有关。但是,环境模型中任意两个相邻点之间的距离是一个已知值,该值为 1 或 $\sqrt{2}$ 。这意味着启发式功能缺乏指导性和启发性,启发式信息无法生效,蚂蚁会随机选择搜索路径,进一步使算法在早期迭代中表现不佳。因此应考虑起点和目标点的位置信息,并将其集成

到启发式函数中。这样路径的搜索过程将变得更有目的性,进一步有效地避免了搜索的盲目性。

此外,传统的 ACO 在路径规划中只考虑距离因素的影响,过多的转弯次数会导致机器人的能量损失。因此,还应考虑最佳路径的性能,以减少转弯次数。本节提出了一种改进的启发式函数,其中,启发式函数是根据转弯次数、下一个点、起点和目标点连接之间的相对距离信息来计算的。所提出的启发式函数的计算公式如式(5)所示:

$$\eta(i, j)' = \frac{1}{g * d_{sj} + h * d_{jT} + a * c(i)} \quad (5)$$

其中, $\eta(i, j)'$ 表示改进的启发式函数; d_{sj} 是起始点到下一点的距离, d_{jT} 为下一个点到目标点的距离; g 为 d_{sj} 的权值, h 为 d_{jT} 的权值, g 和 h 满足公式 $g + h = 1$; $c(i)$ 表示上一个点 $i-1$ 与下一个点 j 之间的回转次数; a 为转弯系数。

改进的启发式函数综合考虑了点距离信息和转弯次数。距离越短,转弯次数越少,启发式函数就越好。因此,基于改进的启发式函数,下一个选定的点可以有更好的位置。这就增加了算法的搜索目的性,改进的启发式函数可以有效地提高算法的搜索目标和规划路径的平稳性。

此外,为了促进收敛速度,在以后的迭代中应减弱启发式信息对路径的影响。因此,引入了自适应的正相关权重调整函数,如式(6)所示:

$$h = w_{h \max} - (w_{h \max} - w_{h \min}) * e^{\wedge}(-k * d_{jT}) / d_{ST} \quad (6)$$

其中, h (d_{jT} 的权重系数) 的变化与 d_{jT} 的值呈正相关。由于 g 和 h 的总和为 1, 因此在固定其中一个之后, 也确定了另一个。 $w_{h \max}$ 和 $w_{h \min}$ 为赋值初始值, k 为通过测试以选择最佳值的系数, d_{ST} 表示从起点到目标点的距离, 距离计算采用欧氏距离计算式, 如式(7)所示:

$$d_{ST} = \sqrt{(x_T - x_s)^2 + (y_T - y_s)^2} \quad (7)$$

如果下一个点 j 离目标点较远, d_{jT} 的值较大, 则需要减少搜索空间以加快搜索过程。因此, h 的值应根据式(6)适当增加。当 h 和 d_{jT} 的值较大时, $\eta(i, j)'$ 将减小, 这可以减弱启发式函数的指导作用, 加快搜索速度。相反, 如果下一个点靠近目标点, 则 d_{jT} 的值较小, 搜索范围扩大, 搜索能力需要提高。因此, h 的值应根据式(6)适当降低。当 h 和 d_{jT} 的值较小时, $\eta(i, j)'$ 会增加, 可以增强启发式函数的指导作用, 提高搜索精度。启发式函数的改进可以调整从当前点到目标点的收敛速度, 平衡启发式信息的强度。改进后的机制增强了算法的目的性, 减少了计划路径的转弯次数。

3.3 改进的状态转移概率规则

在规划移动机器人的路径时, 需要计算每个可选点的移动概率。为了增加群体多样性, 提高搜索效率, 本文提出了一种自适应伪随机转移规则。假设蚂蚁 m 在 t 时刻位于点 i 的位置, 则蚂蚁 m 在下一个时刻 $t+1$ 的位置由 j 表示, j 可以通过如式(8)所示的改进的伪随机转移规则来表示:

$$j = \arg \max \{ [\tau_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta \}, q \leq q_0 \quad (8)$$

其中, $\arg \max(f(j))$ 是找到 j 的值的函数, 可以产生 $f(j)$ 的最大值, q 是自适应转移概率, 它是分布在 $[0, 1]$ 范围内的随机变量, q_0 表示 q 的阈值。基于式(8), 蚂蚁 m 在 $t+1$ 时的位置将定位在可以产生信息素浓度和启发式信息可见性的最大乘积的位置。显然, 该方法属于确定性选择模式, 有利于提高收敛速度。另一方面, 如果 $q > q_0$, 则蚂蚁 m 的概率选择规则

