基于改进蚁群算法的月面机器人的路径规划及虚拟仿真*

刘 畅[1]，王逸璇[1]，王婧馨[1]，张富照[2]，张雪芹[1]，曹 涛[3,4]

(1. 华东理工大学 信息科学与工程学院·上海·200237；
2. 华东理工大学 机械与动力工程学院·上海 ·200237；
3. 上海空间智能控制技术重点实验室·上海·201109；
4. 上海航天控制技术研究所·上海· 201109)

摘 要：针对月面机器人在复杂地形下的路径规划问题，提出了一种改进的蚁群算法。算法构建了栅格化地形图，基于人工势场法改进了蚁群算法的启发函数，加快了算法收敛速度；引入空间信息素划分方法，提高了蚁群在最短路径附近区域的搜索能力；实验证明，改进后的蚁群算法，路径规划成功率显著提高，收敛速度加快。在算法规划出月面机器人的最短路径后，采用虚拟仿真技术，基于 unity3D 构建虚拟月面环境和月球车，直观地展示了月面机器人在月面环境下的路径规划效果。

关键词：月面机器人；路径规划；蚁群算法；人工势场；unity3D
中图分类号：TN911.73；TP391.9 **文献标志码**：A **文章编号**：2096-5974(2021)03-0023-11

Path Planning and Virtual Simulation of Lunar Robot Based on Improved Ant Colony Algorithm

LIU Chang[1], WANG Yixuan[1], WANG Jingxin[1], ZHANG Fuzhao[2], ZHANG Xueqin[1], CAO Tao[3,4]

(1. School of Information Science and Engineering, East China University of Science and Technology, Shanghai 200237;
2. School of Mechanical and Power Engineering, East China University of Science and Technology, Shanghai 200237;
3. Shanghai Key Laboratory of Aerospace Intelligent Control Technology, Shanghai 201109;
4. Shanghai Aerospace Control Technology Institute, Shanghai 201109)

Abstract: An improved ant colony algorithm is proposed to solve the path planning problem of lunar robot in complex terrain. The algorithm constructs a rasterizing topographic map, and based on the artificial potential field, the heuristic function of ant colony algorithm is improved to accelerate the convergence speed of the algorithm. Also, the spatial pheromone partition method is introduced to improve the search ability of ant colony in the area near the shortest path. Experimental results show that the improved ant colony algorithm can significantly improve the success rate of path planning and accelerate the convergence speed. After the algorithm plans the shortest path of the lunar robot, the virtual lunar environment and lunar rover are constructed based on unity3d by using the virtual simulation technology. The path planning effect of the lunar robot in the lunar environment is intuitively displayed.

Keywords: lunar robot; path planning; ant colony algorithm; artificial potential field; unity3D

* 收稿日期：2020-07-22；修回日期：2020-08-31
基金项目：上海航天科技创新基金（SAST2018-086）
作者简介：刘畅（1999—），女，研究方向为移动机器人的路径规划。E-mail：18621055967@163.com
通信作者简介：张雪芹（1972—），女，副教授。E-mail：3445670683@qq.com

0 引 言

月面机器人在月面环境探索、数据信息采集等过程中起着重要作用。为了能够在月球表面自主行走，需要对月面机器人的行走路径进行规划。同时，由于月面机器人所带的能量有限，因此有必要为机器人规划出最短的行走路径。目前，研究最短路径规划的算法有很多，比如 A * 算法、模拟退火算法、遗传算法、粒子群算法等。Li[1]等提出了利用粒子群算法优化蚁群算法参数，合理确定蚁群算法所需要的迭代次数，减小算法的计算规模。但是，该算法的执行周期过长，增加了大量的时间成本。Hu[2]等提出了采用头尾双向搜索策略提高算法的前期效率、引入退火思想以提高跳出局部最优解的概率，减少了算法的能耗，但该算法的收敛速度仍然不够理想。Wang Zhizhong[3]提出了基于头尾双向搜索的 A * 启发函数。但由于相遇即停止的策略，虽然算法的收敛速度获得了加速，但其全局搜索能力也同时出现了下降。Wang[4]提出了自适应启发函数，并采用蚂蚁与目的地之间的欧几里德距离避免了路径搜索的初始盲目性和后期单一性，使得搜索速度及最佳路径成功率获得了提升，并降低了蚂蚁在同一位置落入死锁的概率，但改进后的算法的收敛速度不佳。Han[5]提出了基于确定周边点集(SPS)算法的初始路径生成方式，在几何形状狭小且复杂度高的地图中能够高效生成无冲突路径，但该算法的收敛速度较慢，最佳路径成功率偏低。Tong[6]等提出了利用人工势场法对传统蚁群算法进行改进。算法的收敛速度较优，但所规划的路径较长。

根据月面机器人对路径规划的需求，考虑到蚁群算法具备较强的鲁棒性和搜索能力，并且易于并行实现，本文提出了在蚁群算法与人工势场法融合的基础上，利用空间信息素划分方法，提高寻找最短路径的成功率和算法的收敛速度。

1 机器人路径规划问题建模

机器人路径规划任务是依据所感知到的障碍物的环境信息，按照某种优化指标，规划出一条从给定起点到目标位置的、无碰撞的最优或次最优路径。移动机器人规划路径主要可分为三种类型：基于事例学习的路径规划、基于行为的路径规划和基于环境模型的路径规划。本文研究的月面机器人路径规划问题属于基于环境模型的路径规划，采用了二维栅格法进行环境建模，采用了蚁群算法进行路径规划建模。

1.1 环境建模

环境模型是对环境进行分析时所设定的模型。环境建模包括栅格图、切线图、Voronoi 图、概率图和可视图等[7]。栅格法是对地图进行建模的一种常用方法，简单有效，对不规则障碍物的适应能力强，易于被扩展到三维环境。将地图进行栅格化可以较为清晰地显示出月面地图的环境信息，选择大小合适的栅格既可以保证环境信息的清晰度，又可以提高路径规划的速度。

