

基于改进蚁群算法的无人仓的多 AGV 避碰路径优化策略

杨 洋¹, 张建敏¹, 刘艺林², 宋 馨¹

(1. 中国矿业大学(北京)管理学院, 北京 100083)

(2. 石家庄邮电职业技术学院, 河北 石家庄 050021)

摘 要: 对无人仓库中多 AGV 系统的避碰路径优化问题进行了研究, 提出了一种基于弹性时间窗和改进蚁群算法的多 AGV 避碰路径优化策略. 通过对传统蚁群算法改进启发式信息和信息素更新策略, 来提高算法的执行速度和寻优能力, 提出 AGV 任务优先级排序并改进冲突解决策略来解决多 AGV 之间的不同路径冲突. 基于电商物流无人仓库的环境, 利用 MATLAB 仿真软件对多 AGV 避碰路径规划进行建模分析. 实验结果表明, 基于弹性时间窗和改进蚁群算法的可以实现多 AGV 避碰路径规划, 并能够短时间内找到避碰最优路径.

关键词: 多 AGV 系统; 无人仓; 蚁群算法; 弹性时间窗

1 引言

随着新零售时代的到来, 传统的物流体系正在发生快速变革, 大数据技术、人工智能技术、人工智能、仿真技术等物流行业的广泛应用, 使得无人仓已成为这些技术应用在物流体系的创新, 无人仓的目标就是指实现仓库作业流程的无人化操作, 包括入库、存储、分拣、出库等. 在国内, 无人仓技术起步早, 且发展速度快, 然而, 现在无论国内还是国外, 许多电商巨头都开始建立无人仓. 因此, 在未来发展中, 通过加快技术升级, 降低运行成本, 精准服务消费者来提高自身竞争力才能使无人仓技术应用得以致胜^[1]. 目前, 在无人仓技术设施中, 典型的搬运设备有输送线、自动导引车 (Automated Guided Vehicle, 简称 AGV)、穿梭车、类 Kiva 机器人、无人叉车等等, 智能搬运设备的应用大大提高了作业效率, 但在运行中, 如何快速找到最优路径并防止碰撞成为当前应用中值得研究的问题. 本文研究的是 AGV 在运行过程中寻找最优路径的过程.

AGV 是一种智能搬运设备, 装有电磁学或光学等自动引导装置, 并且能够沿着规定的路径行驶, 具有安全保护及移栽功能. 作为无人仓库运营的关键设备, 其工作效率直接影响无人仓库整体运营效率. 合理的 AGV 行驶路径不仅能提升无人仓库内产品流通效率与工作效率, 还能减少 AGV 派发数量并节约能耗, 增加其有效工作时间^[2], 然而, 随着搬运设备数量的不断扩张, AGV 在路径规划中不可避免地会出现碰撞, 系统死锁等问题, 严重影响路径优化, 降

收稿日期: 2019-07-11

资助项目: 中国矿业大学(北京) 越崎青年学者资助; 中国物流学会、中国物流与采购联合会研究课题面上项目资助 (2019CSLKT3-117); 中央高校基础业务经费 (2014QG01); 研究生科研创新能力提升项目 (2020YJSGL06)

低了系统柔性和使用效率,这是目前 AGV 研究领域的一大热点和难点. AGV 的避碰问题主要分为两类,一类是对静态障碍物的躲避,另一类是将系统中其他 AGV 视为动态障碍物,避免碰撞^[3]. 因此,AGV 避碰路径优化的任务内容即在具有静态、动态障碍物的环境中,能够按照用时最少、路程最短的要求,寻找一条从出发点到目的地的无碰路线.