将根据传统 ACO 操作, 采用轮盘选择方法, 如式(1)所示, 轮盘选择方法是一种随机选择模式, 可以保持良好的全局搜索能力。

实际上, q_0 的值决定了将选择哪种模式(确定性选择模式或随机选择模式)的概率。因此, q_0 对收敛速度和全局搜索能力有显著影响。如果 q_0 的值很大, 则下一个点的选择更有可能是确定性模式。在这种情况下, 它可以加快收敛速度。然而, 全局搜索能力将会降低。如果 q_0 的值较小, 则路径的偏移倾向于随机模式, 这将增加路径选择的随机性, 并进一步增加全局搜索的能力。因此, 有必要提出一种用于设置 q_0 值的规则, 以平衡确定性和随机模式。因此, 引入了 q_0 的自适应调整机制来改进状态转换概率规则。提出的 q_0 的计算公式如式(9)所示:

$$q_0 = \begin{cases} \frac{K-k}{K} q_{0 \text{ initial}}, & k < k_0 \\ \frac{k-k_0}{K} q_{0 \text{ initial}} + \frac{q_{0 \text{ initial}}}{2}, & k \geq k_0 \end{cases} \quad (9)$$

$$k_0 = 0.7K$$

其中, k 是当前迭代次数; K 是最大迭代次数; $q_{0 \text{ initial}}$ 是 q_0 的原始值。根据式(9), 在迭代的早期阶段生成较大的 q_0 值。因此, 该算法可以在快速的收敛速度下有效地搜索最优路径。随着迭代的增加, q_0 的值将变得很小, 群体多样性将得到增强, 算法的全局搜索能力将得到提高, 以防止算法陷入局部最优。

3.4 不均匀分布初始信息素

在传统的 ACO 中, 信息素的初始值是均匀分布的, 这将导致迭代初期的盲搜索和收敛速度低。为了克服传统 ACO 的缺点, 提出了一种不均匀分布初始信息素的新方法, 以增强在路径规划中预先选择可选区域的引导能力。所提出的不均匀分布的初始信息素如式(10)所示:

$$\begin{cases} \tau_{ij}(\text{initial}) = \frac{d_{ST}}{d_{si} + d_{iT}} C(0), & \text{非障碍物网格} \\ \tau_{ij}(\text{initial}) = 0, & \text{障碍物网格} \end{cases} \quad (10)$$

其中, d_{si} 是从起点 S 到当前点 i 的欧氏距离, d_{iT} 是从当前点 i 到目标点 T 的欧氏距离。障碍物网格的信息素浓度为 0。 $C(0)$ 是传统 ACO 中信息素浓度的初始值。

对于图 4 所示的特殊情况, 引入欧几里得距离 d_{ST} , d_{si} 和 d_{iT} 。显然, d_{si} 和 d_{iT} 的最小和就是 d_{ST} 的值。这意味着从起点到目标点的路径上的信息素浓度具有 $\tau_{ij}(\text{initial}) = C(0)$ 的最大值。根据等式(10), 当前点 i 与连接起点和目标点的直线距离越近, 信息素浓度就越高。由于所提出的初始信息素分布不均匀, 因此算法的搜索过程更有目的性和效率。

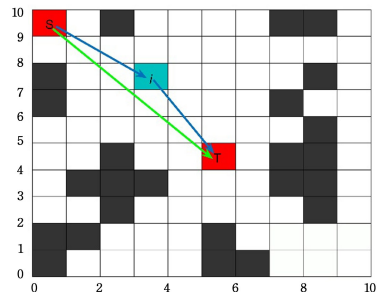


图 4 初始信息素分布不均匀的示意图

Fig. 4 Schematic diagram of uneven distribution of initial pheromones

为了防止陷入局部最优,我们利用 MMAS 来限制信息素浓度,并在更新信息素浓度后将信息素浓度 $\tau_{ij}(t)$ 保持在区间 $[\tau_{\min}, \tau_{\max}]$ 内。基于 MMAS,具体实现规则定义如式(11)所示:

$$\tau_{ij}(t) = \begin{cases} \tau_{\max}, & \tau_{ij}(t) \geq \tau_{\max} \\ \tau_{ij}(t), & \tau_{\min} < \tau_{ij}(t) < \tau_{\max} \\ \tau_{\min}, & \tau_{ij}(t) \leq \tau_{\min} \end{cases} \quad (11)$$

其中, $\tau_{ij}(t)$ 是时间 t 从点 i 到点 j 的信息素浓度, τ_{\max} 和 τ_{\min} 是 τ_{ij} 的最大值和最小值。为了吸收以前迭代的经验,提出了一种新的 τ_{\max} 和 τ_{\min} 的计算方法,如式(12)所示:

$$\begin{cases} \tau_{\max} = \left[\frac{1}{2 * (1-\rho)} * \frac{1}{\min kl} + \frac{1}{\min kl} \right] \\ \tau_{\min} = \frac{\tau_{\max}}{500} \end{cases} \quad (12)$$