在栅格化地图中，障碍物处的栅格以涂黑表示，无障碍物处则为空白，如图 1 所示。在算法中，分别用二进制“0”和“1”表示栅格化地图。

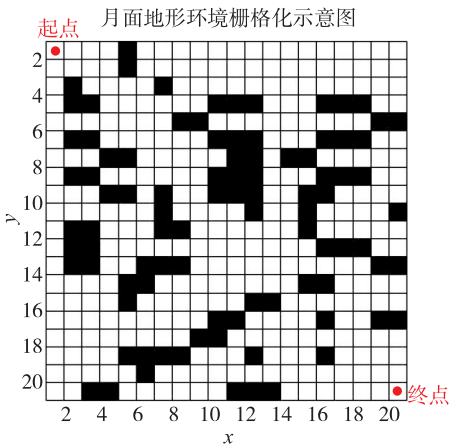


图 1 栅格法月面示意图

Fig. 1 Schematic diagram of the lunar surface of the grid method

图 1 利用“0”和“1”所构成的矩阵建立了 20×20 二维栅格图。网格用整数标记，并且与坐标系的单位长度相同。障碍物的坐标点可用常见的 XOY 正交坐标系来确定。图 1 中，栅格序号值 k 从起点处开始依次向右标记为 1～400，对应的某栅格的坐标可利用如下公式得到

$$\begin{cases} x = \mathrm{mod}(k/m) - 1 \\ y = \mathrm{int}(k/m) \end{cases} \tag{1}$$

式中，mod 为取余运算，int 为取整运算，m 为每一行的栅格数，m、k 均为整数。通过以上方法，可以将二维规划空间离散化。采用栅格法构建环

境空间，采用两点式坐标进行栅格标识，即可以实现月面环境建模。

1.2　机器人路径规划建模

路径规划的目的在于规划出从起始位置到目标位置的最优路径。机器人的路径规划算法包括了遗传算法、A＊算法、人工势场法和蚁群算法等多种算法。其中，蚁群算法具有启发式搜索、并行计算、信息正反馈和鲁棒性强等特点，适合被应用于月面机器人路径规划问题。本文将基于改进的蚁群算法实现对月面机器人的路径规划。

1.2.1　蚁群算法的特点

蚁群算法模拟了自然界中蚂蚁觅食的行为规律。蚂蚁会在走过的路径上留下可被感知的信息素，其他单个蚂蚁在觅食时会选择信息素浓度更大的路径。因此，某一路径上的信息素会以此为积累而越来越大，下一只蚂蚁选择此路径觅食的概率则进一步增加。蚁群算法在路径规划领域有较为广泛的应用，蚁群算法具有以下特点：

（1）蚂蚁可以通过释放信息素来改变周围的环境，且每个个体能够感知周围环境的实时变化，个体间可通过环境进行间接的通信；

（2）采用正反馈机制，使得搜索过程不断收敛，最终逼近最优解；

（3）搜索过程采用分布式计算方式，多个个体可同时进行并行计算，大大提高了算法的计算能力和运行效率；

（4）和其他算法相比，启发式的概率搜索方式不容易陷入局部最优，易于寻找到全局最优解。

1.2.2　蚁群算法的数学模型

设在每个时间 t，根据信息素浓度与启发式函数，蚂蚁 k 由当前节点 i 确定转移到下一节点 j 的转移概率为

$$P_{ij}^{k}(t)=\begin{cases}\dfrac{[\tau_{ij}(t)]^{\alpha}[\eta_{ij}(t)]^{\beta}}{\sum_{s\in \text{allowed}_k}[\tau_{ij}(t)]^{\alpha}[\eta_{ij}(t)]^{\beta}} & \text{if}\quad j\in \text{allowed}_k\\ 0 & \text{else}\end{cases} \tag{2}$$

式中，$P_{ij}^{k}(t)$ 表示 t 时刻蚂蚁 k 从节点 i 移动到下一个节点 j 的转移概率；$\tau_{ij}(t)$ 为 t 时刻 j 点残留的信息素；$\eta_{ij}(t)$ 为节点 i 到节点 j 的启发函数；s 表示蚂蚁能够选择的节点；$j\in \text{allowed}_k$ 表示蚂蚁 k 没有抵达过的点；α 和 β 分别为信息素因子和启发函数因子。其中，启发函数 $\eta_{ij}(t)$ 的定义为

$$\eta_{ij}(t)=\frac{1}{d_{ij}} \tag{3}$$

d_{ij} 为节点 i 到节点 j 的欧氏距离

$$d_{ij}=\sqrt{(x_j-x_i)^2+(y_j-y_i)^2} \tag{4}$$

蚂蚁 k 循环一次后，信息素的更新方式可表示为

$$\tau_{ij}(t+1)=(1-\rho)\tau_{ij}(t)+\sum_{k=1}^{m}\Delta\tau_{ij}^{k} \tag{5}$$

其中，ρ 表示信息素的蒸发速率，$\Delta\tau_{ij}^{k}$ 表示蚂蚁 k 在当前迭代次数下留下的信息素浓度。

$$\Delta\tau_{ij}^{k}=\begin{cases}\dfrac{Q}{L_k} & \text{当蚂蚁 } k \text{ 从节点 } i \text{ 移动到节点 } j\\ 0 & \text{否则}\end{cases} \tag{6}$$

式中，Q 为常量，L_k 表示由蚂蚁 k 搜索到的路径长度。上式表明，蚂蚁留下的信息素浓度与其搜索到的路径长度有关。蚂蚁搜索的路径越短，留下的信息素浓度越高。可见，在蚁群算法中，启发函数和信息素是两个主要的优化参数。

2　月面机器人路径规划优化算法设计

月面机器人携带的能量有限，需要在最短的时间内，规划出起始点到目标点的最短路径。为了进一步提高路径规划的效率，加快蚁群算法的收敛速度以及提升最短路径全局寻优的成功率，本文利用人工势场法，并结合了空间信息素划分改进经典蚁群算法，提高了算法的收敛速度以及最短路径规划的成功率。