传统的最优路径规划方式大多基于 A* 算法^[4], Dijkstra 算法^[5]等,但是都有着计算量大,收敛速度慢,不具备实时性的弊端. 近年来,神经网络、蚁群算法和遗传算法也陆续应用到路径规划中^[6-10]. 与上述算法相比,蚁群算法在移动机器人的路径规划应用过程中,表现出良好的搜索能力、正反馈特性、分布式计算等,具有很强的启发性和稳定性,但传统的蚁群算法在应用过程中仍会存在收敛速度慢、易陷入局部最优解、停滞等缺陷. 另外,在解决 AGV 路径规划上,由于蚁群算法具有易与其他算法融合、速度快、精度高、能够快速找到拟最优解等优点,因此,当前现有研究中对蚁群算法提出了多种改进方案来适应 AGV 的路径规划问题. 比如:李军军等将蚁群算法与粒子群算法相结合,以行驶时间为考量解决多 AGV 路径冲突问题^[11];卜新苹等基于蚁群算法与 Bezier 曲线相结合的方法得到曲率连续的路径,对复杂环境下的大型工装的物料运输与维修的路径规划进行研究^[12];夏扬坤等提出自适应性的禁忌搜索算法,研究了订单拆分多 AGV 物料配送路径规划,考虑了距离、载重、需求等因素,使路径规划更加符合实际需求^[13];胡庆朋等基于传统蚁群算法通过正态分布改变其转移概率,提高了收敛速率,避免了陷入局部最优解的问题^[14]. 何成伟等使用蚁群算法结合 Dijkstra 算法规划 AGV 初始路径,引入节点随机选择机制得到最优路径,或是通过更改启发函数,算法中引入全局信息,得到全局的最优解^[5]. 总之,已有研究并没有很好地解决多 AGV 的避碰路径优化问题.

因此,本文基于传统蚁群算法,结合弹性时间窗原理^[3,15]进行算法改进,对系统中的 AGV 进行任务优先级排序,进而达到多 AGV 系统在物流拣选工作时的避碰路径优化的目标;其次,根据物流无人仓的实际作业情景,进行模拟假设,并使用 MATLAB 软件进行仿真实验,证明改进方法可在保证系统无死锁无碰撞的基础上,实现 AGV 路径规划的优化.

2 问题描述

2.1 无人仓环境描述

电商物流无人仓库是目前发展较为迅速的仓储模式,物流 AGV 在其中承担着重要的“货到人”搬运任务,大大提升了订单拣选的效率. 一般而言,电商无人仓库中包括货物存储区、拣选台、AGV 停放/充电区三大核心区域. AGV 在收到订单拣选命令之后,由停放地点出发,行至目标货架,将其货架从货物存储区送往拣选台区域,随后将空货架送回原位,回归原位之后,拣选系统即能下发下一订单命令. 为提升拣选效率,系统需要同时操控数台 AGV,导致 AGV 相撞冲突率激增. 本文的路径规划就是多 AGV 在收到订单拣选命令之后,要在可行路径中,寻找出“发点-目标货架-拣选台”之间的最短路径,并在该过程中,不仅要躲避货架、拣选台、充电设备等静态障碍物,关键是防止运行中的多台 AGV 之间出现碰撞.

基于此,本文选用栅格法进行环境简化建模. 将实际作业环境用二维平面图形表示,称该平面图为栅格地图,地图中包含若干个单位大小的栅格,将所有栅格分为障碍栅格(不可通行)和可行栅格(可自由通行),分别用黑色和白色表示,如图 1 所示. 在图一中,栅格

4,5,7,8,10,11,18,19,21,22,24,25 代表拣选台, 不可通行. 栅格 44,45,48,59,72,73,47,48,61,62,75, 76,50,51,64,65,78,79,53,54,67,68,81,82,100,101,114,115,128,129,103,104,117,118,131,132, 106, 107, 120, 121,134,135,109,110, 123,124,137,138 代表货物存储区, 不可通行. 栅格 173,174, 187,188, 176,177, 190, 191,179,180,193,194 代表 AGV 停放/充电区域, 不可通行. 其余白色栅格代表 AGV 作业的自由可行空间. 同时, 在此栅格环境中, 假设每台 AGV 满足以下特征:

1	2	3	4	5	6	7	8	9	10	11	12	13	14
15	16	17	18	19	20	21	22	23	24	25	26	27	28
29	30	31	32	33	34	35	36	37	38	39	40	41	42
43	44	45	46	47	48	49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80	81	82	83	84
85	86	87	88	89	90	91	92	93	94	95	96	97	98
99	100	101	102	103	104	105	106	107	108	109	110	111	112
113	114	115	116	117	118	119	120	121	122	123	124	125	126
127	128	129	130	131	132	133	134	135	136	137	138	139	140
141	142	143	144	145	146	147	148	149	150	151	152	153	154
155	156	157	158	159	160	161	162	163	164	165	166	167	168
169	170	171	172	173	174	175	176	177	178	179	180	181	182
183	184	185	186	187	188	189	190	191	192	193	194	195	196
197													

图 1 栅格地图

- 1) 每台 AGV 占据一个栅格, 单位移动速度等于栅格边长;
- 2) 每台 AGV 速度相同, 且空载与载物速度一致, 转弯时耗为常数;
- 3) 单位可行栅格的大小要确保 AGV 能够顺利通行, 避免因栅格大小的因素对 AGV 的路径规划造成不必要的影响;
- 4) 每台 AGV 同一时刻只能接受一个任务, 完成后方能执行下一任务;
- 5) 两台 AGV 之间规定最小安全距离, 设置为一个车身长度.

2.2 多 AGV 避障路径问题描述

在实际的无人仓货物运输过程中, 货物订单较多, 就会需要多台 AGV 共同运行才能更高效地完成订单拣选任务, 对于多 AGV 动态系统, 物流运输过程中情况相对复杂, 并且随着订单的增多, AGV 的不断加入, 路径规划过程也变得更加复杂, 因此, 由于系统原因, 调度不合理或有意外情况发生时, 将很可能会导致 AGV 发生碰撞、路径冲突、死锁等现象发生. 为解决 AGV 之间因碰撞而导致死锁问题, 规定 AGV 之间的最小安全距离; 路径冲突是指在系统中, 当某一 AGV 路径规划已完成时, 其走过节点会有相应的时间窗, 当另一 AGV 的规划路径有在某一时段的相同节点, 称此节点为路径冲突节点. 路径冲突情况复杂, 包括静止冲突、交叉冲突、相向冲突、追击冲突^[16], 因为本文假设 AGV 速度恒定, 且两车之间保持最小安全距离, 然后主要研究如何避免动态障碍物, 所以主要强调交叉冲突和相向冲突两种. 对于该两种冲突 (如图 2) 的具体描述如下:

1) 交叉冲突

如图 2(a) 所示, 由于每台 AGV 运行速度一致且平稳, 每个栅格大小相等, 因此 AGV1 与 AGV2 将要同时到达图中的阴影栅格中, 即 AGV 将要在阴影栅格中发生路径冲突. 这种现象称为多 AGV 系统路径规划中经常遇到的交叉冲突问题.

2) 相向冲突

如图 2(b) 所示, AGV1 与 AGV2 相向行驶, 由于一台 AGV 占据一个栅格, 此路径只允许一台 AGV 通过, 即不能同时供两台 AGV 通过, 因此, 两 AGV 若要继续前行到达阴影部分, 则会发生路径冲突. 这种现象称为多 AGV 系统路径规划中经常遇到的相向冲突问题.

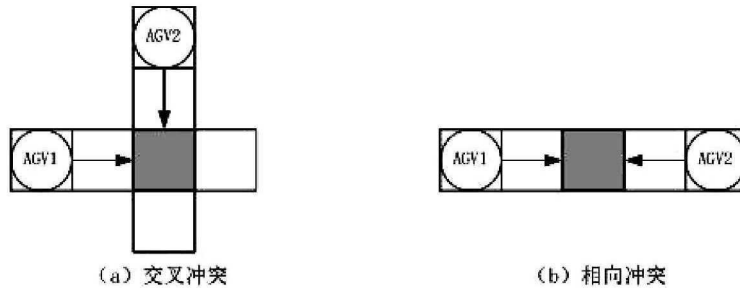


图 2 路径冲突示意图

3 模型建立

3.1 基于蚁群算法的基本模型

蚁群算法是来源于自然界的一种智能随机搜索算法, 有很强的发现较好解的能力, 其原理为蚂蚁在觅食过程中能够在所经过的路径中释放信息素进行信息传递, 信息素浓度则代表此路径的距离长度, 浓度与距离成反比, 浓度高的代表距离短. 其他蚂蚁通过判断各路径上的信息素浓度和启发信息决定下一步的转移方向, 逐步觅得食物. 蚁群算法鲁棒性强, 设置简单, 易于实现, 因而已被广泛应用到路径规划问题的求解.

