

一种基于改进蚁群算法的AGV小车三维路径规划研究

金鑫鑫

(亳州学院电子与信息工程系 安徽 亳州 236800)

摘要 针对智能仓储系统AGV小车拣货作业效率低的问题,提出一种基于改进蚁群算法的AGV小车三维路径规划方法。根据仓库原型构建立体仓库三维模型,并确定多目标三维坐标。根据两目标点的三种不同位置状态,构建AGV小车运行轨迹模型。通过增加禁忌表对基本蚁群算法进行改进,以解决三维折线路径优化问题。对比实验结果表明,改进后的蚁群算法可以避免陷入局部最优,且能够有效提高AGV小车拣货作业效率。

关键词 AGV小车 蚁群算法 智能仓储系统 三维路径规划 运行轨迹

中图分类号 O224 TP3

文献标志码 A

DOI:10.3969/j.issn.1000-386x.2022.07.042

RESEARCH ON 3D PATH PLANNING OF AGV CAR BASED ON IMPROVED ANT COLONY ALGORITHM

Jin Xinxin

(Department of Electronics and Information Engineering, Bozhou University, Bozhou 236800, Anhui, China)

Abstract Aimed at the problem of low picking efficiency of the intelligent storage system AGV trolley, a three-dimensional path planning method for the AGV trolley based on an improved ant colony algorithm is proposed. A three-dimensional warehouse model was constructed based on the warehouse prototype, and multi-target three-dimensional coordinates were determined. According to the three different position states of the two target points, an AGV trolley trajectory model was constructed. The basic ant colony algorithm was improved by adding tabu lists to solve the problem of 3D polyline path optimization. Comparative experimental results show that the improved ant colony algorithm can avoid falling into a local optimum, and can effectively improve the efficiency of AGV trolley picking operations.

Keywords AGV trolley Ant colony algorithm Intelligent storage system 3D path planning Running trajectory

0 引言

智能仓储系统是指以信息化手段实现对仓库的实时监视和全面管理的信息系统,该系统能够实时掌握仓库容量、了解货物所存放位置、实现自主货物转移等^[1-2]。在智能仓储系统中,货物转移快慢直接关系到其管理效率,通常执行货物转移作业的设备主要为AGV小车,如果要提高货物转移的速度,就需要对AGV小车的行驶路径进行规划^[3]。

当前针对仓储系统中AGV小车的路径优化算法

包括人工势场法、遗传算法、蚁群算法等。其中遗传算法虽然全局搜索能力强,但存在产生无效路径的问题;人工势场法在势场合力为零时会出现停滞状态,容易陷入局部最优;相比之下,蚁群算法具有正反馈、高稳健性和并行性的优点,已成功地用于智能仓储AGV小车的行驶路径规划等问题^[4]。

蚁群算法最早是由Dorigo等学者提出的分布式智能仿生算法,该算法模拟了蚂蚁合作觅食行为的性质^[5]。从算法原理上来看,蚁群算法主要存在搜索空间大、局部最优、搜索效率低、计算量大等问题。近年来,国内外众多专家学者致力于利用蚁群算法解决全

局路径规划问题。文献[6]通过重新定义蚁群算法中的转移概率,对蚁群算法进行改进,从而加快了算法的寻优速度,但是其转移概率自适应调整能力较差。文献[7]为了避免蚁群在面对凹型障碍物时易陷入死锁状态的问题,通过采用广义信息素更新原则,代替传统蚁群算法的信息素更新,从而找到最优解,但是该算法易陷入局部最优。文献[8]提出了双层蚁群算法,大大提高了路径规划的效率,但双蚁群的信息素作用会相互干扰,影响寻优结果。文献[9]提出一种基于自适应调整信息素的改进蚁群算法,根据人工蚂蚁所获得解的情况,动态调整路径上的信息素,从而使得算法跳离局部最优解,但是该方法未能考虑较优路径重要路段上的信息素强度,使得算法搜索效率不高。

智能仓储系统中的 AGV 小车路径规划是当前研究的热点和难点。针对 AGV 路径规划,文献[10]在路径优化的方法中增加了对时间和能耗的考虑,实现了仓储系统货物转移作业中的能耗和效率之间的有效平衡。文献[11]通过利用交通规则和预约表解决仓储物流机器人集群在运行时发生的碰撞和死锁问题,并根据所设定的协同机制,减少机器人无任务的待机状态,平衡各机器人之间的工作量,最终实现在保证系统安全运行的基础上缩短系统运行时间的目的。文献[12]在仓储系统货物转移作业的工作中,为平衡行驶距离、行驶时间、作业载重等因素,建立了多因子的路径优化模型,并通过集成学习优化算法降低模型的复杂度,实现了仓储系统货物转移作业时间效率的优化。