2.1　基于人工势场法改进启发函数

人工势场法是由 Khatib 提出的一种虚拟力法，它的基本思想是将机器人在周围环境中的运动设计为一种抽象的人造引力场中的运动。由目标点对移动机器人产生“引力”，由障碍物对移动机器人产生“斥力”，最后通过求取合力来控制移动机器人的运动[8]。利用人工势场法改进启发函数，能够有效避免蚁群算法陷入局部最优，并避免由前期盲目搜索而导致的收敛速度过慢。

（1）引力函数

人工势场法中的引力函数 $G(t)$ 可定义为月面机器人与固定目标点之间的相对距离

$$G(t)=\frac{1}{1+\sqrt{(x_e-x_i)^2+(y_e-y_i)^2}} \tag{7}$$

其中，$(x_i,\ y_i)$ 为机器人当前的坐标，$(x_e,$

y_e）是目标点的坐标。月面机器人的当前节点与目标节点的距离越近，引力越大。

（2）斥力函数

人工势场法中的斥力函数 $R(t)$ 受移动机器人与障碍物的距离影响，可定义为月面机器人与障碍物之间的函数

$$R(t)=\begin{cases}1 & \text{节点}(x_j,y_j)\text{不是障碍物}\\ 0 & \text{节点}(x_j,y_j)\text{是障碍物}\end{cases} \tag{8}$$

其中，$(x_j,\ y_j)$ 是月面机器人下一步要移动到的坐标。“0”表示该节点是障碍物，状态转移函数式（2）则变为 0，这表示月面机器人不会移动到此点，即不会与障碍物发生碰撞。“1”表示该节点不是障碍物，不影响状态转移函数。

（3）改进的启发函数

基于人工势场法改进的蚁群算法启发函数可定义为

$$\eta_{ij}(t)=G(t)R(t) \tag{9}$$

改进后的启发函数可定义为引力函数 $G(t)$ 与斥力函数 $R(t)$ 的乘积。与原启发函数式(3)相比，引力函数 $G(t)$ 只需考虑月面机器人当前节点与目标节点的相对距离，而无需考虑下一节点的位置。引力函数可以引导机器人选择下一节点，使其逐渐逼近目标点，因此可加快蚁群算法的收敛速度[9]；斥力函数使得月面机器人在越接近障碍物时越受到强烈的排斥，进而促使机器人自动规避静态障碍物。

2.2 基于空间划分改进信息素更新算法

由于蚁群算法是模拟蚂蚁觅食行为的群体智能算法，蚂蚁在寻找食物的过程中会在觅食的路径上留下信息素，之后的蚁群可以根据信息素堆积的浓度找到蚁穴与食物之间的最短路径。因此，信息素的更新方式对蚁群算法的性能有着至关重要的影响[10]。

为了进一步加快蚁群算法的收敛速度、提高最短路径的成功率，本文提出了基于空间划分调整信息素更新的方法。当确定好寻迹的起点和终点后，让起点位于栅格地图左上角的顶点，让终点位于栅格地图右下角的顶点，得出相应的栅格地图，并划分对角区域为重要区域，如图 2 所示。设定重要区域的信息素含量较高，设定非重要区域的信息素含量较少。前者可加大蚁群在此区域中的搜索概率，提高收敛速度，防止陷入局部最优。后者可使蚁群在这部分区域的搜索概率降低，减少了搜索的盲目性[11]。该方法可促使蚁群更倾向于在重要区域内完成搜索。

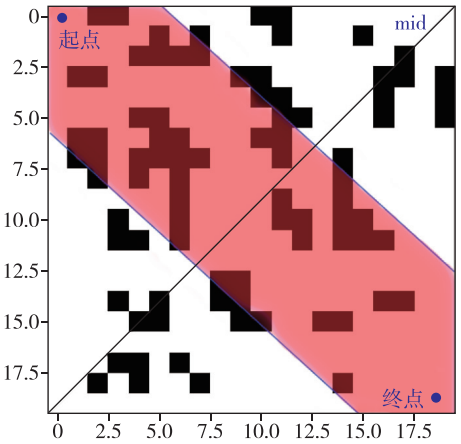


图 2 空间信息素划分区域示意图

Fig. 2 Spatial pheromone zoning map

$$\begin{bmatrix}
1&1&0&0&0&0&0&0&0&0\\
0&1&1&0&0&0&0&0&0&0\\
0&0&1&1&0&0&0&0&0&0\\
0&0&0&1&1&0&0&0&0&0\\
0&0&0&0&1&1&0&0&0&0\\
0&0&0&0&0&1&1&0&0&0\\
0&0&0&0&0&0&1&1&0&0\\
0&0&0&0&0&0&0&1&1&0\\
0&0&0&0&0&0&0&0&1&1\\
0&0&0&0&0&0&0&0&0&1
\end{bmatrix}$$

图 3 信息素划分矩阵

Fig. 3 Information element partitioning matrix

为获取如图 2 所示的空间范围，算法利用如图 3 所示的信息素分布矩阵，为矩阵对角区域赋予了信息素初值，使得地图对角区域内的信息素具备一定的初始浓度。在蚁群进行路径搜索时，重要区域与非重要区域的信息素分布可通过调节信息素的蒸发速率 ρ 来实现

$$\rho=\begin{cases}|x_i-y_i|\sqrt{(x_i-x_s)^2+(y_i-y_s)^2}, & i<\text{mid}\\ |x_i-y_i|\sqrt{(x_i-x_e)^2+(y_i-y_e)^2}, & \text{else}\end{cases} \tag{10}$$

式中，$(x_i,\ y_i)$ 为节点 i 的坐标，$(x_s,\ y_s)$ 为起点坐标，$(x_e,\ y_e)$ 为终点坐标，mid 为地图的

主对角线，该对角线用于区分节点 i 与起点和终点的距离的远近，以采用不同的 ρ 计算公式计算值；$|x_i - y_i|$ 表示节点偏离图 2 所示对角区域的程度，当节点恰好位于起点到终点的对角线上时，取值最小；$\sqrt{(x_i - x_s)^2 + (y_i - y_s)^2}$ 表示节点 i 与起点间的距离；$\sqrt{(x_i - x_e)^2 + (y_i - y_e)^2}$ 表示节点 i 与终点间的距离。