其中, $\min kl$ 表示在完成第 k 次迭代之后当前蚂蚁 m 的最短路径长度。事实上, ρ 是信息素蒸发因子, $(1-\rho)$ 是信息素残留因子。根据方程(12),最短路径长度和信息素残差因子都涉及 $\tau_{ij}(t)$ 的有限区间,这有利于在接下来的迭代中找到最优解。显然,该方法可以提高算法的效率,加快收敛速度。此外,我们限制了信息素的浓度,这在避免陷入局部最优上是有用的。

4 仿真实验

为了测试和验证 MAACO 在解决移动机器人路径规划问题方面的性能,本节进行了 3 组实验。采用当前文献中几种可用的 ACO 变体和两种广泛使用的智能算法(A* 算法和 Dijkstra 算法)进行比较,以证实所提出的 MAACO 的有效性和可行性。

4.1 模拟实验 1

第一部分使用了 5 种智能算法,包括传统蚁群算法、蚁群算法的变体(称为多启发式改进蚁群算法(IACO))、文献[19]

提出的改进自适应蚁群算法(IAACO)、A* 算法和 Dijkstra 算法,与提出的 MAACO 进行比较,以验证其优越性。测试所采用的空间环境模式与文献[19]一致。如图 5 所示,在尺寸为 20×20 的环境模式下,红点表示起点 S,黄点表示目标点 T,黑色区域表示障碍物。起点 S 的索引位置为 1,位于环境模型的左上角。目标点的索引位置为 400,位于右下角。

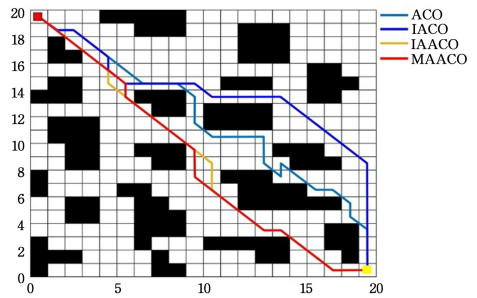


图 5 ACO,IACO,IAACO 和 MAACO 生成的最佳路径

Fig. 5 Optimal paths generated by ACO,IACO,IAACO and MAACO

为了测试所提出的 MAACO 机制的功能,在本模拟实验中,将 ACO 的 4 种变体(ACO, IACO, IAACO, MAACO)的公共参数设置为相同。具体而言,ACO, IACO 和 IAACO 的参数设置为 $M=50, K=100, \lambda=7, K=0.9, \alpha=1, \beta=7, \sigma_1=0.1, \sigma_2=0.9, R_s=0.5, \delta_0=0.15, Q=2.5, k_L=0.7, k_S=0.1, k_E=0.2$ 。对于 IACO 和 IAACO,其具体参数如 λ (角度引导因子的权重系数)、 k (调整系数)、 σ_1 和 σ_2 (距离权重系数和 $\sigma_1 + \sigma_2 = 1$)以及相应的描述可以在文献[19]中找到。所提出的 MAACO 的参数设置为 $M=50, K=100, \alpha=1, \beta=7, \rho=0.2, q_{0_initial}=0.5, Q=2.5, \omega_{h_{\max}}=0.9, \omega_{h_{\min}}=0.2$ 。模拟实验中使用的详细参数如表 1 所列。每个算法独立运行 20 次。

表 1 模拟实验 1 中使用的参数

Table 1 Parameters used in simulation experiment 1

算法	M	K	λ	k	α	β	σ_1	σ_2	R_s	δ_0	Q	k_L	k_S	k_E	ρ	$q_{0_initial}$	a	$\omega_{h_{\max}}$	$\omega_{h_{\min}}$
ACO	50	100	7	0.9	1	7	0.1	0.9	0.5	0.15	2.5	0.7	0.1	0.2	0.2				
IACO	50	100	7	0.9	1	7	0.1	0.9	0.5	0.15	2.5	0.7	0.1	0.2	—				
IAACO	50	100	7	0.9	1	7	0.1	0.9	0.5	0.15	2.5	0.7	0.1	0.2	—				
MAACO	50	100	—	—	1	7	—	—	—	—	2.5	0.7	—	—	0.2	0.5	1	0.9	0.2

此外,本文方法与 A* 算法和 Dijkstra 算法对比。对于 A* 算法,在搜索过程中建立了启发式搜索规则,用于测量搜索位置与目标位置之间的距离。此外,搜索方向优先朝向目标点,最终达到提高搜索效率的效果。Dijkstra 算法的主要

