



改进蚁群算法的送餐机器人路径规划

蔡军, 钟志远

引用本文:

蔡军, 钟志远. 改进蚁群算法的送餐机器人路径规划[J]. 智能系统学报, 2024, 19(2): 370–380.

CAI Jun, ZHONG Zhiyuan. Path planning of a meal delivery robot based on an improved ant colony algorithm[J]. *CAAI Transactions on Intelligent Systems*, 2024, 19(2): 370–380.

在线阅读 View online: <https://dx.doi.org/10.11992/tis.202205056>

您可能感兴趣的其他文章

基于变步长蚁群算法的移动机器人路径规划

Mobile robot path planning based on variable-step ant colony algorithm

智能系统学报. 2021, 16(2): 330–337 <https://dx.doi.org/10.11992/tis.202004011>

多配送中心下生鲜农产品同步取送选址-路径优化

Fresh agricultural cargoes location-routing optimization with simultaneous pickup and delivery for multiple distribution centers

智能系统学报. 2020, 15(1): 50–58 <https://dx.doi.org/10.11992/tis.201905042>

应用改进区块遗传算法求解置换流水车间调度问题

An improved puzzle-based genetic algorithm for solving permutation flow-shop scheduling problems

智能系统学报. 2019, 14(3): 541–550 <https://dx.doi.org/10.11992/tis.201801041>

基于改进粒子群算法的移动机器人多目标点路径规划

Mobile robot multi-goal path planning using improved particle swarm optimization

智能系统学报. 2017, 12(3): 301–309 <https://dx.doi.org/10.11992/tis.201606046>

基于蚁群算法的四旋翼航迹规划

Four-rotor route planning based on the ant colony algorithm

智能系统学报. 2016, 11(2): 216–225 <https://dx.doi.org/10.11992/tis.201509009>

蚁群优化算法的理论研究进展

Advances in theoretical research of ant colony optimization

智能系统学报. 2016, 11(1): 27–36 <https://dx.doi.org/10.11992/tis.201510002>

DOI: 10.11992/tis.202205056

网络出版地址: <https://link.cnki.net/urlid/23.1538.TP.20231116.1412.006>

改进蚁群算法的送餐机器人路径规划

蔡军, 钟志远

(重庆邮电大学 自动化学院, 重庆 400065)

摘要: 蚁群算法拥有良好的全局性、自组织性、鲁棒性, 但传统蚁群算法存在许多不足之处。为此, 针对算法在路径规划问题中的缺陷, 在传统蚁群算法的状态转移公式中, 引入目标点距离因素和引导素, 加快算法收敛性和改善局部最优缺陷。在带时间窗的车辆路径问题(vehicle routing problem with time windows, VRPTW)上, 融合蚁群算法和遗传算法, 并将顾客时间窗宽度以及机器人等待时间加入蚁群算法状态转移公式中, 以及将蚁群算法的解作为遗传算法的初始种群, 提高遗传算法的初始解质量, 然后进行编码, 设置违反时间窗约束和载重量的惩罚函数和适应度函数, 在传统遗传算法的交叉、变异操作后加入了破坏-修复基因的操作来优化每一代新解的质量, 在 Solomon Benchmark 算例上进行仿真, 对比算法改进前后的最优解, 验证算法可行性。最后在餐厅送餐问题中把带有障碍物的仿真环境路径规划问题和 VRPTW 问题结合, 使用改进后的算法解决餐厅环境下送餐机器人对顾客服务配送问题。

关键词: 蚁群算法; 遗传算法; 状态转移公式; 适应度函数; 引导素; 局部最优; 初始种群; 时间窗约束; 路径规划
中图分类号: TP273 **文献标志码:** A **文章编号:** 1673-4785(2024)02-0370-11

中文引用格式: 蔡军, 钟志远. 改进蚁群算法的送餐机器人路径规划 [J]. 智能系统学报, 2024, 19(2): 370-380.

英文引用格式: CAI Jun, ZHONG Zhiyuan. Path planning of a meal delivery robot based on an improved ant colony algorithm[J]. CAAI transactions on intelligent systems, 2024, 19(2): 370-380.

Path planning of a meal delivery robot based on an improved ant colony algorithm

CAI Jun, ZHONG Zhiyuan

(College of Automation, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

Abstract: The ant colony algorithm has advantages, such as good global ability, self-organization, and robustness, but the conventional ant colony algorithm has several drawbacks. Herein, with the goal of addressing these drawbacks, in the path planning problem, according to the conventional ant colony algorithm, the distance factor of the target point and the guide element are added to the state transition formula to optimize the convergence of the ant colony algorithm in the path planning and enhance the defect of local optimal point. On the vehicle routing problem with time window (VRPTW), the ant colony algorithm and genetic algorithm are combined, and the customer time window width and robot waiting time are added to the state transition formula of the ant colony algorithm. The solution of the ant colony algorithm is taken as the initial population of the genetic algorithm to enhance the quality of the initial solution of the genetic algorithm, and subsequently, coding is conducted. The penalty function and fitness function violating the time window constraint and load are set. After the mutation and crossover operation of the conventional genetic algorithm, the damage and repair gene operation is added to optimize the quality of each generation of new solutions. Simulations are performed on the Solomon benchmark example, and the optimal solutions before and after enhancement of the algorithm are compared to confirm the feasibility of the algorithm. To apply this to a real-world problem, the path planning problem in a simulative environment with obstacles and the VRPTW problem are joined using the enhanced algorithm to solve the customer service delivery problem of the restaurant delivery robot.

Keywords: ant colony algorithm; genetic algorithm; state transition formula; fitness function; guiding element; local optimum; initial population; time window constraint; path planning

收稿日期: 2022-05-31. 网络出版日期: 2023-11-17.

基金项目: 国家自然科学基金项目 (61673079, 61703068); 重庆市基础研究与前沿探索项目 (cstc2018jcyjAX0160).

通信作者: 蔡军. E-mail: 1073230317@qq.com.

餐饮服务越来越强调无接触式的用餐服务, 而智能送餐机器人能够完美解决餐馆的无接触送餐服务的问题。这类机器人的优点在于开发高

效、使用方便,对硬件要求不高^[1]。但是现在市面上的送餐机器人也还存在不足之处,一些普通餐厅中的送餐机器人仅仅能实现简单循迹。而且现在大多数的送餐机器人只能单对单的服务,这种机器人需要前台在上位机上进行操作才能送达,并不能对多位顾客的需求和等待时长进行实时分析。因此,设计一种能够对餐厅多位顾客的需求和点餐顺序进行分析并规划出合理的配送顺序和最优路径的算法具有一定意义。机器人规划路径的优劣需要综合考虑多种因素,如路径长度、运行时间和算法的迭代次数等^[2-4]。早期使用的路径规划方法有人工势场法^[5]、A*算法^[6]和Dijkstra算法^[7]等,这些方法在实际应用中表现出了很大的不足。遗传算法^[8-9]、蚁群算法^[10]、粒子群算法^[11-12]、萤火虫算法^[13]、果蝇优化算法^[14]和灰狼算法^[15]等属于智能优化算法,与传统路径最优算法相比也有很多传统算法没有的优势。蚁群算法作为经典智能优化算法,因其正反馈性和并行性等优势,被广泛应用于移动机器人路径规划问题中^[16-18]。Cao^[19]在改进的蚁群算法中,每个周期的一些短路径上增加了信息素数量,并动态调整了其蒸发率,从而提高转移概率,结果证明算法的全局寻优能力增强,搜索速度明显提高。Tsai等^[20]提出了一种新的多目标灰狼优化算法解决移动机器人路径规划问题。张毅等^[21]将A*算法和改进的人工鱼群方法相结合,先采用A*算法规划出次优路径,然后在基本鱼群方法中提出了非线性惯性权重因子的自适应行为,引入衰减函数,优化算法的全局搜索能力和局部搜索能力,得到更优的路径,且收敛速度也更快。Brand等^[22]将萤火虫算法应用在动态空间中移动机器人路径规划,仿真证明这个算法能够成功规划出可行路径,并且在长度和计算成本方面都较低。