假设蚂蚁个数为 M , 在 G 次迭代中找到最优路径. 设在 t 时刻蚂蚁 k 位于栅格 i , $P_{ij}^k(t)$ 是 t 时刻蚂蚁由栅格 i 转移到栅格 j 的转移概率.

$$P_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [n_{ij}(t)]^\beta}{\sum_{s \in j_k(i)} [\tau_{is}(t)]^\alpha \cdot [n_{is}(t)]^\beta}, & j \in j_k(i) \\ 0, & else \end{cases} \quad (1)$$

其中: k 表示第 k 只蚂蚁: $k = 1, 2, 3, \dots, M$;

i, j 为栅格地图中的具体坐标;

$\tau_{ij}(t)$ 表示 t 时刻 i 栅格到 j 栅格的信息素, 信息素初始值一般设为一较小常数;

$n_{ij}(t)$ 是栅格 i 到栅格 j 的启发信息, $n_{ij}(t) = 1/d_{ij}$, d_{ij} 表示栅格 i 到栅格 j 的距离;

$j_k(i)$ 为蚂蚁 k 在栅格 i 的可行域;

α, β 分别表示信息素和启发信息的重要性程度;

禁忌表 $Tabu_k$ 存入蚂蚁 k 已经路过的栅格. 若蚂蚁陷入死胡同无后续栅格可选, 则默认该蚂蚁已经死亡, 算法删除该蚂蚁及其所寻路径.

每次迭代中当所有蚂蚁寻径完成后, 对所有路径上的信息素进行更新.

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

$$\Delta\tau_{ij}(t) = \sum_{k \in K_{ij}} \Delta\tau_{ij}^k(t) \quad (3)$$

$$\Delta\tau_{ij}^k(t) = Q/L_k \quad (4)$$

其中, ρ 是信息素挥发系数, $\Delta\tau_{ij}(t)$ 是本次循环节点 ij 间信息素增量, K_{ij} 为经过节点 ij 的

所有蚂蚁, $\Delta\tau_{ij}^k(t)$ 为第 k 只蚂蚁所留下的信息素量, L_k 是蚂蚁 k 所寻路径的长度, Q 为一正常数. 在全局信息素更新完成后, 把所有蚂蚁放回至起点重新寻径, 经过 G 次迭代完成后, 计算出最短路径.

3.2 基于弹性时间窗的改进模型

在蚁群算法的基本模型中, 蚂蚁在搜索路径时是以所处栅格与相邻栅格直线距离的倒数为启发式信息, 直线距离作为依据, 不能好的反映 AGV 在实际路径中的运行时间; 两点之间路径搜索盲目性较大, 未知因素较多; 信息素在更新时是依据第 k 只蚂蚁所留下的信息素量, 信息素量相对较大, 搜索速度慢. 因此, 为了缩短 AGV 收到任务后进行路径规划到完成任务的时间, 针对传统蚁群算法中的这些不足, 本文对算法部分作出的改进如下:

①为了提高算法搜索的精确性, 提高蚂蚁对终点的可见度, 本文以蚂蚁所处节点与目标点的曼哈顿距离的倒数为启发式信息. 如式 (5) 所示.

$$\eta(P) = \frac{1}{|(E(x) - P(x))| + |(E(y) - P(y))|} \quad (5)$$

注: E 代表终点栅格, P 代表蚂蚁所处栅格.