从当前的研究现状来看,AGV 路径规划的研究主要针对二维平面避障、二维路径优化等,但是不适用于货架式仓储系统中的三维折线路径规划^[10],且大多数路径规划算法运行效率低,容易陷入局部最优。鉴于此,本文提出了一种基于改进蚁群算法的 AGV 小车三维路径规划方法。首先,通过构建三维仓库模型来确定不同目标的三维坐标,并根据多个位置坐标建立 AGV 小车轨迹模型;其次,针对仓库 AGV 小车三维折线路径,利用改进的蚁群算法来实现路径优化问题;最后通过对比实验来验证基于改进后的蚁群算法的性能。

1 三维空间路径优化问题建模

1.1 立体仓库模型构建

在智能仓储系统中,AGV 小车的作业模式主要有入库和出库作业^[13]。在智能仓储系统中,单一的 AGV

小车作业往往需要在多个目标之间进行遍历工作且货物所存货架高度不一,因此需要对 AGV 小车的作业路径进行三维空间规划。在路径规划过程中不仅需要考虑到二维平面内的最优路径,还需要注意小车在取货物时的垂直运动及变换不同货道时 AGV 小车的运动。鉴于此,本文建立立体仓库模型,确定多目标的三维坐标,并通过模拟现实货架和 AGV 小车来进行实验验证和操作。图 1 所示为立体仓库货架实物图,图 2 为立体仓库模型。模型将每排货架进行网格化划分,其坐标表示为 (y,z) ,货架的排数表示为坐标 x 。通过这种方式可以将整个货架式仓储系统表示为三维网格地图,这样货架的每一格均以三维坐标的形式表示出来,从而为 AGV 小车的路径规划提供保障。



图 1 立体仓库货架实物图

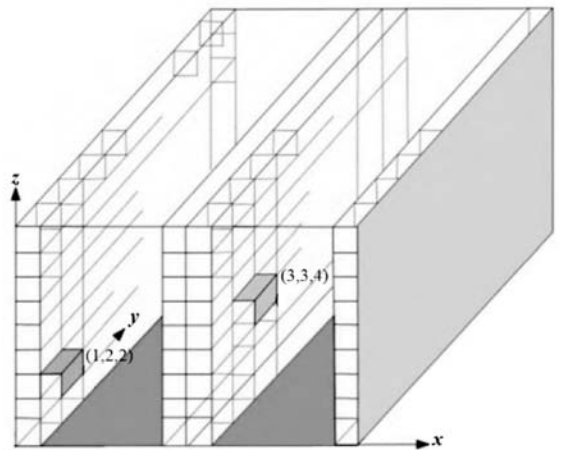


图 2 立体仓库模型

本文研究所使用的 AGV 小车并非单一两点式作业小车,而是可以在多个目标点之间遍历作业。智能仓储系统为货架式仓库,每个货架之间为货道,AGV 小车在同一个货道上可以进行上下、前后运动,如果要到另一个货架上取货物,则需要先退出该货道,再进入到另一个货道。图 3 所示为实验 AGV 小车。



图3 实验AGV小车

当建立立体仓库模型,并确定不同目标的三维空间坐标后,AGV小车入库、出库作业就可以按照以下三个步骤进行:(1)AGV小车计算所有目标点与自身之间的距离,并选取其中最近的目标点作为本次作业的起始点,并运动至起始点。(2)AGV小车根据所有目标点的位置计算出本次作业的顺序,并按顺序依次到达下一目标点进行存取货。(3)AGV小车完成所有目标点的任务后回到起始点,即完成了本次作业,并等待下一次的作业指令。

通过上述分析可知,多目标点之间的顺序决定了AGV小车路径规划的效率,同时也直接影响了AGV小车的单次作业时间。

1.2 运行轨迹模型构建

在立体仓库模型中,AGV小车的运动主要包括水平和垂直两个方向,其中水平方向主要由小车驱动系统自主运行,垂直方向运动主要借助升降机来实现。假设小车为匀速运行,考虑其在启动及停止时的速度影响,取速度和时间均值,因此不会对路径优化产生影响。