虽然上述改进策略能够在一定程度上改善蚁群算法的不足,但蚁群算法的收敛速度慢和易陷入局部最优仍是该算法的缺陷。针对以上缺陷,本研究提出了一种蚂蚁引导素,引导素能够让蚂蚁对自己走过的路径进行分析,对规划效果更好的路径留下引导素,能让后续蚂蚁倾向于更优的路径,避开相对较差的路径来改进局部最优的缺陷。

而餐厅多位顾客的配送顺序实际就是解决带时间窗的车辆路径(vehicle routing problem with time windows, VRPTW)问题。VRPTW问题可描述为:若干车辆从某个确定的节点出发,为该节点周围的其他节点提供一次服务(如配送货物),已知各个节点处在不同的地理位置,并且具有不

同的货物需求和不同的服务时间段(即时间窗)要求,车辆服务完毕返回始发节点。在该问题上,本研究将蚁群算法和遗传算法进行融合,用蚁群算法的初始解代替遗传算法的初始种群,并提出一种破坏基因和修复基因的操作,利用相关性公式找出所有基因与被破坏掉的基因的相关性,对基因进行删减,然后利用最远插入思想将删掉的基因进行插回,提高了算法鲁棒性。通过破坏基因和修复基因的操作能够有效提高组合优化问题的求解质量。

将上述的路径规划和配送顺序问题结合就是本研究拟解决的送餐机器人的路径规划问题。通过对比试验证明了改进后算法能够规划出最优配送方案以及最优的实际路线。

1 蚁群算法

蚁群算法(ant colony, ACO)最初是由意大利的研究学者Dorigo等^[23-24]在1992年期间第一次提出来的。蚂蚁在寻找食物的过程中,会在每条走过的路径下留下自己的信息素从而来引导其他的蚂蚁。随着各个蚂蚁经过不同的路径到达食物点时,每条路径上留下的信息素浓度各不相同,后面的蚂蚁更多地会选择信息素浓度更高的一条路径去移动。其中每条路径信息素的浓度与经过该路径的蚂蚁数量和路径长短有关。当路径较远时,同一只蚂蚁留下的信息素浓度越低,经过蚂蚁不断探索路径,最终蚂蚁会更加倾向信息素浓度最高路径。蚂蚁们最终通过这种方式来找到一条通往食物的最短路径。在 t 时刻蚂蚁从位置 i 到未知位置 j 的转移概率为

$$p_{ij}^k = \begin{cases} \frac{\tau_{ij}^\alpha(t)\eta_{ij}^\beta(t)}{\sum_{s \in A_k} \tau_{is}^\alpha(t)\eta_{is}^\beta(t)}, & j \in A_k \\ 0, & \text{其他} \end{cases} \quad (1)$$

式中: τ_{ij}^α 为信息素; α 为信息素的启发因子; η_{ij}^β 为节点 i 到节点 j 的期望函数; β 为期望启发因子; A_k 为蚂蚁可以选择的下一个点的所有集合。其中期望函数表达式为

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

式中 d_{ij} 为 i, j 两地之间的距离。

信息素的更新公式为

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

$$\Delta\tau_{ij}(t+n) = \sum_{k=1}^K \Delta\tau_{ij}^k \quad (4)$$

式中: ρ 为信息素挥发系数; $\Delta\tau_{ij}^k$ 为信息素增量,信

息素增量具体表达式为

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{L_k}, & \text{蚂蚁 } k \text{ 从节点 } i \text{ 至 } j \\ 0, & \text{其他} \end{cases} \quad (5)$$

式中: Q 为一个常数; L_k 为每一只蚂蚁从起点走到目的地的总路程。通过该方式进行信息素增量的计算加强了蚁群算法的全局性。

2 改进应用于路径规划的蚁群算法

2.1 考虑目标点距离因素的概率转移函数

在传统的蚁群算法转移概率式 (1) 中的启发信息 η_{is}^β 是当前蚂蚁所在点到下一步的各个可选点之间的距离的倒数。这样的转移概率没有考虑到蚂蚁的下一个可选点离目标点的距离, 算法缺乏全局性。因此将可选点集合中各个点与目标点的欧几里得距离的倒数代替该期望函数。改进后的概率公式为

$$p_{ij}^k = \begin{cases} \frac{\tau_{ij}^\alpha(t)[d^{-1}(j,e)]^\beta}{\sum_{s \in A_k} \tau_{is}^\alpha(t)[d^{-1}(s,e)]^\beta}, & j \in A_k \\ 0, & \text{其他} \end{cases} \quad (6)$$

式中: $d(j, e)$ 为可选点 j 与目标点 e 之间的欧几里得距离。

2.2 加入引导素改进局部最优

蚁群算法中使用的轮盘赌法使得该算法在复杂环境下, 有低概率选择相对较差的位置。由于蚁群算法的信息素正反馈机制会使得蚂蚁在该位置留下信息素, 导致后代蚂蚁选择该位置的几率增加, 反而降低了最优解质量。针对该缺陷, 提出一种引导素。当第一只蚂蚁到达终点之后开始分析从起点到达目的地的路径上有无相对较差的点位, 对这些点位, 找到适当的新路线并留下一一种引导素, 以此来引导后续的蚂蚁在寻找最优路径时避开那些相对较差的点位, 从而提高最优解的质量, 改善蚁群算法陷入局部最优的问题。加入引导素后蚁群算法的状态转移概率公式为

$$p_{ij}^k = \begin{cases} \frac{\tau_{ij}^\alpha(t)[d^{-1}(j,e)]^\beta v_{ij}^\gamma}{\sum_{s \in A_k} \tau_{is}^\alpha(t)[d^{-1}(s,e)]^\beta v_{is}^\gamma}, & j \in A_k \\ 0, & \text{其他} \end{cases} \quad (7)$$

式中: v_{ij} 代表 i 点到 j 点的引导素浓度; γ 为引导素启发因子。

引导素对于相对较差点位的更改情况可分为 3 类。第 1 种情况如图 1 所示, 虚线为蚂蚁实际路线。若蚂蚁从栅格 1 直接前往栅格 3 的距离小于实际路线距离, 则增加栅格 1 到栅格 3 的引导素浓度。