②为了减少不良路径信息素的干扰, 同时为提高蚂蚁搜索的速度, 本文参考陈国良等专家通过设置蚁群信息素更新机制来避免传统蚁群算法收敛性差、局部最优缺点的思路^[17], $\Delta\tau_{ij}^k(t)$ 改进为选取第 k 只蚂蚁所留下的信息素中的最短路径作为信息素更新的依据. 如式 (6) 所示.

$$\begin{aligned} \Delta\tau_{ij}^k(t) &= Q/L_k \\ L_k &= \min L \end{aligned} \quad (6)$$

在传统系统中, 在 AGV 即将发生碰撞时, 系统没有依据来决定 AGV 的先后通过顺序, 那么只能面临碰撞, 出现系统错误, AGV 将无法再继续运行. 因此, 为了 AGV 在路径规划过程中解决避碰问题, 本文对多 AGV 系统作出的改进: 对系统中的每台 AGV 进行任务优先级排序. 可以根据命令下达的时间顺序, 即 AGV 所承担订单任务的时间先后进行排序, 时间在前的任务优先级高, 时间在后的任务优先级低, 即将发生碰撞时, 任务优先级高的优先通过, 任务优先级低的相应采取等待或绕行.

基于 AGV 任务优先等级排序后, 可有效解决路径冲突问题, 面对不同类型的冲突, AGV 可采取的策略如下:

①当在某一节点发生路径交叉冲突的时候, 解决的方法可以通过等待策略解决, 任务优先级低的 AGV 停车等待, 优先级高的 AGV 先行通过.

②当在某一节点发生路径相向冲突的时候, 等待策略无法解决冲突, 则采取的策略是对任务优先级低的 AGV 重新进行路径搜索, 寻找避碰前提下的最优路径.

3.3 改进蚁群算法的实施步骤

1) 环境建模. 建立仓储环境栅格模型, 根据任务优先级顺序依次给 AGV 分配任务, 确定已分配任务 AGV 的目标节点, 对承载任务优先级最高的 AGV 首先进行路径搜索, 初始化每个节点的时间窗表和其它信息参数.

2) 路径初步规划. 设置迭代计数器 $G=0$, 将 M 只蚂蚁放在起始位置上, 并把初始点加入到禁忌表中, 根据转移概率公式计算出蚂蚁 k 所有可转移节点的概率, 根据轮盘赌规则选择

下一节点,更新禁忌表,存储所有蚂蚁的所寻路径,并保存路径长度,删除死亡蚂蚁,默认其路径长度为无穷大;根据信息素更新公式进行全局信息素更新,并清空禁忌表;判断 G 是否等于最大迭代次数,若是,流程结束,搜索出该 AGV 的最优路径;若否,重复进行上述步骤.

3) 计算时间窗. 计算该 AGV 经过节点的时间窗,更新每个节点的时间窗表. 按顺序搜索下一辆 AGV 的最优路径,并计算出该 AGV 所经过节点的时间窗,更新所有节点的时间窗表,比较每个节点是否发生时间窗冲突,如果没有冲突,则进行下一辆 AGV 的路径规划,如果有冲突,则将冲突节点加入禁忌表,则解决时间窗冲突,并进行再次比较.

4) 重新搜索. 如果在进行路径规划的过程中发生与相邻节点在时间窗上均发生冲突而导致任务无法执行的情况,就通过扩大时间节点的搜索范围,重复进行新一轮的搜索.

4 仿真实验及结果分析

表 1 仿真参数说明

参数	取值
栅格地图大小	14*14
迭代次数 G	100
蚂蚁数量 K	80
初始信息素 α	1
启发信息的重要程度 β	7
信息素蒸发次数 p	0.3
常数 Q	1
AGV 数量	2

为了验证提出算法改进的有效性,本文运用 MATLAB 进行了仿真模拟实验. 假设现在有两辆不同优先级的 AGV1 与 AGV2,起点栅格分别为 33 和 37,目标栅格都为 147. 其中假设 AGV1 的任务级优先于 AGV2. AGV1 与 AGV2 同时向目标栅格前进. 基于表 1 初始化的参数设置,在此模拟环境下进行 AGV 避碰实验.