为了便于对AGV小车路径进行规划,本文对AGV小车的运动轨迹进行分类,主要包括:

(1)两个目标点位于同一个平面内,且X、Y、Z坐标中仅有一个坐标值不同,此时AGV小车在两目标点间的运动轨迹为直线;

(2)两个目标点位于同一个平面,且X、Y、Z中有两个坐标值不同;

(3)两个目标点位于不同平面,且X、Y、Z中坐标值均不相同。

根据上文分析,假设小车当前所在位置坐标为 $i = (x_1, y_1, z_1)$,目标位置坐标为 $j = (x_2, y_2, z_2)$,则如果AGV小车的运动符合第一种情况,那么其两点之间的位移如式(1)所示。

$$S = |y_1 - y_2| \text{ 或 } |z_1 - z_2| \quad (1)$$

此时,若AVG小车需要在X轴方向上进行操作,那么其需要退出当前所在的货道,即运行到货道口,此时位移如式(2)所示。

$$S = 2|y| + |x_1 - x_2| \quad (2)$$

若AGV小车的运动符合第二种情况,此时虽然在同一个平面内但并非在同一条直线上,因此需要借助辅

助点来实现AGV小车的运动。若AGV小车需要在X轴方向上进行操作,则AGV小车需要退出当前所在的货道,故两目标点之间的位移如式(3)所示。

$$S = |x_1 - x_2| + |y_1| + |y_2| \text{ 或 } |x_1 - x_2| + |z_1 - z_2| + 2|y| \text{ 或 } |y_1| + |y_2| + |z_1 - z_2| \quad (3)$$

若AGV小车的运动符合第三种情况,此时两目标点不在同一个平面内,可以认为AGV小车需要进行跨平面运动。为实现两点之间的运动还需要借助三个辅点,如果需要在X轴方向上进行操作,则需要退出当前所在的货道,故两目标点之间的位移如式(4)所示。

$$S = |x_1 - x_2| + |y_1| + |y_2| + |z_1 - z_2| \quad (4)$$

2 改进蚁群算法三维路径优化

2.1 蚁群算法理论

蚁群算法是由Dorigo根据蚁群外出觅食的行为模拟总结出的一种寻找优化路径的概率性算法。蚂蚁在外出觅食时,将信息素释放在所经过的路径上,其他蚂蚁对信息素的浓度进行感知,如果信息素浓度较高,说明此路径目标点距离自己较近,其他蚂蚁沿着此路径行进的同时也会释放信息素,进而使得此路径上信息素浓度进一步增加,在正反馈机制的作用下,该路径上的信息素越来越多。通过不停迭代计算,最终可以得到两点之间的最短路径。式(5)所示为蚁群算法中蚂蚁选择下一目标点的概率。

$$P_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}^\alpha(t) \cdot \eta_{ij}^\beta(t)}{\sum_{s \notin tabu_k} \tau_{is}^\alpha(t) \cdot \eta_{is}^\beta(t)} & j \notin tabu_k \\ 0 & \text{其他} \end{cases} \quad (5)$$

式中: i, j 分别表示两个目标点; α 表示信息素因子,即信息素浓度的高低会对蚂蚁对此路径是否选择产生一定的影响; β 表示启发式因子,即蚂蚁在路径选择时信息素所占的重要程度; τ_{ij} 表示 i, j 两点之间的信息素浓度; $\eta_{ij}(t)$ 为启发函数, $\eta_{ij}(t) = 1/d_{ij}$, d_{ij} 表示 i, j 两点之间的距离; $tabu_k$ 为禁忌表。

当蚂蚁遍历完所有路径之后返回到初始位置时,实现了一次路径循环,之后再一次对信息素进行更新。此时每条路径上的信息素包括随着时间推移遗留下来的信息素及当前蚂蚁释放的信息素。因此信息素的迭代如式(6)所示。

$$\begin{cases} \tau_{ij}(t+n) = (1-\rho) \cdot \tau_{ij} + \Delta\tau_{ij}(t, t+n) \\ \Delta\tau_{ij}(t, t+n) = \sum_{k=1}^m \Delta\tau_{ij}^k(t, t+n) \end{cases} \quad (6)$$