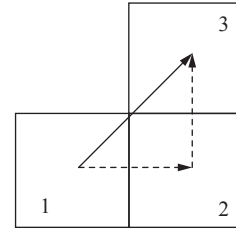


图 1 第 1 种情况

Fig. 1 Case 1

公式可以表示为

$$d_{1-3} < d_{1-2} + d_{2-3}, \quad v_{13} = v_{13} + \Delta v \quad (8)$$

式中: d 为两点之间的距离; Δv 为增加的引导素。

第 2 种情况如图 2 所示, 原理与第 1 种情况一样且公式也可以用式 (8) 表示。

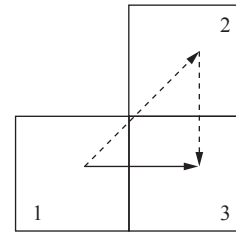


图 2 第 2 种情况

Fig. 2 Case 2

第 3 种情况如图 3 所示, 在前面 2 种的基础上需要多出一个对格子 1 和格子 3 之间的中间格子进行是否具有障碍物的判定, 如果中间格子为可通行格, 则增加格子 1 到格子 4 的引导素浓度; 反之则保持原始的引导素浓度。公式如下所示

$$d_{1-3} < d_{1-2} + d_{2-3} \ \&\& G(\text{mid}) = 1, \quad v_{14} = v_{14} + \Delta v \quad (9)$$

式中: d 为两点之间的距离; G 表示格子的障碍物状态; 0 表示可通行状态; 1 表示存在障碍物; $\&\&$ 表示与运算; mid 表示节点 1 与节点 3 直线上的中间节点。

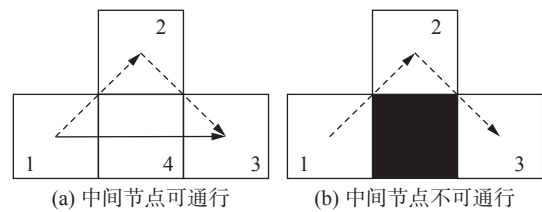


图 3 第 3 种情况

Fig. 3 Case3

2.3 蚁群算法参数整定

为了整定蚁群算法的试验参数, 首先采用文献 [25] 的试验参数, 挥发系数 $\rho=0.3$ 、信息素启发因子 $\alpha=2$ 、启发信息影响因子 $\beta=6$ 。随后用控制变量法, 控制如: 挥发因子 ρ 、信息素启发因子 α 、启发信息影响因子 β 等参数值固定不变, 依次改变

α 的取值为 $\{1.0, 1.5, 2.0, 2.5, 3.0\}$, 分批进行 20 次试验, 其余参数为固定值将 20 次的 5 批数据取其平均值, 不同 α 值的平均最优路径长度随平均迭代次数的变化图如图 4 所示。

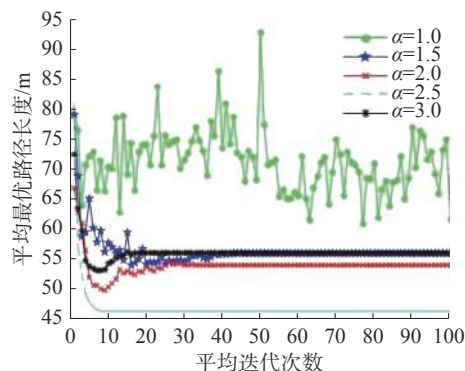


图 4 不同 α 参数实验结果

Fig. 4 Different α parameter experiment results

由图 4 中可以看出, 当 $\alpha=2.5$ 时, 算法得到的平均最优路径同时也是最快收敛的, 于是取 $\alpha=2.5$ 。

随后取 $\alpha=2.5$, 其他参数值固定不变, 依次改变 β 的取值为 $\{4, 5, 6, 7, 8\}$, 分批进行 20 次试验, 不同 β 的平均最优路径长度随平均迭代次数的变化图如图 5 所示。

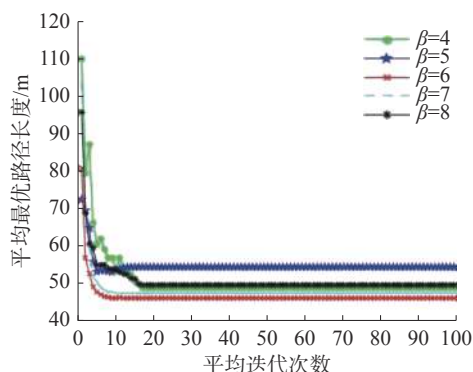


图 5 不同 β 参数实验结果

Fig. 5 Different β parameter experiment results

由图 5 中可以看出, 当 $\beta=6$ 时, 算法得到的最优路径长度最小同时也是收敛最快的, 于是取 $\beta=6$ 。

随后取 $\alpha=2.5, \beta=6$, 其他参数值固定不变, 依次改变 ρ 的取值为 $\{0.1, 0.2, 0.3, 0.4, 0.5\}$, 进行分批多次试验, 不同 ρ 的平均最优路径长度随平均迭代次数的变化图如图 6 所示。

由图 6 中可以看出, 当 $\rho=0.3$ 时, 算法得到的平均最优路径同时也是最快收敛的, 于是取 $\rho=0.3$ 。

2.4 蚁群算法改进前后仿真对比

在构建 20×20 的栅格环境进行仿真对比验证, 以下所有对比试验都把起点设置为 $(0.5,$

$19.5)$, 终点设置为 $(19.5, 0.5)$, 试验中的信息素影响因子 $\alpha=1.5$ 、启发信息影响因子 $\beta=6$ 、引导素影响因子 $\gamma=6$, 挥发系数 $\rho=0.3$ 、信息素浓度 $Q=100$ 。在该栅格环境下分别使用基本蚁群算法和改进后的蚁群算法进行路径规划, 最优规划路径仿真结果如图 7 所示。各算法仿真结果的路径长度随迭代次数的变化过程对比如图 8 所示。

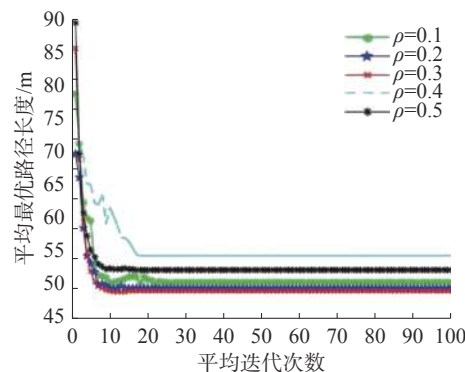


图 6 不同 ρ 参数实验结果

Fig. 6 Different ρ parameter experiment results

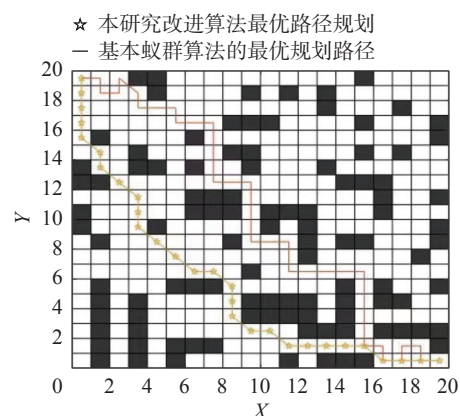


图 7 基本蚁群算法与改进后的蚁群算法对比

Fig. 7 Comparison between basic ant colony algorithm and improved ant colony algorithm