特点是,在每次迭代中,下一个选择节点是当前节点最近的子节点,并且每次迭代的遍历距离最短。6 种算法的实验结果如表 2 所列。获得的最佳路径如图 5 和图 6 所示,4 种变体 ACO 的收敛迭代曲线如图 7 所示。

表 2 6 种算法实验对比结果

Table 2 Experimental result comparison of six algorithms

算法	路径长度			收敛迭代次数			转弯次数
	最佳长度	平均值	标准差	最佳值	平均值	标准差	
ACO	31.80	33.764	2.9262	17	27.4	5.5353	16
IACO	32.00	33.260	0.8294	6	9.3	3.6069	9
IAACO	28.63	29.505	0.7497	5	5.3	0.2202	7
A*	28.63	28.630	0	—	—	—	9
Dijkstra	29.80	29.800	0	—	—	—	9
MAACO	28.63	28.630	0	3	4.0	0.8944	7

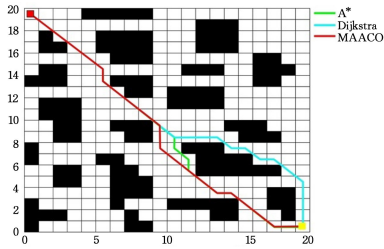


图 6 A*, Dijkstra 和 MAACO 生成的最佳路径

Fig. 6 Optimal paths generated by A*, Dijkstra and MAACO

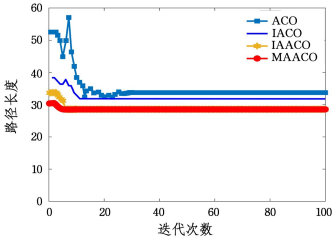


图 7 ACO, IACO, IAACO, MAACO 的收敛迭代曲线

Fig. 7 Convergence iteration curves of ACO, IACO, IAACO and MAACO

如表 2 所列, 尽管所提出的 MAACO 与 A* 算法和 IACO 具有相同的最优路径长度 (28.63), 小于 Dijkstra 算法、ACO 和 IACO 的最优路径长 ($28.63 < 29.80 < 31.80 < 32.00$), 但 MAACO 在均值和标准差方面性能最佳。具体而言, MAACO 获得的路径长度的均值仅为 28.630, 小于 IAACO, IACO 和 ACO 均值 ($28.630 < 29.505 < 33.260 < 33.764$)。MAACO 获得的标准差为 0, 这意味着 MAACO 在 20 次独立操作中总能获得最佳路径长度 (28.63)。均值和标准差的实验结果证明了所提出的 MAACO 具有良好的稳定性和鲁棒性。

此外, MAACO 获得的最优路径的转弯次数小于 ACO、A* 算法、Dijkstra 算法和 IACO 的转弯次数 ($7 < 9 = 9 = 9 < 16$), 与 IAACO 的相同。MAACO 的平均收敛迭代数小于 ACO 和 IACO ($5.3 < 9.3 < 27.4$)。虽然 MAACO 的最佳路径长度和转弯次数与 IAACO 相同, 但 MAACO 的收敛迭代次数的最佳值和平均值均小于 IAACO。图 7 也证明了这一结果, 并且 MAACO 的标准差具有明显的优越性, 如表 2 所列。MAACO 的执行时间比 ACO 更短。

然而, MAACO 和 ACO 的执行时间略长于 A* 和 Dijk-

stra。这是由于 A* 和 Dijkstra 算法都是直接搜索算法, MAACO 和 ACO 是迭代搜索算法。通常, 由于搜索机制的不同, 与直接搜索算法相比, 迭代搜索算法需要更多的时间来找到最优路径。考虑到最优路径在路径长度、收敛生成次数和转弯次数方面的综合性能, 表 2 所列的实验结果表明, 与 ACO, IAACO, IAACO, A* 算法和 Dijkstra 算法相比, MAACO 具有优越的性能。图 5 和 6 也证实了这一结果, 这表明 MAACO 获得的最佳路径是枚举算法中最好的。可以发现, 与 A* 算法和 Dijkstra 算法相比, MAACO 获得的最优路径转弯次数更少。事实上, MAACO 获得的最优路径的高质量主要得益于所提出的启发式函数和状态转移概率规则。最优路径的高效搜索得益于引入的具有方向信息和初始信息素分布不均匀的启发式机制。

从上述结果可以得出结论, 与 ACO, IAACO, IAACO, A* 算法、Dijkstra 算法相比, MAACO 能够以更高的效率获得更优的最优解。要进一步证明 MAACO 的优越性, 应该采用更多不同的案例, 并使用更多的方法进行比较。