式中: ρ 为挥发因子,取值范围为 $\rho \in [0, 1)$; $(t, t+n)$ 表示

蚂蚁从初始点到目标点再返回初始点完成一次循环所需要的步数为 n ; $\Delta\tau_{ij}(t, t+n)$ 表示在蚂蚁完成一次路径循环后, i, j 两点的信息素增加的量; $\Delta\tau_{ij}^k(t, t+n)$ 表示蚂蚁 k 在完成一次路径循环后, i, j 两点信息素增加的量。

根据上述对蚁群算法的原理分析, 可知蚁群算法的优势包括: (1) 通过正反馈不断增加路径上的信息素, 使得算法在计算的过程中形成收敛的搜索过程, 最终得到最优路径。(2) 蚁群算法通过蚂蚁个体释放信息素来记录路径, 每个蚂蚁个体都可以对周围环境进行改变, 并可以对环境中的信息素进行感知。(3) 算法为分布式并发计算的过程, 即每个蚂蚁的路径选择概率计算是并行的, 可大大提高算法效率。(4) 通过信息素浓度进行启发式搜索, 有利于找到全局最优路径, 而不会陷入局部最优。

2.2 改进蚁群算法三维路径优化方法

在基本蚁群算法中, 蚂蚁通常会被分配到各个点位, 并且需要对所有的目标点进行遍历, 不同目标点之间的顺序选择在很大程度上取决于转移概率, 而转移概率又与每条路径的启发式因子和信息素浓度有关。

对于货架式立体仓库而言, 要想实现 AGV 小车的路径规划, 必须同时考虑到水平路径选择和垂直路线规划。目前, 基本蚁群算法主要用于解决二维平面路径规划即平面两点之间的路径, 而对于货架式立体仓库这种平行于各坐标轴的折线路径无法适用, 因此需要对基本蚁群算法进行优化。

针对两目标点的三种位置关系, 通过 $cenpt$ 变量来辨别两目标点为何种位置关系, 进而确定需要增加的辅助点的数量。图 4 所示为增加辅助点数量的部分 MATLAB 代码。

```
cenpt = 0;
if C(R(i),1)~C(R(i+1),1)
cenpt = cenpt+1;
end
if C(R(i),2)~C(R(i+1),2)
cenpt = cenpt+1;
end
if C(R(i),3)~C(R(i+1),3)
cenpt = cenpt+1;
end
```

图 4 增加辅助点数量的部分 MATLAB 代码

当 $cenpt = 3$, 表示 AGV 小车在两点之间运动属于第三种情况, 即两目标点的 x, y, z 坐标均不一样, 需要增加三个辅助点 a, b, c , 此时三个点坐标分别为 $a = (x_1, 0, z_1)$, $b = (x_2, 0, z_1)$, $c = (x_2, y_2, z_1)$, 两点之间由 i 到 j 所经过的点的顺序为 $i \rightarrow a \rightarrow b \rightarrow c \rightarrow j$ 。当 $cenpt = 2$, 表示小车在两点之间运动属于第二种情况, 即两目标点有两个坐标值不同。此时可以分成两种情况, 当 x 坐标相同则只需要增加一个辅助点 $a = (x, y_2, z_1)$,

由 i 到 j 所经过的点的顺序为 $i \rightarrow a \rightarrow j$ 。当 y 坐标相同或者 z 坐标相同, 那么小车需要退出货道, 此时需要增加三个坐标点 $a = (x_1, 0, z)$, $b = (x_2, 0, z)$, $c = (x_2, y_2, z)$, 两点之间由 i 到 j 所经过的点的顺序为 $i \rightarrow a \rightarrow b \rightarrow c \rightarrow j$ 。当 $cenpt = 1$, 表示两个目标点只有一个坐标值不同, 此时不需要增加辅助点, 两点之间直接由 i 运动到 j , 即 $i \rightarrow j$ 。本文提出的改进蚁群算法三维空间路径优化方法, 主要分为以下五个步骤:

步骤 1 假设算法中共有 m 个蚂蚁, 同时把这些蚂蚁随机放置在三维空间的不同点上, 设置 Nc_max 为本次计算的最大迭代次数, 并初始化蚁群算法中的各个变量, β 为启发因子, α 为信息素因子, ρ 为随着时间推移信息素的挥发系数, Q 为信息素的增强系数。