当节点 i 位于如图 2 中红色区域所示的重要区域（即对角区域内）时，根据公式（10）计算而得的 ρ 值较小，这意味着信息素的蒸发速率小，信息素的含量较高。此时，应加大蚂蚁在此区域的搜索概率，防止陷入局部最优；当节点位于非重要区域时，ρ 的取值较大，信息素的蒸发速率较大，信息素的含量较低，则应减小蚂蚁在此区域的搜索概率，以降低搜索的盲目性。

2.3　基于人工势场和空间信息素划分的蚁群算法流程

基于人工势场和空间信息素划分的蚁群算法的流程图如图 4 所示。

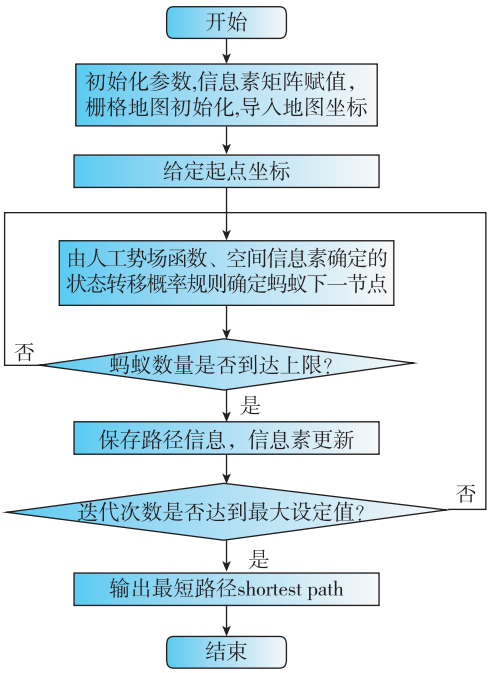


图 4　算法流程图

Fig. 4　Algorithm flow chart

算法步骤如下：

第一步：导入月面地图信息，给定起点与终点坐标，初始化栅格地图；初始化信息启发因子 α、期望启发因子 β、信息素蒸发速率 ρ、蚂蚁数量 m 以及最大迭代次数 $N_{\max}$。通过信息素矩阵对角区域赋予信息素浓度初值，并用于划分区域。

第二步：在起点释放 m 只蚂蚁，根据式（7）、式（8）、式（9）确定启发函数 $\eta_{ij}(t)$，再根据式（2）寻找蚂蚁可行的下一个节点 j，并将此节点加入到禁忌表中。循环此过程，直至达到最大的蚂蚁数量。

第三步：保存路径信息，依据式（5）和式（6），进行信息素的更新。判断当前的迭代次数是否达到了所设定的最大迭代次数 $N_{\max}$。如果达到了设定值，则结束算法，并输出此最短路径信息。

通过利用人工势场法改进启发函数，以及结合空间信息素划分方法，可以有效防止蚁群陷入局部最优，加快蚁群算法的收敛速度，提高最短路径寻优的成功率。

3　月面机器人路径规划虚拟仿真设计

3.1　月面机器人和月面环境建模

为实现月球车在虚拟月面环境上的自主寻路的仿真过程，提升仿真系统的沉浸感和真实感，同时直观展示月球车根据改进蚁群算法实现的寻路过程，可采用 Solidworks、Unity3D、3Dsmax 等软件对月球车和月面环境进行三维建模和渲染。

模拟的玉兔号月球车模型主要由车体、悬架、车轮、太阳翼、定向天线和相机六部分组成，采用 Solidworks 进行了建模。月球车的模型贴图、材质指定、渲染烘焙等工艺采用 3Dsmax 软件完成。在 Unity 提供的地形编辑器中创建月面地形，模拟真实情况，为月面地形添加适合的材质和纹理，结果如图 5 所示。

3.2　Unity 与蚁群算法的交互

蚁群算法的开发基于 Python，Unity3D 不具备直接与 Python 进行数据传递的能力。本文提供了一种依赖于 XML 的读写策略，从而可使 Unity 引擎从外部算法获取输出结果。

在 Unity 中创建一个负责信息交互的脚本，在 Using 中引入 System. Xml、System. IO 两个命名空间，使脚本具有对 XML 文档进行读写操作的功能。在脚本中使用 CreateXmlFile 函数，创建与脚本关联的 XML 文档，并通过 xml. CreateElement 和 element. SetAttribute 函数创建用于存放点位数组的节点，并设置节点的属性。

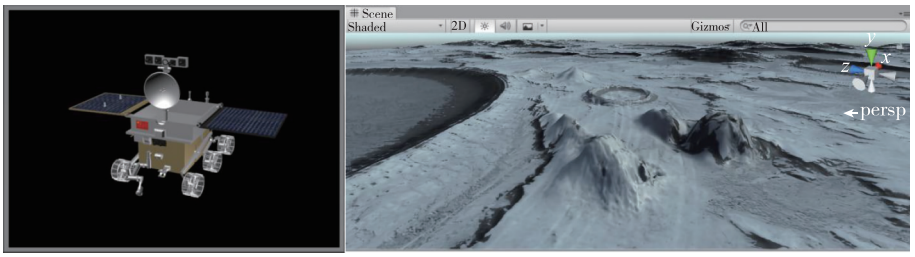


图 5 玉兔号月球车及月面地形建模

Fig. 5 Lunar rover and lunar terrain modeling

在用户选好路径规划的起始与结束点后，Unity 引擎需要将障碍物与路径规划始末点位信息写入 XML 文档中，从而调用蚁群算法。在脚本中，使用 element. AppendChild 及 element. InnerText 函数将数组分别写入不同的节点，并通过 root. AppendChild (element) 函数将节点一层层添加到 XML 中，运用 xml. Save (path) 指定 XML 文件路径并实现文档的保存。

在蚁群算法给出路径规划的结果并将路径信息写入 XML 文档中后，脚本中的 xml. SelectSingNode (节点名称). ChildNodes 与 foreach 函数将遍历所有子节点，并使用 textList. Add 函数将读取到的数组点存入到指定的数组中，供执行寻路的脚本调用。