4.2 模拟实验 2

为了测试 MAACO 的性能, 使用了 3 种不同的方法以及文献[23]中提到的一个复杂实例。这 3 种方法是基于种群信息熵的自适应改进蚁群算法 (AIACSE)、动态多角色自适应协同蚁群优化算法 (MRACO) 和粒子群优化算法 (PS-ACO)。我们还将它们与传统的 ACO、A* 算法和 Dijkstra 算法进行了比较。为了保证公平性, 使用了与文献[23]中相同的 30×30 空间环境模型 (如图 8 所示)。红色点表示起点, 蓝色点表示目标点。起点的坐标为 (0.5, 29.5), 目标点的坐标为 (29.5, 0.5)。同时, MAACO, ACO, PS-ACO, MRACO 和 AIACSE 的常用参数设置相同。详细地说, MAACO 的参数设置为 $M=60, K=200, \rho=0.2, \alpha=1.1, \beta=7, Q=100, q_{0_initial}=0.5$ 。AIACSE 的参数设置为 $M=60, K=200, \rho=0.2, \alpha=1.1, \beta=7, Q=100, q_{0_min}=0.4, q_{0_max}=0.9$ 。PS-ACO 的参数设置为 $M=2000, K=2000, \rho=0.015, \alpha=1, \beta=2, Q=100$ 。我们使用了表 3 中列出的参数来进行模拟实验。每种算法独立运行 20 次以解决该问题实例。表 4 中列出了 MAACO, ACO, PS-ACO, MRACO 和 AIACSE 的实验结果, 同时也包括了 A* 算法和 Dijkstra 算法的结果。图 8—图 11 展示了 5 种算法生成的最佳路径, 而图 12 和图 13 展示了 A* 算法、Dijkstra 算法和 MAACO 生成的最优路径。

表 3 模拟实验中使用的算法参数

Table 3 Parameters of algorithms used in simulation experiments

算法	M	K	α	β	Q	ρ	q_{0_min}	q_{0_max}	$q_{0_initial}$	α	w_{h_max}	w_{h_min}
ACO	60	200	1.1	7	100	0.200	—	—	—	—	—	—
PS-ACO	2000	2000	1.0	2	100	0.015	—	—	—	—	—	—
MRCACO	60	2000	1.0	4	1	0.100	—	—	—	—	—	—
AIACSE	60	200	1.1	7	100	0.200	0.4	0.9	—	—	—	—
MAACO	60	200	1.1	7	100	0.200	—	—	0.5	1	0.9	0.2

表 4 ACO, A* 算法和 Dijkstra 算法的变体生成的实验结果

Table 4 Experimental results for variant generation of ACO, A* and Dijkstra's algorithms

算法	路径长度			收敛迭代次数			转弯次数
	最佳长度	平均值	标准差	最佳值	平均值	标准差	
ACO	43.3553	47.9413	1.3199	17	27.4	5.5353	13
PS-ACO	43.3553	45.4234	1.7904	6	9.3	3.6069	11
MRCACO	42.7696	42.7696	0	3	4.0	0.8944	9
AIACSE	44.5269	44.5269	0	—	—	—	17
A*	42.7696	42.7696	0	—	—	—	9
Dijkstra	42.7696	42.7696	0	—	—	—	9
MAACO	42.7696	42.7696	0	5	5.3	0.2202	7