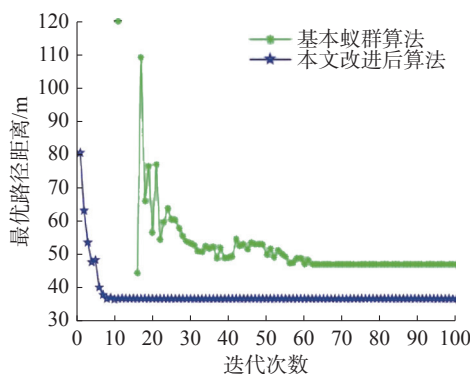


图 8 2 种不同算法对比

Fig. 8 Comparison of two different algorithms

通过基本蚁群算法和改进后蚁群算法的最优路径规划图以及各个算法的最优路径长度随迭代次数变化对比曲线可以看出, 基本蚁群算法经过

50多次迭代之后规划的最优路径规划效果十分不理想,这是因为基本蚁群算法在算法初期信息素浓度的缺乏,导致不能对蚂蚁进行有效的路径搜索引导,使得蚂蚁在前期的搜索陷入盲目状态,经过算法初期的盲目搜索导致许多不合理的路径上留下了一定量的信息素,使得算法中期蚂蚁在信息素的正反馈机制下极易陷入此局部最优而无法跳出。相较于基本蚁群算法,本研究改进后的算法在栅格环境中路径规划效果较好。因为该算法的状态转移公式考虑了目标点距离和引导素,让蚂蚁避免了许多相对较差的点位,改进了局部最优的缺陷。经计算,基本蚁群算法最优路径长度为48.623 m,本研究改进后算法最优路径长度为36.552 m。本研究改进后算法最优路径长度比基本蚁群算法减少了24.83%。

3 餐厅服务机器人配送问题

在餐厅中的服务机器人配送问题可以看作是一个经典的VRPTW问题。VRPTW问题可描述为:若干车辆从某个确定的节点出发,为该节点周围的其他节点提供一次服务(如配送货物),已知各个节点处在不同的地理位置,并且具有不同的货物需求和不同的服务时间段(即时间窗)要求,车辆服务完毕返回始发节点。设 K 为送餐中心拥有的机器人集合;设 N 为需要送餐服务的 n 个顾客点的集合, $N = \{1, 2, \dots, n\}$; V_k 为机器人 k 访问顾客的集合; $V = \{0\} \cup N$ 。设0为餐饮中心; i, j 表示顾客编号; x_{ij}^k 表示机器人 k 服务完顾客 i 以后是否服务顾客 j ; y_i^k 表示顾客 i 是否被机器人 k 服务; Q 表示服务机器人的装载能力约束; q_i 表示顾客 i 的需求量,且 $q_0 = 0$ 。 C_{ij} 为机器人从 i 点走向 j 点的运输成本,运输成本在不同的情况下,实际意义也不一样,需要以实际情况来确定运输成本。

餐厅服务机器人配送问题可以描述为

$$\min \sum_{k \in K} \sum_{i \in V_k} \sum_{j \in V_k \setminus \{i\}} C_{ij} x_{ij}^k \quad (10)$$

$$\sum_{j \in V_k \setminus \{i\}} x_{ij}^k = y_i^k, \forall i \in V, \forall k \in K \quad (11)$$

$$\sum_{i \in V_k \setminus \{j\}} x_{ij}^k = y_j^k, \forall j \in V, \forall k \in K \quad (12)$$

$$\sum_{i \in V_k} q_i y_i^k \leq Q, \forall k \in K \quad (13)$$

$$\sum_{j \in N} x_{0j}^k = \sum_{i \in N} x_{i0}^k = 1, \forall k \in K \quad (14)$$

$$\sum_{k \in K} y_i^k = 1, \forall i \in N \quad (15)$$

$$\sum_{i, j \in V_k} x_{ij}^k \leq |V_k| - 1, \forall k \in K \quad (16)$$

$$x_{ij}^k \in \{0, 1\}, \forall i, j \in V, \forall k \in K \quad (17)$$

$$y_i^k \in \{0, 1\}, \forall i \in N, \forall k \in K \quad (18)$$

目标函数(10)表示费用最小的目标函数;约束(11)表示机器人在服务完顾客 i 后直接服务一个顾客 j ;式(12)表示机器人在服务顾客 j 之前只服务一个顾客 i ;式(13)表示每个机器人 k 运送的顾客需求总量不超过机器人的最大容量;式(14)表示每个机器人从中心出发最终必须回到中心;式(15)指每个顾客 i 只能由一位机器人服务;式(16)是避免配送过程中产生子回路; x_{ij}^k 、 y_i^k 为决策变量。当机器人 k 服务完顾客 i 后直接服务顾客 j 时, $x_{ij}^k = 1$;否则 $x_{ij}^k = 0$ 。当顾客 i 由机器人 k 服务时, $y_i^k = 1$;否则 $y_i^k = 0$ 。

在餐厅服务机器人配送过程中涉及到时间窗问题。通常时间窗可分为软时间窗约束和硬时间窗约束,在餐厅送餐问题中没有涉及到惩罚费用,故在该问题中采用硬时间窗约束,即服务机器人一定要在顾客的容忍时间范围到达并开始服务。容忍时间范围可以用 $[E_{Ti}, L_{Ti}]$ 来表达,其中 E_{Ti} 是左时间窗,代表顾客能够容忍的最早服务的时间点; L_{Ti} 是右时间窗,代表顾客能容忍的最晚服务的时间点。

4 混合蚁群-遗传算法解决餐厅服务机器人配送问题

4.1 改进蚁群算法的概率转移函数

利用第2章的改进蚁群算法求解机器人中心点位(即配送餐饮中心)和所有餐位点互相之间的最短距离和最优路径并进行保存,本研究采用文献[26]在硬时间窗约束改进,由于顾客只接受在时间窗约束内的服务,所以当送餐机器人到达顾客 j 的时间 a_j 早于左时间窗 E_{Tj} ,意味着机器人需要等待时间 $w_{ij} = E_{Tj} - a_j$ 。除此之外还要考虑每位顾客 j 的时间窗宽度 $w_d = L_{Tj} - E_{Tj}$ 。由此可以改进蚁群算法,蚂蚁选择下一个顾客 j 的状态转移概率公式为

$$p_{ij}^k = \frac{\tau_{ij}^{\alpha'} \eta_{ij}^{\beta'} [1/w_{dj}]^{\varepsilon} [1/w_{tj}]^{\delta}}{\sum_{s \in N_{ik}} (\tau_{is}^{\alpha'} \eta_{is}^{\beta'} [1/w_{ds}]^{\varepsilon} [1/w_{ts}]^{\delta})} \quad (19)$$

式中: τ_{ij} 是机器人在顾客 i 选择下一个顾客 j 的信息素浓度; α' 是信息素启发因子; η_{ij} 是顾客 i 到顾客 j 的最短距离的倒数; β' 是最短距离启发因子; w_{dj} 是顾客 j 的时间窗宽度; ε 是时间窗宽度启发因子; w_{tj} 是机器人将要在顾客 j 等待的时间; δ 是机器人等待时间启发因子; N_{ik} 为可选顾客点集合。

4.2 混合蚁群遗传算法

4.2.1 提高初始种群质量

为了提高混合蚁群遗传算法的初始种群质量, 本研究提出了让蚂蚁自主构建当前的可行顾客点集合。蚂蚁 k 在顾客 i 时的可选顾客点集合 N_{ik} 的确定方式如图 9 所示。