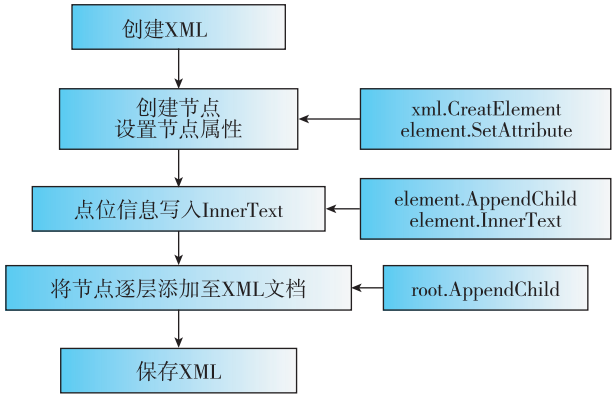


图 6 基于 XML 读写策略的信息交互

Fig. 6 Information interaction based on XML read-write strategy

3.3 月面机器人执行寻路过程

在 Unity 中使物体对象按指定路径移动的方法有多种，主要可分为 Rigidbody 类、Transform 类和 WheelCollider 类函数中的方法。在月球车自主寻路场景中，需要确保寻路过程贴近真实情况。月球车轨迹贴合算法给出的规划路径，可控制多次仿真过程中出现的位置、方向变换差异。为保证寻路路径的准确性、寻路过程的仿真性和降低过程的随机性，此处采用结合 Transform 空间变换类和 WheelColider 轮胎碰撞器类的函数执行寻路的方法。

在月球车位置变换过程中，采用 Vector3. MoveTowards (thisPosiyion, targetPosion, Speed) 对月球车进行移动操作，通过函数 Vector3. Distance 判断移动的距离。在转弯过程中，调用四元数函数 Quaternion. LookRotation 得到月球车应当旋转的参数 rotate，最后调用四元数插值函数 Quaternion. Slerp (transform. localRotation, rotate, angleSpeed)，实现月球车方向的平滑调转。

在此过程中，WheelColider 轮胎碰撞器的作用表现为绑定月球车实体模型，在月球车转弯过程中提供前轮转向条件、悬架系统仿真和对轮胎摩擦碰撞情况仿真的功能上，与 Transform 类函数、物理引擎互不干涉。通过在脚本中定义轮胎碰撞器变量 WheelColider 与车轮模型变量 WheelMesh，为变量指定在场景中的轮胎碰撞器与车轮模型对象，使用 WheelColider. GetWorldPose (out, out) 函数为 WheelMesh. transform 参数进行赋值，从而使车轮模型的位置和角度始终跟随轮胎碰撞器，以实现寻路过程的移动仿真。

该种方法中的 Transform 类函数可确保月球车位置和方向的正确性，WheelColider 类函数可确保物理效果的真实度。该方法使得寻路过程的仿真程度较高，月球车的运动状态在符合动力学原理的同时，能很好地确保月球车行进路线和规划路线之间的吻合程度。

3.4 月面机器人路径规划虚拟仿真实现流程

在基于 Unity 搭建的虚拟场景中，当引擎对月面地形及月球车模型的渲染烘焙完毕时，用户开始选择路线规划的起始点与终点；当选择完毕

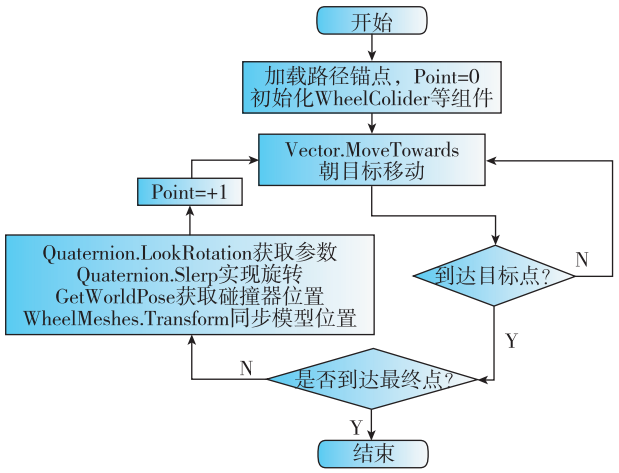


图 7　执行寻路方法的流程图

Fig. 7　Flowchart of the execution pathfinding method

时，Unity 将三维地形模型经过二维栅格化并加以筛选后的障碍物点位以及用户选择出的始末点位以二维数组的形式传递给蚁群算法；蚁群算法在接收到带有障碍物和规划信息的数组后，返回规划出的路线点位；在获取了蚁群算法给出的路线后，虚拟场景将渲染出规划路线的可视化轨迹。月球车将按照蚁群算法给出的路线行进，直至抵达用户指定的终点，具体流程如图 8 所示。

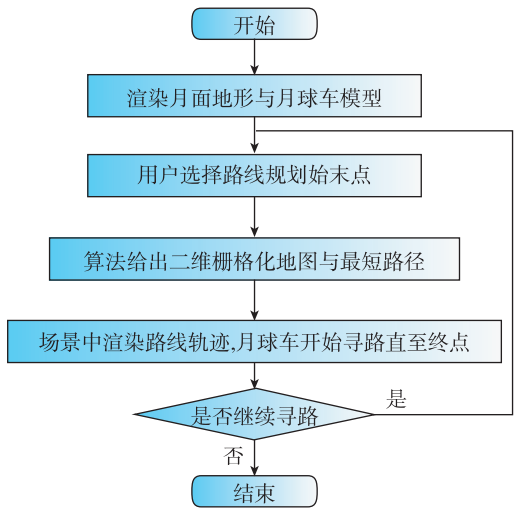


图 8　仿真及虚拟展示实现流程

Fig. 8　Simulation and virtual display implementation process

为了真实再现月球车在月面行进的情况，采用 Unity 内置的 NVIDIA 的 Physx 物理引擎，赋予月球车一定的物理特性和运动特性，展示如刚体碰撞、轮胎碰撞、车辆驾驶、重力影响等效果的仿真。