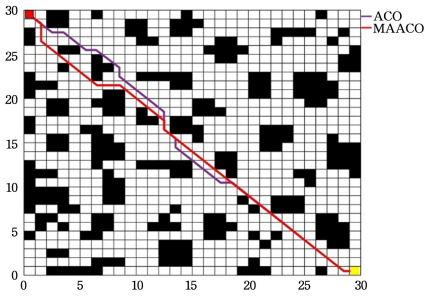


图 8 ACO 和 MAACO 生成的最佳路径

Fig. 8 Optimal path generated by ACO and MAACO

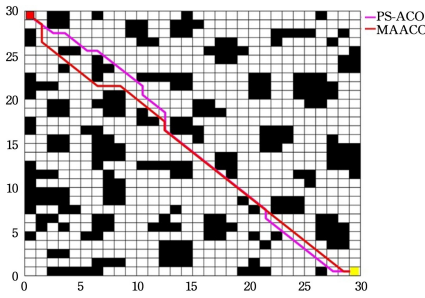


图 9 PS-ACO 和 MAACO 生成的最佳路径

Fig. 9 Optimal path generated by PS-ACO and MAAC

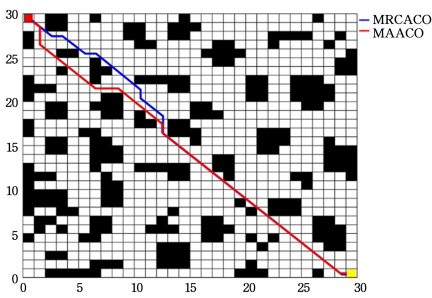


图 10 MRCACO 和 MAACO 生成的最佳路径

Fig. 10 Optimal path sgenerated by MRCACO and MAACO

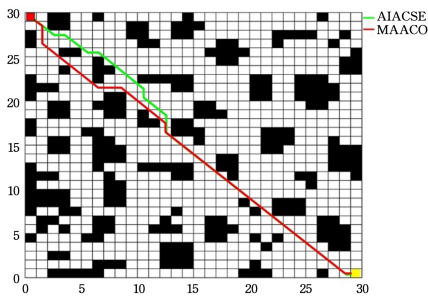


图 11 AIACSE 和 MAACO 生成的最佳路径

Fig. 11 Optimal path generated by AIACSE and MAACO

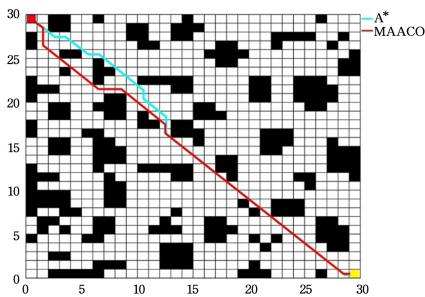


图 12 A* 和 MAACO 生成的最佳路径

Fig. 12 Optimal path sgenerated by A* and MAACO

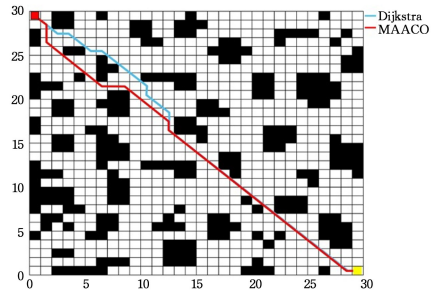


图 13 Dijkstra 和 MAACO 生成的最佳路径

Fig. 13 Optimal paths generated by Dijkstra and MAACO

根据表 4 的结果,可以看出 MAACO 相比于 ACO、A* 算法、Dijkstra 算法、PS-ACO、MRACO 和 AIACSE 的性能有所提升。虽然 MAACO、A* 算法、Dijkstra 算法和 MRACO 产生的最优路径长度相同,但 MAACO 在生成最优路径的转弯次数方面最少($7 < 9 = 9 = 9 < 11 < 13 < 17$)。而且,MAACO 的最优路径长度和平均路径长度均为 42.769,这表明了 MAACO 的优越性能。此外,在最优路径长度、转弯次数和平均路径长度方面,MAACO 相比于 ACO 和 PS-ACO 以及 AIACSE 表现出明显的优势。

MAACO 的优越性能得益于提出的 4 种机制,包括具有方向信息的启发式机制、改进的启发式函数、改进的状态转移概率规则和不均匀分布的初始信息素。图 8—图 13 也证明了 MAACO 的优越性能。总而言之,与其他几种对比算法相比,MAACO 能够生成更优的最优解,优势更明显。

4.3 模拟实验 3

本节将采用传统蚁群算法、A* 算法、Dijkstra 算法、文献[24]中介绍的人工势场蚁群优化算法(APFACO)和文献[25]中提出的方法改进蚁群算法(IACO)进行比较。空间环境模型设置为 20×20 ,与文献[25]中相同。如图 14 所示,红点表示起始点 S,黄点表示目标点 T,它们的直角坐标分别为 $(0.5, 19.5)$ 和 $(19.5, 0.5)$ 。为了比较不同的算法,将 ACO、APFACO、文献[25]中的算法 IACO 和 MAACO 的迭代次数 K 设为 100。文献[24]中 APFACO 和算法的其余参数设为 $M=50, \alpha=1.1, \beta=6, Q=100, \rho=0.2, \lambda=0.2, N_0=5, q_0=0.3$ 。MAACO 的其余参数设为 $M=50, \alpha=1.1, \beta=6, Q=100, \rho=0.2$ 。