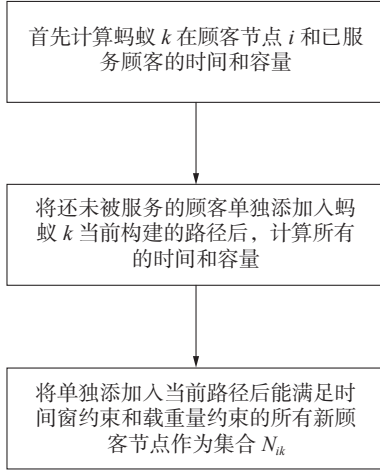


图 9 N_{ik} 的确定方法

Fig. 9 Method for determining N_{ik}

通过可行路径的构建方法, 让蚂蚁在选择下一位顾客时, 只会在考虑满足所有约束条件的顾客中进行选择。利用能够自主构建可选顾客节点的改进蚁群算法的解作为遗传算法的初始种群, 提高初始种群的质量。

4.2.2 编码操作

混合蚁群遗传算法需要对蚁群算法得到的解首先进行编码操作才能进行后续操作。本研究编码方式为当顾客数目为 N 且最大车辆使用数目为 K 时, 染色体长度为 $N+K-1$, 染色体表达形式为 $(1, 2, \dots, N, N+1, N+2, \dots, N+K-1)$ 。例如一共 5 位顾客, 3 辆车的第 1 条路径为 0-1-2-0, 第 2 条路径为 0-3-4-0, 第 3 条路径为 0-5-0。最终路径用染色体编码为 1263475, 其中 6 和 7 代表配送中心。

4.2.3 制定适应度函数

为了解决在混合蚁群遗传算法中经过变异、交叉操作之后不满足约束条件的情况, 制定惩罚函数, 惩罚函数为

$$f(s) = c(s) + \chi q(s) + \phi w(s) \quad (20)$$

式中: $c(s)$ 表示机器人总行走距离; $q(s)$ 表示各条路径违反的容量约束之和; $w(s)$ 表示所有顾客违反的时间窗约束之和; χ 表示违反载重量约束的惩罚系数; ϕ 表示违反时间窗约束的惩罚系数。

$$q(s) = \sum_{i=1}^m \max\{(L_i - c_{ap}), 0\} \quad (21)$$

$$w(s) = \sum_{i=1}^m \max\{(a_i - l_i), 0\} \quad (22)$$

式中: L_i 表示该机器人的载重量; c_{ap} 表示该机器人的最大载重量; a_i 表示机器人达到顾客的时间; l_i 表示顾客 i 的右时间窗。

通过惩罚函数可知, 该问题需要得到最小的惩罚函数, 所以适应度应该为惩罚函数的倒数, 适应度函数应该为 $1/f(s)$ 。

4.2.4 选择操作

对初始种群进行编码以及计算初始种群的适应度函数之后, 需要进行选择操作。通过对初始种群中每个个体都计算它的适应度函数值, 并对初始种群的适应度函数值进行归一操作, 通过轮盘赌法选择出种群中的 90% 的个体进行后续操作。

4.2.5 加入破坏基因操作

针对传统遗传算法经过变异和交叉操作之后的求解质量低的问题, 本研究提出破坏基因的操作, 对经过变异和交叉操作之后得到的新路径上的顾客进行移除。首先对机器人所有路径中随机选择一位顾客进行移除, 假如构建的 100 位顾客的配送首先移除顾客 9, 然后通过相关性计算公式, 计算出顾客 9 与其他 99 为顾客的相关性值 R 。相关性计算公式如下所示

$$R(i, j) = 1/(c'_{ij} + v_{ij}) \quad (23)$$

$$c'_{ij} = \frac{c_{ij}}{\max c_{ik}}, k \in N \quad (24)$$

式中: v_{ij} 表示顾客 j 与顾客 i 是否由同一个机器人服务, 如果是同一个机器人服务为 0, 否则为 1; c_{ij} 代表顾客 i 与 j 的距离; $\max c_{ik}$ 代表离顾客 i 最近的顾客 k 的距离; N 属于顾客集合。

由相关性公式可知, 顾客 i 与 j 的相关性的值越大则代表顾客 i 与顾客 j 的相关性越大。假如一共排除 10 个顾客, 首先随机排除一位, 通过计算已移除的第一位顾客与其他所有顾客的相关性值, 并进行归一处理通过轮盘赌法排除另外 9 个顾客点位。将该 10 位顾客从遗传算法种群中移除。

4.2.6 加入修复基因操作

修复基因的思想为最近插入启发式思想, 即求到移除的顾客依次添加回各个机器人服务的顾客中间的最小距离增量, 找到移除顾客中的最小距离增量中最大的顾客重新插回顾客服务顺序中。例如已经移除了 10 位顾客, 将移除的 9 号顾客点依次添加回每一个机器人服务的顾客点位中间去, 一共有 N_r+90 种情况, N_r 为机器人的数量。依次计算 N_r+90 种情况中机器人到达每位顾客的时间和该机器人的载重量, 判断插入顾客后

的机器人到达每位顾客的时间是否满足每位的时间窗约束和载重量约束,如果同时满足记录下插入顾客前后的距离增量,如果不满足其中之一的约束,就派出新的机器人进行服务该新增顾客。由此可以得到 N_r+90 个距离增量,从中找出最小距离增量的插入点和机器人序号。由此可以得到之前移除的 10 位顾客的最小距离增量。找到 10 位顾客中最小距离增量最大的移除顾客,将其重新添加回对应的最小距离增量的插入点和机器人序号。重新插回一位已移除的顾客后,把剩下的 9 位已移除的顾客代入新解中,继续重新计算的最小距离增量,再进行比较,重新插回种群中。直至 10 位顾客都重新插回到新解中得到新的 100 位顾客的新解。

4.3 混合蚁群-遗传算法解决配送顾客的算法步骤

1) 初始化顾客点信息、机器人信息以及蚁群算法和遗传算法实验参数。

2) 首先设置 m 只蚂蚁利用改进状态转移公式 (20) 来得到每只蚂蚁遍历完 100 位顾客点的机器人配送方案,由此得到 m 个 100 位顾客的机器人配送方案。

3) 对 m 个 100 位顾客的机器人配送方案进行编码,以此作为遗传算法的初始种群。

4) 对初始种群计算种群的各个目标函数值,并进行归一操作,运用轮盘赌法从 m 个基因片段中选择出 N_p 个进行交叉操作和变异操作得到新种群。

5) 对新种群进行解码,并随机移除一位顾客,并计算该移除顾客和其余 99 位顾客的相关性值,进行归一处理,用轮盘赌法得到一共移除的 D 个顾客。

6) 计算 D 个顾客重新插回到新解中的最小距离增量。并从 D 个最小距离增量中找到最大的增量值并插回到相应插入顾客点以及服务机器人序号,直到 D 个顾客被重新插回到新解中。

7) 重新将新解编码重插入子代形成新种群并删除种群中重复个体,并补齐删除的个体。

8) 是否达到遗传算法最大迭代值,如果没有达到就返回 4), 达到最大迭代值则停止算法并输出最优配送方案。

4.4 混合蚁群-遗传算法解决配送顾客的算法试验验证

针对传统的蚁群算法和遗传算法在求解大规模 VRPTW 问题质量低下的问题,本研究提出混合蚁群遗传算法,将蚁群算法初代解用作遗传算法的初始种群。随后在传统的遗传算法的变异交叉操作外新加入本研究提出的破坏-修复基因操