4　仿真验证与结果分析

4.1　实验一：参数优化

本实验用于寻找信息启发因子 α、期望启发因子 β、信息素蒸发速率 ρ 的最优组合值，使得算法的成功率最高，收敛速度最快。

信息启发因子 α、期望启发因子 β、信息素蒸发速率 ρ 均对算法的收敛速度和成功率有一定的影响。α 过大，会增大路径上的信息素影响权值，导致蚂蚁容易陷入局部最优；α 过小，路径上的信息素对蚂蚁的影响降低，导致蚂蚁行走的随机性提高，算法的收敛速度减慢。相较于 α，β 的波动对最短路径的影响较小。β 过大，蚂蚁在某个局部点上选择局部最短路径的可能性越大，但蚂蚁搜索最优路径的随机性减弱，易陷入局部最优；β 过小，将导致蚂蚁群体陷入纯粹的随机搜索。在这种情况下，很难找到最优解。ρ 过大，会导致蚂蚁在搜索最短路径时陷入局部最优，无法找到全局最优解；ρ 过小，则不能较快地找到最优路径，收敛速度也会较慢。在寻找较优的参数组合之前，通过多次反复实验大致确定各参数最优值所在的范围，在各个参数范围内采用控制变量法试凑而得到接近最优值的参数组合[12]。

实验地图如图 5 所示，网格坐标用整数标记且与坐标单位长度相同。将蚂蚁行走起点设为 (0，0)，终点设为 (19，19)。蚂蚁数量为 20 只，迭代次数为 50 次。初始 $\alpha=1$、$\beta=13$、$\rho=0.8$，采用控制变量法，分别在一定范围内改变参数的大小，实验结果如表 1、表 2 和表 3 所示。

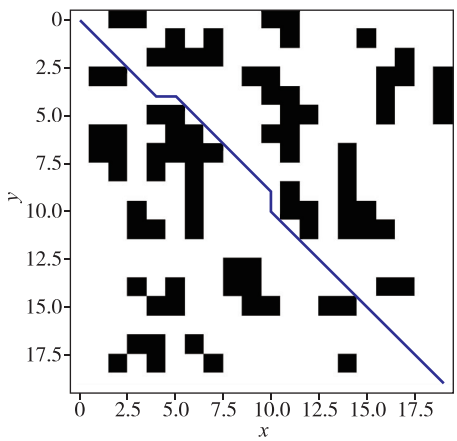


图 9　参数寻优测试地图

Fig. 9　Parameter optimization test map

表 1 固定参数 β、ρ，改变参数 α 下的最短路径成功率和平均收敛次数

Tab. 1 Shortest path success rate and average convergence times β、ρ, changing parameter α with fixed parameters

参数 α	最短路径成功率	平均收敛次数
0.7	78%	41
0.9	90%	50
1.1	98%	19
1.3	**100%**	**1**
1.5	98%	4
1.6	98%	6

表 2 固定参数 α、ρ，改变参数 β 下的最短路径成功率和平均收敛次数

Tab. 2 Shortest path success rate and average convergence times α、ρ, changing parameter β with fixed parameters

参数 β	最短路径成功率	平均收敛次数
8	98%	2
12	98%	22
15	**100%**	**1**
18	100%	1
20	100%	1
30	94%	27

表 3 固定参数 α、β，改变参数 ρ 下的最短路径成功率和平均收敛次数

Tab. 3 Shortest path success rate and average convergence times α、β, changing parameter ρ with fixed parameters

参数 ρ	最短路径成功率	平均收敛次数
0.4	98%	20
0.5	**100%**	**1**
0.6	100%	1
0.8	100%	1
0.9	100%	1
1	98%	24

从表 1、表 2 和表 3 可以看出，当 $\alpha=1.3$、$\beta=15$、$\rho=0.5$ 时，算法的性能达到了最佳。

为进一步验证算法的收敛速度，给出算法收敛次数曲线如图 10 所示。从图 10 可以看到，传统蚁群算法由于前期的震荡而收敛速度较慢，在迭代了 42 次之后才出现了收敛，而本文改进的蚁群算法的收敛速度明显加快，在迭代 4 次时就已经出现了收敛，大大提高了收敛速度。

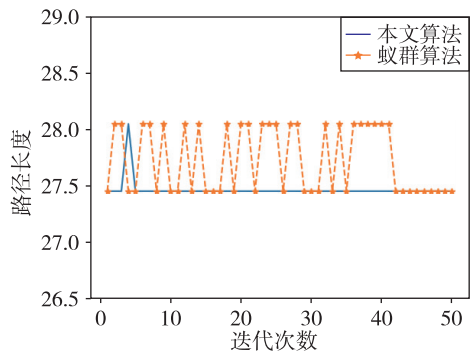
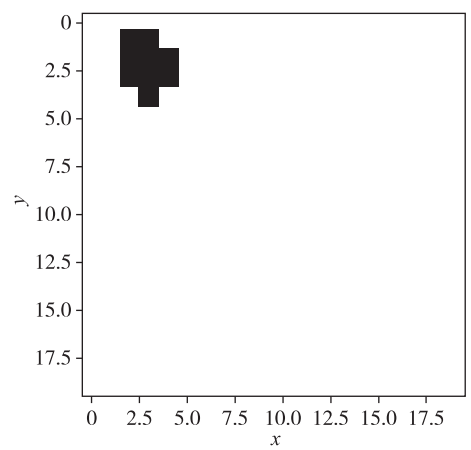


图 10 算法收敛曲线对比

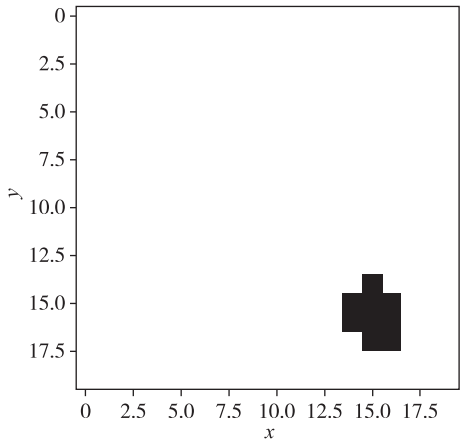
Fig. 10 Comparison of convergence curves