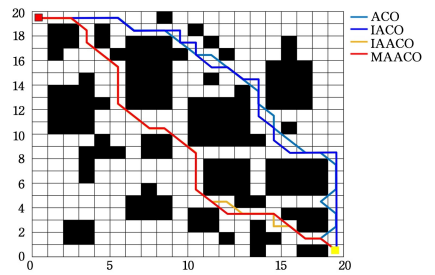


图 14 4 种蚁群算法生成的最优路径

Fig. 14 Optimal paths generated by four antcolony algorithms

实验结果表明,MAACO 在性能上表现优异。尽管 MAACO 和 IACO 生成的最优路径长度最短,但 MAACO 在所有对比算法中具有最小的平均收敛生成次数($5.3 < 7.7 < 11.4 < 27.4$)。此外,MAACO 生成的回合次数最少($13 < 14 < 15 = 15 < 19$)。总体而言,MAACO 在最优路径长度、转弯次数和收敛生成次数方面的表现最优。图 14 和

图 15 也验证了这一结果。通过实验结果对比,可以看出 MAACO 更加稳定且收敛速度更快(见图 16)。综上所述,MAACO 的综合性能进一步证明了其优越性。

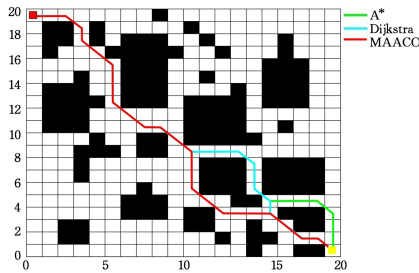


图 15 A* 算法、Dijkstra 算法和 MAACO 生成的最优路径

Fig. 15 Optimal paths generated by A* algorithm, Dijkstra's algorithm and MAACO

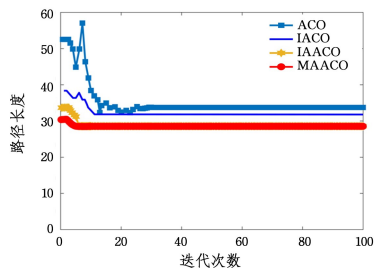


图 16 4 种蚁群算法的收敛迭代曲线

Fig. 16 Convergence iteration curves of four ant optimal path colony algorithms

表 5 6 种算法生成的实验结果

Table 5 Experimental results generated by six algorithms

算法	路径长度			收敛迭代次数			转弯次数
	最佳长度	平均值	标准差	最佳值	平均值	标准差	
ACO	31.80	33.7640	1.0484	17	27.4	3.3372	19
APFACO	35.07	35.6573	—	6	7.7	—	15
IACO	30.97	30.9700	0	6	11.4	—	15
A*	30.97	30.9700	0	—	—	—	15
Dijkstra	32.14	32.1400	0	—	—	—	14
MAACO	30.97	30.9700	0	5	5.3	0.2036	13

结束语 针对传统蚁群算法在移动机器人路径规划中的不足,提出了一种改进的自适应蚁群算法(MAACO)。首先,提出了一种带有方向信息的启发式机制,使算法具有方向智能;其次,通过引入规划路径的起点、目标点和转弯次数的位置信息,对启发式函数进行改进;然后,提出了一种自适应伪随机传输规则来选择下一个点。在此基础上,提出了一种初始信息素不均匀分布的新方法,增强了初始信息素的引导能力。将 4 种机构整合后,形成了 MAACO,并对 MAACO 的主要参数进行优化。

随后,进行了 3 组仿真实验,将 MAACO 与几种现有的改进蚁群方法和两种智能算法进行了比较,即 A* 算法和 Dijkstra 算法。实验结果表明,MAACO 不仅能够生成高质量的最优解,而且在收敛速度和转弯次数方面具有明显的优势。在一般情况下,与已有的方法相比,MAACO 的最优路径是最平滑的、调优次数最少的路径。