步骤 2 对算法中的每个个体的初始信息素进行确定, 并将蚂蚁随机分配到不同的起始点上, 让其逐个选择路径, 并将其经过的目标点放到禁忌表中。路径选择中, 每个蚂蚁根据概率函数来判断应该选择哪个目标点作为下一点, 如式(7)所示。

$$\begin{cases} p_{ij}^s = \frac{(\tau_{ij})^\alpha \cdot (\eta_{ij})^\beta}{\sum \tau^\alpha \cdot \eta^\beta} \\ \eta_{ij} = \lambda \exp(l_{n,e} - l_{n+1,e}) \end{cases} \quad (7)$$

在所有蚂蚁都遍历完一次路径之后, 会对各个蚂蚁的路径距离进行比较, 并记录其中最短路程, 同时会记录所有蚂蚁行走路径的平均数。图 5 所示为实现最短路径选择的部分代码。

```
for i=1:(ceil(m/n)) Randpos =[Randpos ,randperm (n)];
end Tabu (:,1)=( Randpos (1,1:m)); for j=2:n
for i=1:m
visited =Tabu (i,1:(j-1));
J=zeros (1,(n-j+1)); P=J;
Jc=1;
for k=1:n
If length (find (visited ==k))== 0 J(Jc)=k;
Jc=Jc+1;
end
end
%计算待选点的概率分布
for k=1:length (J)
P(k)=( Tau (visited (end),J(k)))*( Eta (visited (end),J(k)));
end
P=P/(sum (P));
%按概率原则选取下一个点
Pcum =cumsum (P); Select =find (Pcum >=rand );
end
end
to_visit =J(Select (1)); Tabu (i,j)=to_visit ;
if NC >=2
Tabu (1,:)=R_best (NC-1,:);
End
```

图 5 选取最短路径的部分代码

步骤 3 将 k 值与 m 值进行比较, 若 $k \geq m$, 则说明 k 蚂蚁已经走完了所有目标点, 此时进入步骤 4, 否则转入步骤 2。

步骤 4 对全局中的信息素、信息素挥发系数、信息素增强系数进行更新,信息素的更新如式(8)所示。

$$\begin{cases} \tau_{ij}(t+1) = \rho \cdot \tau_{ij}(t) + \Delta\tau_{ij}(t) \\ \Delta\tau_{ij}(t) = \sum_{k=1}^n \Delta\tau_{ij}^k(t) \end{cases} \quad (8)$$

式中: $\Delta\tau_{ij}^k(t)$ 表示在本次迭代计算中*ij*之间因蚂蚁*k*所增加的信息素; $\Delta\tau_{ij}(t)$ 表示增加的信息素总和。其实现过程如图 6 所示。

```

L=zeros(m,1);
for i=1:m
R=Tabu(i,:);
for j=1:(n-1)
L(i)=L(i)+D(R(j),R(j+1));%原距离加上第 j 个点到第 j+1 个点的距离
end L(i)=L(i)+D(R(1),R(n));
end L_best(NC)=min(L);
pos=find(L==L_best(NC));
R_best(NC,:)=Tabu(pos(1,:));
L_ave(NC)=mean(L);%此轮迭代后的平均距离
NC=NC+1;
Delta_Tau=zeros(n,n);%更新信息素
for i=1:m
for j=1:(n-1)
Delta_Tau(Tabu(i,j);
Tabu(i,j+1))=Delta_Tau(Tabu(i,j);
Tabu(i,j+1))+Q/L(i);
End
    
```