基于蚁群算法的基本模型进行路径搜索后,AGV1 的路径规划节点为 33,34,49,63,77, 91,105, 119,133,147;AGV2 的路径规划节点为 37,36,49,63,77, 91,105,119,133,147. AGV1 和 AGV2 在执行任务过程中,二者在 49 栅格处发生碰撞,系统出现报错,无法继续前行,最终导致 AGV1 与 AGV2 无法完成搬运任务,因此完成任务时间为空,记为“—”,实验结果如表 2 所示. 其相应路径冲突二维平面图、三维图分别如图 3(a) 和图 3(b) 所示.

表 2 传统蚁群算法任务路线

路径	完成时间
AGV1: 33,34,49,63,77,91,105,119,133,147	—
AGV2: 37,36,49,63,77,91,105,119,133,147	—

注:完成任务时间“—”表示为“空”. 原因:AGV1 与 AGV2 同时到达栅格 49 处,发生碰撞,不再执行搬运任务,任务未完成.

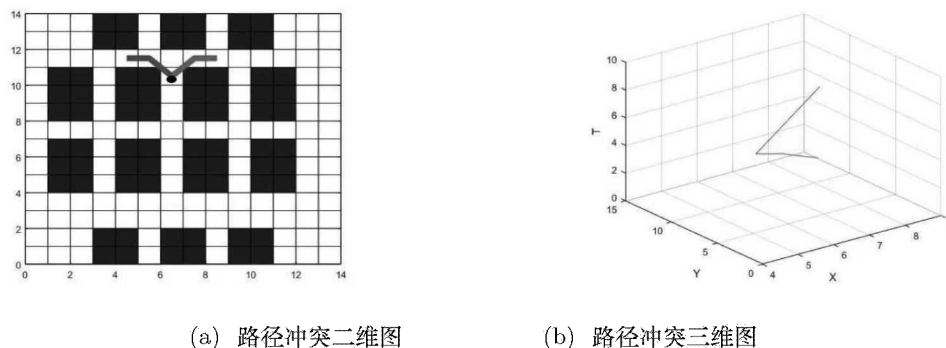


图 3 路径冲突示意图

基于改进蚁群算法, 其中 AGV1 任务优先级高, 其路径不需做出改变, 可继续执行任务, 此时 AGV2 任务优先级低, 在感知即将会与 AGV1 出现路径冲突时, 有两种策略可供选择. 因此, 进行了两种应对策略实验对比.

策略 1: AGV2 选择重新寻路策略, 此时 AGV2 基于改进蚁群算法, 重新选择其他最短路径, 经系统搜索后, 路径为: 37, 52, 66, 80, 94, 108, 122, 136, 149, 148, 147, 耗时 410s, MATLAB 实验结果如图 4(a)(二维)、图 4(b)(三维) 所示;

策略 2: AGV2 选择等待策略, 此时, AGV2 要在 36 栅格处等待, 继 AGV1 通过后再继续前行, 此时 AGV2 的路径: 37, 36, 36, 49, 77, 91, 105, 119, 133, 147, 耗时 340s, MATLAB 实验结果如图 4(c)(二维)、图 4(d)(三维) 所示. AGV1 与 AGV2 在两种策略下的路径及任务完成用时见表 3.