此外,MAACO 得到的平均路径长度表明了 MAACO 的稳定性。从以上实验可以看出,与其他算法相比,MAACO 能

够以更高的效率和更高的质量获得相应实例环境下的最优解。

参考文献

- [1] FOUNTAS N A, VAXEVANIDIS N M, STERGIOUS C I, et al. A virus evolutionary multi-objective intelligent tool path optimization methodology for 5axis sculptured surface CNC machining[J]. *Engineering With Computers*, 2016, 33(7): 375-391.
- [2] WANG Q, TANG C L. Deep reinforcement learning for transportation network combinatorial optimization: A survey [J]. *Knowledge-Based Systems*, 2021 233(2): 231-239.
- [3] TAKWA T, SAOUSSEN K. A Simulated annealing - based recommender system for solving the tourist trip design problem [J]. *Expert Systems with Applications*, 2021, 186: 115723-115731.
- [4] HUYNH T, PHAM D, THANG T. New approach to solving the clustered shortest-path tree problem based on reducing the search space of evolutionary algorithm [J]. *Knowledge-Based Systems*, 2019, 180(4): 12-25.
- [5] CHEN Y Q, GUO J L, YANG H D. Research on navigation of bidirectional A* algorithm based on ant colony algorithm[J]. *The Journal of Supercomputing*, 2021, 77(2): 1958-1975.
- [6] OROZCO ROSAS U, PICOS K, PANTRIGO J J. Mobile robot path planning using a QAPF learning algorithm for known and unknown environments [J]. *IEEE Access*, 2022, 10: 84648-84663.
- [7] LI Y J, WEI W, GAO Y, et al. PQ-RRT*: An improved path planning algorithm for mobile robots[J]. *Expert Systems with Applications*, 2020, 152: 113425-113436.
- [8] ZHAO Y J, ZHENG Z, LIU Y. Survey on computational-intelligence-based UAV path planning [J]. *Knowledge-Based Systems*, 2018, 158(5): 54-64.
- [9] LYU D D, CHEN Z W, CAI Z S. Robot path planning by leveraging the graph-encoded Floyd algorithm[J]. *Future Generation computer Systems*, 2021, 122(7): 204-208.
- [10] HOLLAND J H. *Adaptation in natural and artificial systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*[M]. Ann Arbor: University of Michigan Press, 1975.
- [11] CHONG Y, CHAI H Z, LI Y H. Automatic recognition of geomagnetic suitability areas for path planning of autonomous underwater vehicle[J]. *Marine Geodesy*, 2021, 44(4): 1-15.
- [12] KATHEN M, FLORES I J, REINA D G. An informative path planner for a swarm of asvs based on an enhanced PSO with gaussian surrogate model components intended for water monitoring applications[J]. *Electronics*, 2021, 10(13): 1605-1609.
- [13] DORIGO M, GAMBARDELLA L M. Ant colony system: A cooperative learning approach to the traveling salesman problem [J]. *IEEE Transactions on Evolutionary Computation*, 1997, 1(1): 53-66.
- [14] DORIGO M, BIRATTARI M, STUTZLE, T. Ant colony optimization: artificial ants as a computational intelligence technique [J]. *IEEE Computational Intelligence Magazine*, 2006, 1(6): 28-39.
- [15] FATEMIDOKHT H, RAFSANJANI M K. F-Ant: An effective routing protocol for ant colony optimization based on fuzzy logic

- in vehicular ad hoc networks[J]. *Neural Computing and Applications*, 2018, 29(6): 127-137.
- [16] WANG X Y, YANG L, ZHANG Y, et al. Robot path planning based on improved potential field ant colony algorithm [J]. *Control and Decision*, 2018, 33(10): 1775-1781.
- [17] ZHU Y, YOU X M, LIU S, et al. Research on Robot Path Planning Problem Based on Improved Ant Colony Algorithm [J]. *Computer Engineering and Applications*, 2018, 54(19): 129-134.
- [18] HUI T. Research on robot optimal path planning method based on improved ant colony algorithm [J]. *International Journal of Computing Science and Mathematics*, 2021, 13(1): 80-89.
- [19] MIAO C W, CHEN G Z, YAN C L. Path planning optimization of indoor mobile robot based on adaptive ant colony algorithm [J]. *Computers and Industrial Engineering*, 2021, 156(6): 107230-107241.
- [20] TAO Y, GAO H, REN F. A mobile service robot global path planning method based on ant colony optimization and fuzzy control [J]. *Applied Sciences*, 2021, 11(8): 3605-3621.
- [21] HSU C H, JUANG C H. Multi-Objective continuous Ant-Colony-Optimized FC for robot Wall-Following control [J]. *Computational Intelligence Magazine IEEE*, 2013, 8(3): 28-40.
- [22] LI X X, HU P. Robot 3D Path Planning Algorithm Based on Improved Elitist Potential Field Ant Colony Algorithm [J]. *Computer Science and Application*, 2021, 11(4): 849-858.
- [23] ZHANG S C, PU J, SI Y N. An adaptive improved ant colony system based on population information entropy for path planning of mobile robot [J]. *IEEE Access*, 2021, 9: 24933-24945.
- [24] LIU J H, YANG J G, GENG P. Robot global path planning based on ant colony optimization with artificial potential field [J]. *Transactions of The Chinese Society of Agricultural Machinery*, 2015, 46(9): 18-27.
- [25] LUO Q, WANG H B, ZHENG Y. Research on path planning of mobile robot based on improved ant colony algorithm [J]. *Neural Computing and Applications*, 2020, 32: 1555-1566.



WEI Shuxin, born in 1999, master. His main research interest is mobile robot path planning.



WANG Qunjing, born in 1960, Ph. D., professor. His main research interests include electrical machines and their control systems, power electronics and electric drives.