4.2 实验二：比较实验

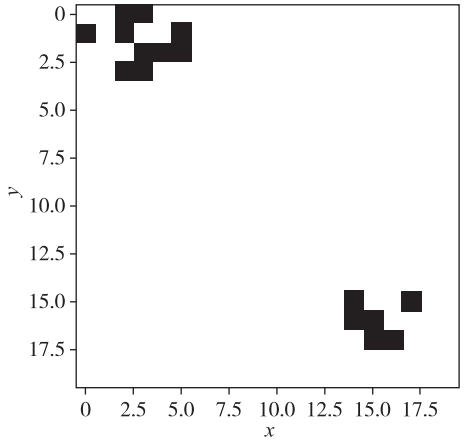
为验证本文算法的性能，需模拟五种不同障碍物环境进行实验，分别为仅起点出现障碍物、仅终点出现障碍物、起点与终点均存在障碍物、对角线中间区域出现简单障碍物和复杂障碍物环境，如图 7 所示。将本文所提到的蚁群算法和文献［13］所提到的基于人工势场的蚁群算法、遗传算法、A * 算法进行了比较，验证了最短路径的规划能力。在栅格图中，设路径规划起点为（0，0），终点为（19，19），蚂蚁数量为 20 只，迭代次数为 100 次，实验结果如图 11 和表 4 所示。



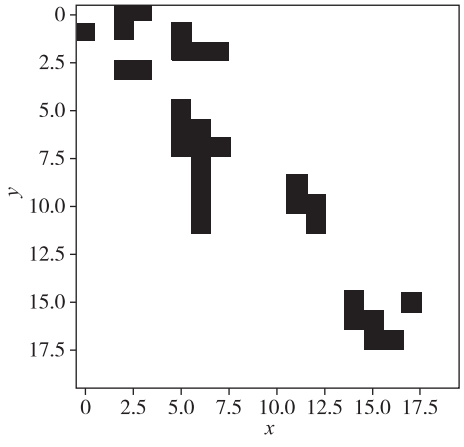
(a) 仅起点处存在障碍物



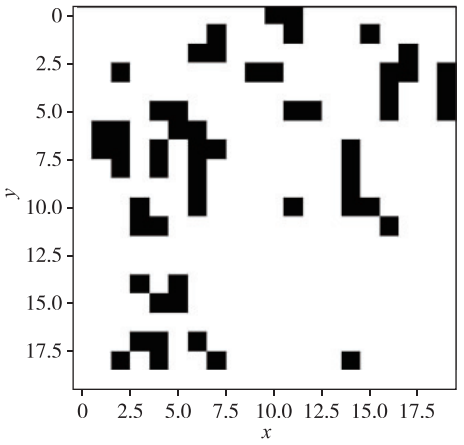
(b) 仅终点处存在障碍物



(c) 起点终点均存在障碍物



(d) 简单地形环境



(e) 复杂地形环境

图 11　五种不同的月面地形环境

Fig. 11　Topographic environment of five different lunar surfaces

表 4　五种不同环境下的最短路径

Tab. 4　Shortest paths in five different environments

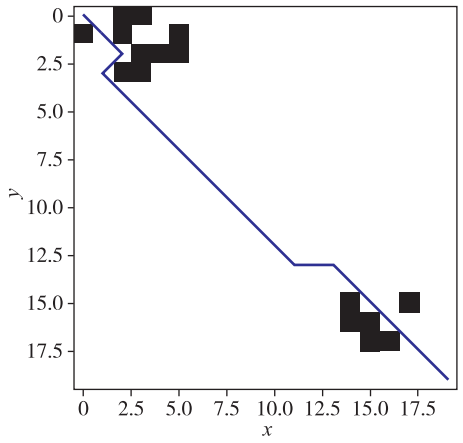
算法名称	地形 a	地形 b	地形 c	地形 d	地形 e
A* 算法	28.63	28.63	28.63	寻路错误	寻路错误
遗传算法	28.04	28.87	31.46	27.45	28.04
基于人工势场的蚁群算法[13]	28.04	28.04	28.87	29.21	28.04
本文算法	28.04	28.62	28.04	27.44	27.45

实验结果显示，A* 算法在于不同障碍物环境下进行最短路径寻优时，在图 11（d）与图 11（e）环境下发生了寻路错误；遗传算法在进行最短路径寻优时，没有出现寻路错误，但其所规划的路径长度不是最短路径。蚁群算法具备较优的路径寻优能力，而本文提出的结合空间信息素划分的改进蚁群算法能够规划出最短的路径长度。

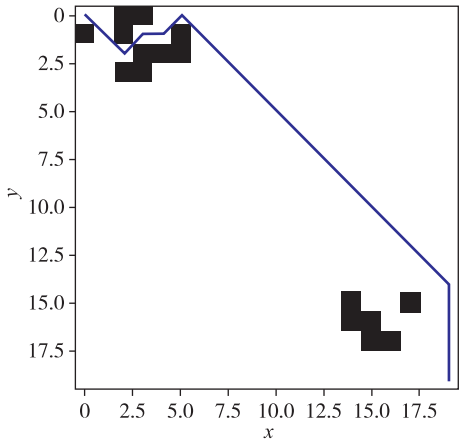
图 12 给出了一组对照示例。图 12（a）为本文算法在地形（c）下的路径规划，图 12（b）是遗传算法在地形（c）下的路径规划。由图 12 可以看到，本文所提算法能够使规划路径较好地保持在对角区域内，使得路径最短。

4.3　实验三：展示实验

为还原玉兔号月球车实现路径规划的过程，直观展示本文算法性能，在Unity3D搭建的寻路



(a) 本文算法在对角区域内可规划最优路径



(b) 遗传算法在对角区域内无法规划最优路径

图 12 重要区域内路径寻优对照组示例

Fig. 12 Example of path optimization control group in important areas

场景及在后端蚁群算法的支持下，可模拟玉兔号在月球上的智能导航、自动寻径过程。在场景中，用户可选择玉兔号寻路的目标点，系统内的蚁群算法将给出最短规划路线，玉兔号将沿规划路线行驶至目的地。