图 6 更新信息素部分代码

步骤 5 禁忌表清零。在达到最大迭代次数之后,将禁忌表清零,对每次迭代中所获得的最短路径进行比较,并最终得到最短路径和所有目标点的行进顺序,并输出最优路径结果。

图 7 所示为改进蚁群算法实现流程。

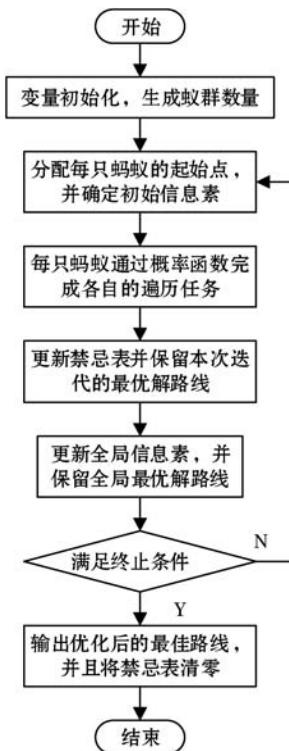


图 7 改进蚁群算法实现流程

3 实验结果与讨论

为了验证改进蚁群算法的有效性,在 Window 10 操作系统下采用 MATLAB 2013a 对算法进行仿真实验,实验硬件环境: Intel (R) Core (TM) i5-3337U Duo CPU 1.8 GHz/8 GB 内存。首先,实验选取具有 12 列,每列 4 层,每层 30 个,共计 1 440 个仓位的仓库;其次,选取可以实现多目标点遍历作业的 AGV 小车,该小车需要满足在水平货道和垂直货架上运动。为了验证所提出算法的性能,将其与遗传算法、基本蚁群算法进行对比实验,实验中所设置的目标点数均相同。在本次实验中,若参数 α 值过大,则搜索路径的随机性会减弱,过小的话会陷入局部最优; β 值过大容易选择局部最短路径,过小算法收敛速度太低。通过计算分析文中信息启发因子 α 和期望启发因子 β 的取值,分别取值为 1.5 和 2。初期为了尽快找到最优解,信息素挥发系数 ρ 取 0.1,算法后期根据实际情况对 ρ 值进行调整。另外,蚂蚁个数 m 取 30,最大迭代次数 NC_{max} 为 100。图 8、图 9 分别为 50 个目标点和 25 个目标点情况下三种算法的收敛曲线。

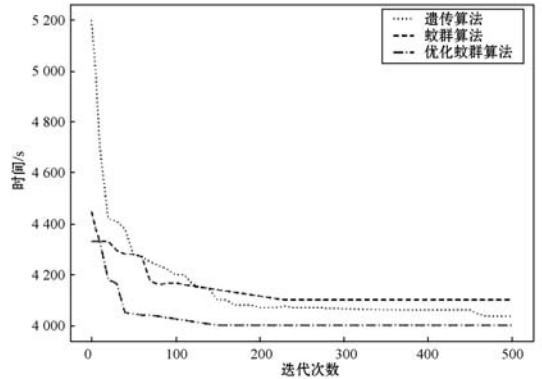


图 8 50 个目标点收敛曲线

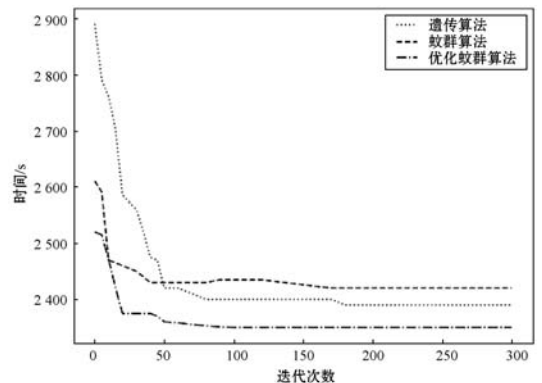


图 9 25 个目标点收敛曲线

由图 8、图 9 比较可知,优化蚁群算法的迭代计算次数最少,反应时间较短。为了验证改进后的蚁群算法能够保证 AGV 小车在每列货架位置都能够正常运行且检测 AGV 小车最大化运行时间,故在每一列货架

随机选取一个目标点。表 1 所示为随机生成的 12 个目标点坐标,其中 X 轴的取值区间为(1,12),Y 轴的取值区间为(1,30),Z 轴的取值区间为(1,4)。

表 1 仿真实验的随机生成的目标点

目标点	1	2	3	4	5	6
坐标	(1,6,2)	(8,17,4)	(5,12,4)	(4,19,1)	(3,4,3)	(7,9,1)
目标点	7	8	9	10	11	12
坐标	(9,29,4)	(2,20,3)	(10,24,2)	(6,13,2)	(12,3,1)	(11,30,4)

图 10 所示为改进后蚁群算法计算出的平均距离和最短距离。可以看出随着迭代次数的增加,平均距离越来越小,由此表明增加迭代次数对于路径规划有一定作用。

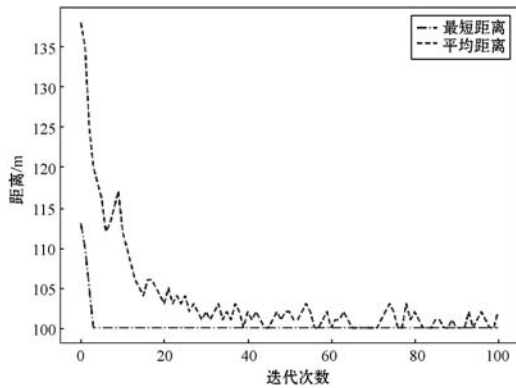


图 10 平均路径和最短路径

图 11 所示为本次实验中所得到的 AGV 小车最优路径轨迹,表 2 为实验中目标点的运行顺序。图 11 中实心圆为最优路径起始点,空心圆为目标点,菱形为辅助点。由此可知,本文设计的优化蚁群算法在货架式立体仓库的 AGV 小车多目标路径规划中有较好的应用效果,算法实现所需要的迭代次数较少,计算速度快,收敛性强,且能获得最短路径。