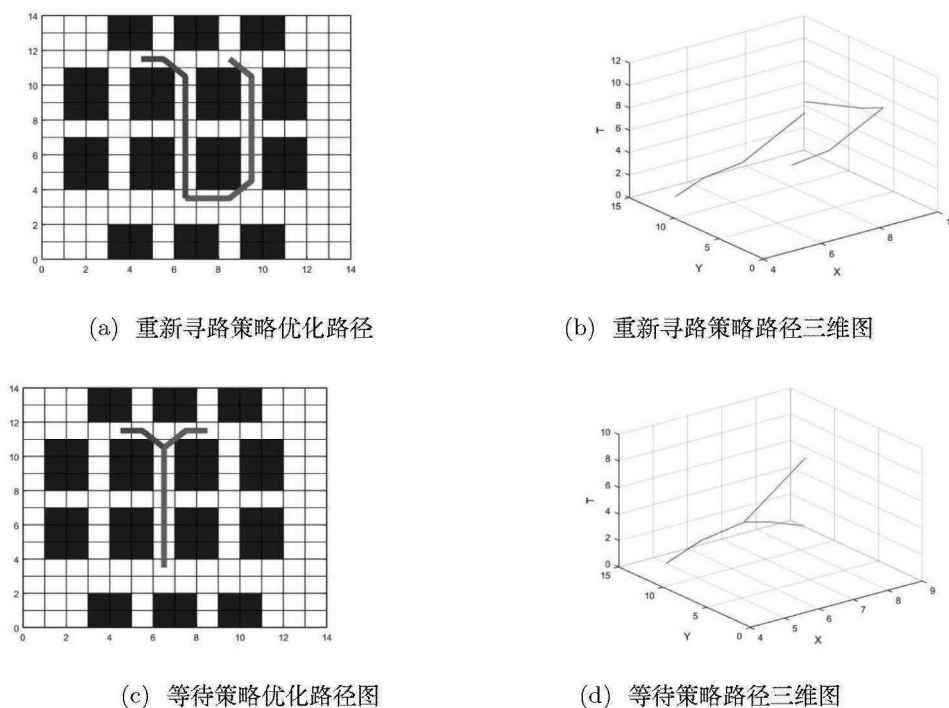


图 4 优化路径示意图

表 3 改进蚁群算法任务路线

策略	路径	完成任务用时
重新寻路策略	AGV1: 33,34,49,77,91,105,119,133,147	410s
	AGV2: 37,52,66,80,94,108,122,136,149,148,147	
等待策略	AGV1: 33,34,49,77,91,105,119,133,147	340s
	AGV2: 37,36,36,49,77,91,105,119,133,147	

经仿真实验对比发现, 传统蚁群算法中多 AGV 出现路径冲突, 将无法继续执行订单任务, 会造成系统锁死等现象; 经改进蚁群算法后, 搜索范围的减少, 可以在更短的时间内找到最优路径, 且在出现多 AGV 路径冲突时, 完全有效地避免了 AGV 之间的碰撞问题, 且能够顺利完成订单任务, 避免了系统锁死现象的发生, 并可以有两种不同的应对策略可供选择, 包括等待策略和重新寻路策略, 在此实验中, 实验结果表明等待策略可以在避免 AGV 碰撞的前提下更高效地完成订单任务。

5 结论

本文的主要贡献: 首先, 对传统蚁群算法做出了系列改进: ①将启发式信息调整为所处节点与目标点的曼哈顿距离的倒数. ②将信息素更新的依据调整为最短路径. ③对 AGV 进行任务优先级的排序.

其次, 根据无人仓内实际作业情况进行模拟环境假设, 并基于改进蚁群算法的多 AGV 避碰路径优化算法利用 MATLAB 进行仿真模拟, 来观察无人仓内多 AGV 的运行状况, 实验结果显示, 改进蚁群算法可以实现缩短路径规划时间以及有效避免路径冲突的目标. 仿真实验验证了该方法的可靠性和有效性, 无论是在路径规划时间还是避碰问题上改进蚁群算法都优于传统蚁群算法.

随着算法的不断改进, 在未来的研究中还可考虑当运输设备发生路径冲突时, 系统可自动识别出不同应对策略间的用时时长并直接寻找到最优路径, 将使其更加符合无人仓的实际作业情况.