作,并引入基因相关性的因素丰富了求解的多样性,提高了求解质量。为了验证本研究改进算法的优越性,本次实验选用 Solomon Benchmark 中的 C101.100 的数据集进行仿真验证。对试验中应用最优路径的改进蚁群算法中的应用机器人对顾客服务分配顺序的蚁群算法中时间窗宽度启发因子 $\varepsilon=2$ 、机器人等待时间启发因子 $\delta=3$, 每台机器人的容量 $c_{ap}=200$ 。违反容量约束的惩罚函数系数为 1, 违反时间窗约束的惩罚函数系数为 6, 迭代次数为 100, 交叉概率 $P_c=0.9$, 变异概率 $P_m=0.05$ 。在参数一致的情况下,将基本蚁群算法、文献 [26] 算法、以及本研究改进算法在 C101.100 算例中进行对比分析,各算法的最小总距离迭代图如图 10 所示,机器人配送方案如表 1 所示。

表 1 机器人配送方案
Table 1 Robot distribution scheme

编号	路径顾客点
1	0 32 33 31 35 37 38 39 36 34 0
2	0 43 42 41 40 44 46 45 48 51 50 52 49 47 0
3	0 98 96 95 94 92 93 97 100 99 0
4	0 90 87 86 83 82 84 85 88 89 91 0
5	0 57 55 54 53 56 58 60 59 0
6	0 67 65 63 62 74 72 61 64 68 66 69 0
7	0 20 24 25 27 29 30 28 26 23 22 21 0
8	0 81 78 76 71 70 73 77 79 80 0
9	0 53 78 10 11 9 6 4 2 1 75 0
10	0 13 17 18 19 15 16 14 12 0

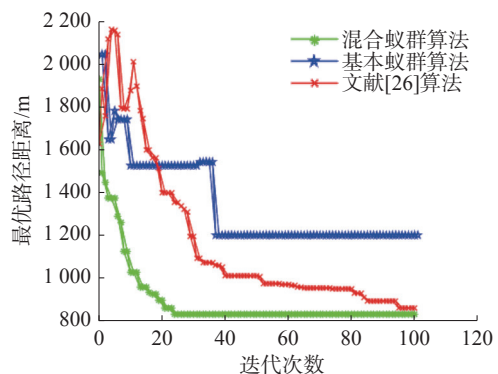


图 10 最小总距离迭代图

Fig. 10 Iteration diagram of minimum total distance

混合蚁群算法、基本蚁群算法以及文献 [26] 算法的车辆行驶最小总距离分别为 828.939 6、1200.526 8、860.205 1 m。在距离方面混合蚁群算法有了显著降低。在 C101.100 中,随着顾客数量变多,文献 [26] 算法最优解的质量出现下滑并且算法的收敛性也十分差,算法一直到迭代次数到了 80 代以后才收敛。由图 10 可以看出,本研究改

进算法无论是在最小总距离的长度还是在迭代次数方面都要比基本蚁群算法和文献[26]算法有所改善。

其次, Solomon Benchmark 的网站给出了各个算例历来使用启发式算法求解得到的最优路径,将其与本次改进后得到的最优解对比,在 C101.100 算例中历来最短行驶距离为 828.94 m,本研究算法的数据为 828.94 m,虽然在试验数据上的一些精确值上存在略微差异,但可以看出本研究改进的算法结果能够与历年最优路径值一致,证明了算法的可行性。

本研究改进算法在 C101.100 算例中得到的最优方案配送路线图如图 11 所示。其中得到的最优解为机器人使用数目为 10,车辆行驶总距离为 828.9369 m。配送路线如图 11 所示。

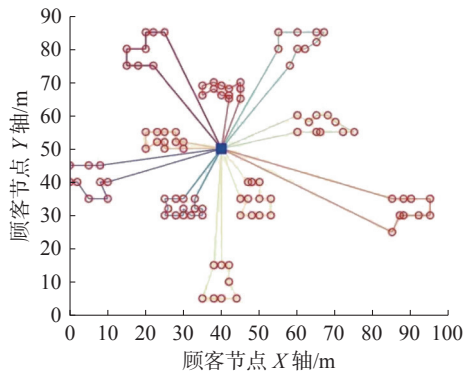


图 11 最优方案配送路线

Fig. 11 distribution route of the optimal scheme

5 餐厅环境下仿真试验

餐厅配送的问题实际就是基于将有障碍物的实际环境中每位顾客点之间和送餐中心的最短距离求出并记录实际路线,将有障碍物的实际最短距离融入 VRPTW 问题中解出最优配送方案并通过实际路线进行配送。所以在实际问题中不仅要得到合理的顾客配送顺序,还要得到用最少的机器人去服务以减少成本。本研究的成本函数公式为

$$C_t = 1\,000 \cdot N_v + 1 \cdot T_D \quad (25)$$

式中: N_v 代表本次送餐一共派出的机器人数目; T_D 代表所有机器人行走的距离。

首先要进行对餐厅环境的建模,这里通过栅格法,并且修改 c101.25 的一些顾客点位置信息后作为餐厅顾客点。在仿真餐厅环境下的算法步骤为:

1) 初始化顾客点信息、机器人信息以及混合蚁群算法试验参数,对餐厅环境进行建模。

2) 通过前文应用于路径规划的改进蚁群算法

找到顾客点之间和顾客点与送餐中心之间的最短距离,并记录下来其之间互相的实际路线。

3) 通过前文的混合蚁群算法,将其中最短距离替换为前文的顾客点之间和顾客点与送餐中心之间的最短距离,解出餐厅中顾客的最优配送方案。

4) 结合配送方案以及前文解出的实际路线,完成整个餐厅的机器人送餐配送调度。

一共 25 位顾客点的点位以及机器人送餐中心和餐厅的仿真环境如图 12 所示。

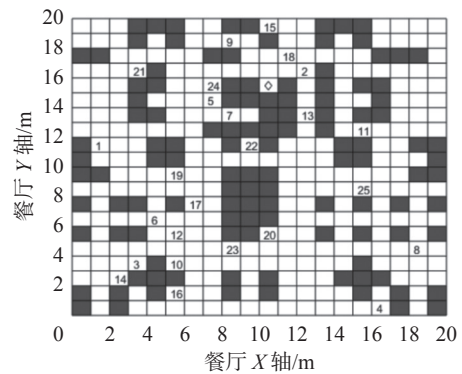


图 12 餐厅仿真环境

Fig. 12 Restaurant simulation environment

本研究选取 C101.25 的数据集,并根据实际生活情况修改其顾客坐标、顾客需求以及机器人的容量。设◇为餐饮中心,需求量单位为 0.1 kg,时间窗单位取 30 s,送餐机器人最大载重量为 20 kg。具体数据如表 2 所示。