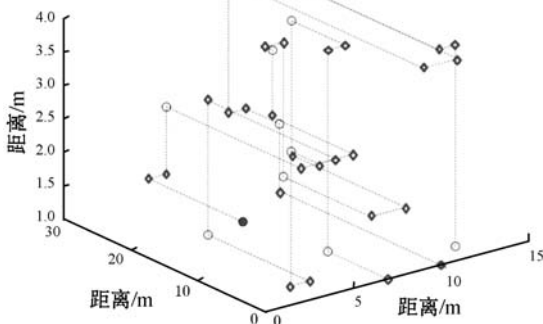


图 11 优化路径轨迹

表 2 仿真实验目标点运行顺序

运行顺序	1-8-4-5-3-10-2-6-9-7-12-11
坐标顺序	(1,6,2) - (2,20,3) - (4,19,1) - (3,4,3) - (5,12,4) - (6,13,2) - (8,17,4) - (7,9,1) - (10,124,2) - (9,29,4) - (11,30,4) - (12,3,1)

4 结 语

本文研究了智能仓储系统中 AGV 小车多目标路径规划,利用蚁群算法将二维平面路径规划扩展到三维空间中,并引入禁忌表对基本蚁群算法进行改进,优化 AGV 小车行驶路径。通过实验得出以下结论:(1) 与其他算法相比,本文改进的蚁群算法所需迭代次数少、收敛性强,能够有效避免算法进入局部最优;(2) 改进后的蚁群算法在智能仓储系统 AGV 小车多目标路径规划中有较好的应用效果,能获得最短路径。

参 考 文 献

- [1] 张原艺,章政,王泉. 基于改进多步长蚁群算法的机器人路径规划[J]. 计算机工程与设计,2018,39(12):3829-3834,3866.
- [2] 吴耕锐,郭三学,吴虎胜,等. 改进多目标蚁群算法在动态路径优化中的应用[J]. 计算机应用与软件,2019,36(5):249-254,288.
- [3] 张金炜,王文扬,郭蓬,等. 基于蚁群四次贝塞尔曲线的无人车路径规划[J]. 现代电子技术,2019,42(13):113-116.
- [4] 梁凯,毛剑琳. 基于改进蚁群算法的室内移动机器人路径规划[J]. 电子测量技术,2019,42(11):65-69.
- [5] 张玮,马焱,赵捍东,等. 基于改进烟花-蚁群混合算法的智能移动体避障路径规划[J]. 控制与决策,2019,34(2):335-343.
- [6] Ali H, Kar A K. Discriminant analysis using ant colony optimization-An intra-algorithm exploration[J]. Procedia Computer Science,2018,132:880-889.
- [7] 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[J]. 计算机工程与应用,2019,55(17):35-43.
- [8] 许凯波,鲁海燕,黄洋,等. 基于双层蚁群算法和动态环境的机器人路径规划方法[J]. 电子学报,2019,47(10):2166-2176.
- [9] Wang H, Zhang H J, Wang K, et al. Off-road path planning based on improved ant colony algorithm [J]. Wireless Personal Communications,2018,102(2):1705-1721.
- [10] 于赫年,白桦,李超. 仓储式多 AGV 系统的路径规划研究及仿真[J]. 计算机工程与应用,2020,56(2):233-241.
- [11] 陈明智,钱同惠,张仕臻,等. 仓储物流机器人集群避障及协同路径规划方法[J]. 现代电子技术,2019,42(22):174-177.
- [12] 王秀红,刘雪豪,王永成. 基于改进 A* 算法的仓储物流移动机器人任务调度和路径优化研究[J]. 工业工程,2019,22(6):34-39.
- [13] 夏青松,唐秋华,张利平. 多仓储机器人协同路径规划与作业避碰[J]. 信息与控制,2019,48(1):22-28.