参考文献

- [1] <http://it.people.com.cn/n1/2018/0530/c1009-30022080.html>
- [2] 郭兴海, 计明军, 刘双福. 融合多目标与能耗控制的无人仓库内 AGV 路径规划 [J/OL]. 计算机集成制造系统: 1-15[2019-03-31]. <http://kns.cnki.net/kcms/detail/11.5946.tp.20190315.0932.012.html>.
- [3] 谷宝慧. 基于时间窗的自动导引车系统路径优化研究 [D]. 兰州交通大学, 2015.
- [4] 泰应鹏, 邢科新, 林叶贵, 张文安. 多 AGV 路径规划方法研究 [J]. 计算机科学, 2017, 44(S2): 84-87.
- [5] 何成伟, 茅健. 基于改进蚁群算法的 AGV 路径研究 [J]. 物流科技, 2019, 42(3): 60-65.
- [6] 孟冲, 任彧. 基于多种群遗传算法的多 AGV 调度 [J]. 电子科技, 2018, 31(11): 47-50+68.
- [7] 袁瑞萍, 王慧玲, 孙利瑞, 李俊韬. 基于物流 AGV 的“货到人”订单拣选系统任务调度研究 [J]. 运筹与管理, 2018, 27(10): 133-138.
- [8] 唐世文, 洪洲. 基于蚁群算法的机器人路径规划研究 [J]. 机电信息, 2019(8): 46-47.
- [9] 赵艳东, 张申申. 基于改进蚁群算法的智能交通路径规划 [J]. 工业仪表与自动化装置, 2019(2): 30-32.

- [10] 卫玉梁, 靳伍银. 基于神经网络 Q-learning 算法的智能车路径规划 [J]. 火力与指挥控制, 2019, 44(2): 46-49.
- [11] 李军军, 许波桅, 杨勇生, 吴华锋. 多自动导引车路径规划的诱导蚁群粒子群算法 [J]. 计算机集成制造系统, 2017, 23(12): 2758-2767.
- [12] 卜新苹, 苏虎, 邹伟, 王鹏, 周海. 基于非均匀环境建模与三阶 Bezier 曲线的平滑路径规划 [J]. 自动化学报, 2017, 43(5): 710-724.
- [13] 夏扬坤, 符卓, 谢九勇. 依订单拆分的多自动导引车物料配送路径规划 [J]. 计算机集成制造系统, 2017, 23(7): 1520-1528.
- [14] 胡庆朋, 王涛, 张如伟. 改进蚁群算法在 AGV 全局路径规划中的研究 [J]. 信息技术与信息化, 2019(3): 116-118.
- [15] FISHER M L, RNSTEN J, et al. Vehicle Routing with Time Windows[J]. Two Optimization Algorithms Operations Research, 1997, 45(3): 488-492.
- [16] 王淑青, 毛月祥, 袁晓辉. 有限状态机的多 AGV 路径优化策略 [J]. 华侨大学学报 (自然科学版), 2019, 40(2): 239-244.
- [17] 陈国良, 刘杰, 张钊钊. Ant Colony Optimization with Potential Field Based on Grid Map for Mobile Robot Path Planning[J]. Journal of Donghua University(English Edition), 2016, 33(5): 764-767.

Multi-AGV Collision Avoidance Path Optimization Strategy for Unmanned Warehouse Based on Improved Ant Colony Algorithm

YANG Yang¹, ZHANG Jian-min¹, LIU Yi-lin², SONG Xin¹

(1. School of Management of China University of mining and technology, Beijing 100083, China)

(2. ShiJiaZhuang Posts and Telecommunication Technical College, Shijiazhuang 050021, China)

Abstract: In this paper, the problem of collision avoidance path optimization for multi-AGV systems in unmanned warehouses is studied. A multi-AGV collision avoidance path optimization strategy based on elastic time window and improved ant colony algorithm is proposed. In this paper, the traditional ant colony algorithm is improved by heuristic information and pheromone update strategy to improve the execution speed and optimization ability of the algorithm. The priority scheduling of AGV tasks and the improvement of conflict resolution strategies are proposed to solve the different path conflicts between multiple AGVs. Based on the environment of the e-commerce logistics unmanned warehouse, the MATLAB simulation software is used to model and analyze the multi-AGV collision avoidance path planning. The experimental results show that the multi-AGV collision avoidance path planning can be realized based on the elastic time window and the improved ant colony algorithm, and the optimal collision avoidance path can be found in a short time.

Keywords: Multiple AGV system; unmanned warehouse; ant colony algorithm; elastic time window