表 2 餐厅信息表

Table 2 Restaurant information sheet

序号	X坐标/ 标/m	Y坐标/ 标/m	需求量/ kg	左时间窗/ s	右时间窗/ s	服务时间/ s
0	10.5	15.5	0	0	1236	0
1	1.5	11.5	1	912	967	90
2	12.5	16.5	3	825	870	90
3	3.5	3.5	1	65	146	90
4	16.5	0.5	1	727	782	90
5	7.5	14.5	1	15	67	90
6	4.5	6.5	2	621	702	90
7	8.5	13.5	2	170	225	90
8	18.5	4.5	2	255	324	90
9	8.5	18.5	1	534	605	90
10	5.5	3.5	1	357	410	90
11	15.5	12.5	1	448	505	90
12	5.5	5.5	2	652	721	90
13	12.5	13.5	3	30	92	90
14	2.5	2.5	1	567	620	90

续表 2

序号	X坐标/ 标/m	Y坐标/ 标/m	需求量/ kg	左时间窗/ s	右时间窗/ s	服务时间/ s
15	10.5	19.5	4	384	429	90
16	5.5	1.5	4	475	528	90
17	6.5	7.5	2	99	148	90
18	11.5	17.5	2	179	254	90
19	5.5	9.5	1	278	345	90
20	10.5	5.5	1	10	73	90
21	3.5	16.5	2	914	965	90
22	9.5	11.5	2	812	883	90
23	8.5	4.5	1	732	777	90
24	7.5	15.5	1	65	144	90
25	15.5	8.5	4	169	224	90

前面已经分别进行了带有障碍物的栅格环境下的路径规划算法验证对比和在 VRPTW 问题上各算法的配送方案算法的验证对比。在本节餐厅仿真环境下,进行 25 位顾客在餐厅用餐的送餐仿真算法对比。在同一仿真环境和顾客信息一致情况下,将基本蚁群算法和文献 [26] 算法结合后得到的送餐机器人最小距离迭代图与本研究算法实验得到的最小距离迭代图对比,如图 13 所示。两种算法的最小距离和最小成本对比如表 3 所示。

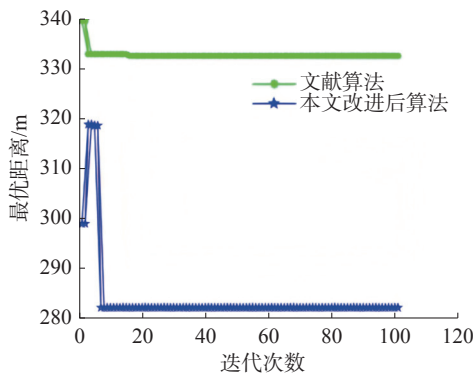


图 13 2 种算法最优距离对比

Fig. 13 Comparison diagram of optimal distance of two algorithms

表 3 2 种算法的最小距离和最小成本对比表

Table 3 Comparison table of minimum distance and minimum cost of two algorithms

文献算法最小 距离/m	本研究算法最 小距离/m	文献算法最小 成本	本研究算法最 小成本
332.7696	282.2498	4332.7696	3282.2498

由图 13 中 2 种算法的对比可以看出,由于本研究算法构建了满足约束条件的可选顾客集,因此只用了较少的迭代次数就寻到了最优解,并且由于本研究算法添加了适应度函数,在算法寻到了最优解时,收敛速度会加快,因此出现了最优

解迅速收敛的现象。在 25 位顾客的餐厅送餐问题下,本研究算法相比文献算法得到的机器人最小总距离结果更好,并且从 2 种算法的最小成本上来看,本研究算法只需要 3 个机器人便可以完成 25 位顾客的配送。而结合了基本蚁群算法的文献 [26] 算法需要 4 个机器人进行配送。这主要是因为基本蚁群算法在计算 25 位顾客之间的最优路径效果不如本研究在第 2 章改进的算法得到最优路径。并且文献算法在 25 位顾客之间的分配问题中,算法迭代次数在接近 20 代收敛,而本研究算法在 10 代以前收敛,在本研究算法的收敛性上也体现本研究算法的优越性。

通过本研究改进后算法得到 25 位最优配送方案如表 4 所示。

表 4 餐厅最优配送方案表

Table 4 Optimal distribution scheme of restaurant

机器人编号	配送的顾客点
1	0 20 17 25 8 15 9 6 23 22 1 0
2	0 5 24 7 19 10 16 14 12 2 21 0
3	0 13 3 18 11 4 0

机器人送餐配送仿真图如图 14-16 所示。图中◇为机器人的出发点,即餐饮配送中心。1-25 的数字分别代表该餐厅中,不同的顾客所在的位置信息。

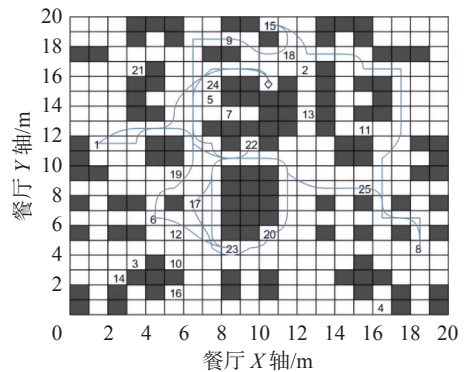


图 14 一号机器人的配送方案

Fig. 14 Distribution scheme of robot 1

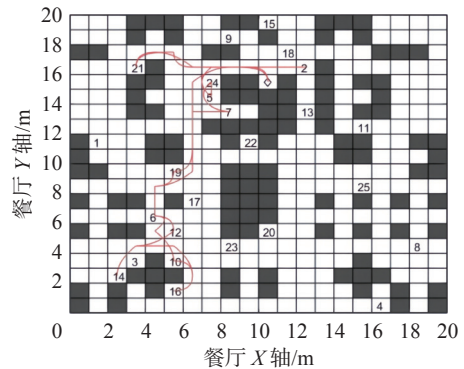


图 15 二号机器人的配送方案

Fig. 15 Distribution scheme of robot 2

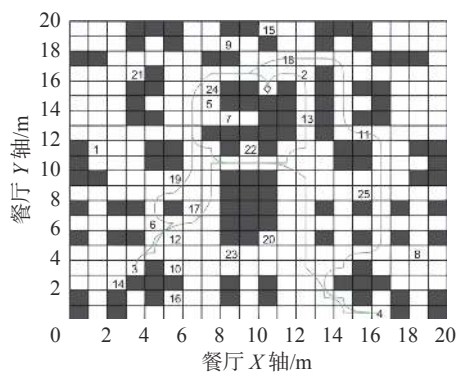


图 16 三号机器人的配送方案
Fig. 16 Distribution scheme of robot 3

6 结束语

本研究首先介绍了蚁群算法的原理和公式,将目标点对地图上每个点的距离因素以及引导素加入到蚁群算法的状态转移公式中,对比仿真验证了算法能够在栅格环境下得到良好的路径规划。接着提出了混合蚁群-遗传算法,通过加入顾客点的时间窗宽度和机器人等待时间因素到状态转移公式中,将蚁群算法得出的解作为遗传算法的初始种群,并加入了破坏和修复基因操作,得到了跟 Solomon Benchmark 的网站给出的历代最优解一致的结果,验证了改进算法的可行性。最后结合餐厅仿真环境通过仿真结果表明,改进后的算法能够在餐厅复杂环境下,规划出多位顾客合理配送方案以及最优路径规划路线。

参考文献:

- [1] 陶永,王田苗,刘辉,等.智能机器人研究现状及发展趋势的思考与建议[J].*高技术通讯*,2019,29(2):149-163.
TAO Yong, WANG Tianmiao, LIU Hui, et al. Insights and suggestions on the current situation and development trend of intelligent robots[J]. *Chinese high technology letters*, 2019, 29(2): 149-163.
- [2] KUANG Hua, XU Zhipeng, LI Xingli, et al. An extended car-following model accounting for the average headway effect in intelligent transportation system[J]. *Physica A: statistical mechanics and its applications*, 2017, 471: 778-787.
- [3] POOLE A, KOTSIALOS A. Swarm intelligence algorithms for macroscopic traffic flow model validation with automatic assignment of fundamental diagrams[J]. *Applied soft computing*, 2016, 38: 134-150.
- [4] CHMIEL W, DAŃDA J, DZIECH A, et al. INSIGMA: an intelligent transportation system for urban mobility enhancement[J]. *Multimedia tools and applications*, 2016, 75(17): 10529-10560.
- [5] 程志,张志安,李金芝,等.改进人工势场法的移动机器人路径规划[J].*计算机工程与应用*,2019,55(23):29-34.
CHENG Zhi, ZHANG Zhian, LI Jinzhi, et al. Mobile robots path planning based on improved artificial potential field[J]. *Computer engineering and applications*, 2019, 55(23): 29-34.
- [6] WANG Xiuhong, LIU Xuehao, WANG Yongcheng, et al. Research on path planning of mobile robot based on improved A* algorithm[C]//IE&EM 2019. Singapore: Springer, 2020: 153-161.
- [7] WANG Huijuan, YU Yuan, YUAN Quanbo. Application of Dijkstra algorithm in robot path-planning[C]//2011 Second International Conference on Mechanic Automation and Control Engineering. Hohhot. IEEE, 2011: 1067-1069.
- [8] ELHOSENY M, THARWAT A, HASSANIEN A E. Bezier curve based path planning in a dynamic field using modified genetic algorithm[J]. *Journal of computational science*, 2018, 25: 339-350.
- [9] SONG Baoye, WANG Zidong, SHENG Ligang. A new genetic algorithm approach to smooth path planning for mobile robots[J]. *Assembly automation*, 2016, 36: 138-145.
- [10] FERNANDES C M, MORA A M, MERELO J J, et al. KANTS: a stigmergic ant algorithm for cluster analysis and swarm art[J]. *IEEE transactions on cybernetics*, 2014, 44(6): 843-856.
- [11] 陈嘉林,魏国亮,田昕.改进粒子群算法的移动机器人平滑路径规划[J].*小型微型计算机系统*,2019,40(12):2550-2555.
CHEN Jialin, WEI Guoliang, TIAN Xin. Smooth path planning for mobile robots based on improved particle swarm optimization algorithm[J]. *Journal of Chinese computer systems*, 2019, 40(12): 2550-2555.
- [12] ALAM M S, RAFIQUE M U, KHAN M U. Mobile robot path planning in static environments using particle swarm optimization[EB/OL]. (2020-08-23)[2021-01-01]. <https://arxiv.org/abs/2008.10000.pdf>.
- [13] PATLE B K, PANDEY A, JAGADEESH A, et al. Path planning in uncertain environment by using firefly algorithm[J]. *Defence technology*, 2018, 14(6): 691-701.
- [14] JIANG Tao, WANG Jianzhong. Study on path planning method for mobile robot based on fruit fly optimization algorithm[J]. *Applied mechanics and materials*, 2014, 536/537: 970-973.
- [15] PANDA M, DAS B, PATI B B. Grey wolf optimization for global path planning of autonomous underwater vehicle[C]//Proceedings of the Third International Con-

- ference on Advanced Informatics for Computing Research. New York: ACM, 2019: 1–6.
- [16] 楼传炜, 葛泉波, 刘华平, 等. 无人机群目标搜索的主动感知方法 [J]. *智能系统学报*, 2021, 16(3): 575–583.
LOU Chuanwei, GE Quanbo, LIU Huaping, et al. Active perception method for UAV group target search[J]. *CAAI transactions on intelligent systems*, 2021, 16(3): 575–583.
- [17] 徐玉琼, 娄柯, 李志鲲. 基于变步长蚁群算法的移动机器人路径规划 [J]. *智能系统学报*, 2021, 16(2): 330–337.
XU Yuqiong, LOU Ke, LI Zhikun. Mobile robot path planning based on variable-step ant colony algorithm[J]. *CAAI transactions on intelligent systems*, 2021, 16(2): 330–337.
- [18] 夏小云, 周育人. 蚁群优化算法的理论研究进展 [J]. *智能系统学报*, 2016, 11(1): 27–36.
XIA Xiaoyun, ZHOU Yuren. Advances in theoretical research of ant colony optimization[J]. *CAAI transactions on intelligent systems*, 2016, 11(1): 27–36.
- [19] CAO Jingang. Robot global path planning based on an improved ant colony algorithm[J]. *Journal of computer and communications*, 2016, 4(2): 11–19.
- [20] TSAI P W, NGUYEN T T, DAO T K. Robot path planning optimization based on multiobjective grey wolf optimizer[C]// International Conference on Genetic and Evolutionary Computing. Cham: Springer, 2017: 166–173.
- [21] 张毅, 杨光辉, 花远红. 基于改进人工鱼群算法的机器人路径规划 [J]. *控制工程*, 2020, 27(7): 1157–1163.
ZHANG Yi, YANG Guanghui, HUA Yuanhong. Robot path planning based on improved artificial fish swarm algorithm[J]. *Control engineering of China*, 2020, 27(7): 1157–1163.
- [22] BRAND M, YU Xiaohua. Autonomous robot path optimization using firefly algorithm[C]//2013 International Conference on Machine Learning and Cybernetics. Tianjin: IEEE, 2014: 1028–1032.
- [23] DORIGO M, MANIEZZO V, COLORNI A. The ant system: an autocatalytic optimizing process Technical Report91-O16[R]. Milan: Dipartimento di Elettronica, Politecnico di Milano, 1991.
- [24] DORIGO M, MANIEZZO V, COLORNI A. Ant system: optimization by a colony of cooperating agents[C]//IEEE Transactions on Systems, Man, and Cybernetics, Part B. Cybernetics: IEEE, 2002: 29–41.
- [25] 董炫良, 赵桂清. 人工势场引导蚁群算法的机器人导航路径规划 [J]. *机械设计与制造*, 2021(6): 169–173.
DONG Xuanliang, ZHAO Guiqing. Robot navigation path planning based on ant colony algorithm guided by artificial potential field[J]. *Machinery design & manufacture*, 2021(6): 169–173.
- [26] 李琳, 刘士新, 唐加福. 改进的蚁群算法求解带时间窗的车辆路径问题 [J]. *控制与决策*, 2010, 25(9): 1379–1383.
LI Lin, LIU Shixin, TANG Jiafu. Improved ant colony algorithm for solving vehicle routing problem with time windows[J]. *Control and decision*, 2010, 25(9): 1379–1383.

作者简介:



蔡军, 教授, 主要研究方向为机器人技术、信号处理、模式识别。主编和参编教材 5 部, 发表学术论文 9 篇。
E-mail: 1073230317@qq.com。



钟志远, 硕士研究生, 主要研究方向为机器人路径规划。E-mail: 879202752@qq.com。