图 13 展示了用户通过界面选择目标点后，系统调用后端蚁群算法给出规划路径的场景。

图 14 展示了玉兔号月球车的寻路场景。为方便用户观察寻路过程，实验场景增添了微缩二维栅格地图，用以指示玉兔号的实时位置和行进方向。

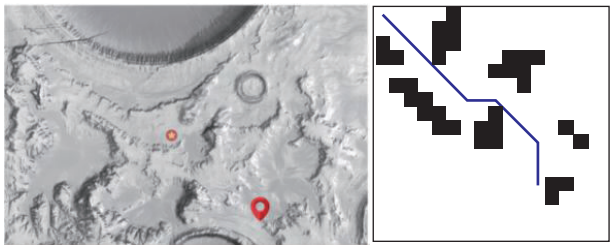


图 13 目标点选择及路径规划场景

Fig. 13 Target point selection and path planning scenarios

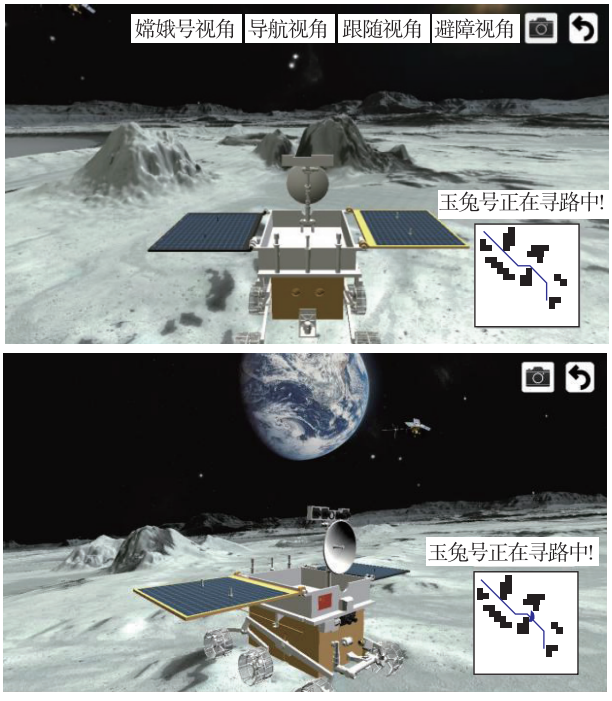


图 14 玉兔号月球车的寻路场景

Fig. 14 The Yutu lunar rover's pathfinding process

5 结 论

本文提出了采用蚁群算法制定月面机器人在月面环境下的路径规划。同时，为了提高蚁群算法路径规划的成功率和收敛速度，将蚁群算法与人工势场方法进行了结合，并提出了基于空间信息素划分的方法。通过提高蚁群在最短路径区域附近进行搜索的可能性，提高了最短路径规划的成功率。实验证明，蚁群算法寻找最短路径的成功率获得了显著提高，算法的迭代次数减少，实现了预定目标。通过建模以及采用Unity3D等技

术，蚁群算法在三维虚拟场景中展示了月面机器人的路径规划过程。

参考文献（References）

[1] 李擎，张超，陈鹏，等．一种基于粒子群参数优化的改进蚁群算法［J］．控制与决策，2013，28（6）：873-878，883.
LI Q，ZHANG C，CHEN P，et al. An improved ant colony algorithm based on particle swarm optimization［J］. Control and Decision，2013，28（6）：873-878，883（in Chinese）.

[2] 胡春阳，姜平，周根荣．改进蚁群算法在 AGV 路径规划中的应用［J］．计算机工程与应用，2020，56（8）：270-278.
HU C Y，JIANG P. ZHOU G R. Improved ant colony algorithm in the application of AGV path planning［J］. Computer Engineering and Application，2020，56（8）：270-278（in Chinese）.

[3] 王志中．基于改进蚁群算法的移动机器人路径规划研究［J］．机械设计与制造，2018，(1)：242-244.
WANG Z Z. Research on path planning of mobile robot based on improved ant colony algorithm［J］. Machinery Design and Manufacturing，2018，(1)：242-244（in Chinese）.

[4] WANG L. Path planning for unmanned wheeled robot based on improved ant colony optimization［J］. Measurement and Control，2020，53（5-6）：1014-1021.

[5] HAN J，SEO Y. Mobile robot path planning with surrounding point set and path improvement［J］. Applied Soft Computing，2017，57：35-47.

[6] 童帮裕，胡坚堃．基于改进蚁群算法的船舶冰区航行路径规划［J］．中国航海，2020，43（1）：24-28.
TONG B Y，HU J K. Navigation path planning for ships in ice area based on improved ant colony algorithm［J］. China Maritime，2020，43（1）：24-28（in Chinese）.

[7] 郭银景，孟庆良，孔芳，等．AUV 路径规划算法研究现状与展望［J/OL］．计算机科学与探索：1-16［2020-08-27］. http：//kns. cnki. net/kcms/detail/11. 5602. TP. 20200727. 1736. 002. html.
GUO Y J，MENG Q L，KONG F，et al. Current situation and prospect of AUV path planning algorithm［J/OL］. Journal of Frontiers of Computer Science and Technology：1-16［2020-08-27］（in Chinese）. http：//kns. cnki. net/kcms/detail/11. 5602. TP. 20200727. 1736. 002. html.

[8] LIANG X X，LIU C Y，SONG X L，et al. Research on improved artificial potential field approach in local path planning for mobile robot［J］. Computer Simulation，2018，35（4）：291-294，361.

[9] ZHU S N，ZHU W Y，ZHANG X Q，et al. Path planning of lunar robot based on dynamic adaptive ant colony algorithm and obstacle avoidance［J］. International Journal of Advanced Robotic Systems，2020，17（3）：1-14.

[10] 张松灿，普杰信，司彦娜，等．蚁群算法在移动机器人路径规划中的应用综述［J］．计算机工程与应用，2020，56（8）：10-19.
ZHANG S C，PU J X，SI Y N，et al. Application review of ant colony algorithm in path planning of mobile robots［J］. Computer Engineering and Applications，2020，56（8）：10-19（in Chinese）.

[11] 康冰，王曦辉，刘富．基于改进蚁群算法的搜索机器人路径规划［J］．吉林大学学报（工学版），2014，44（4）：1062-1068.
KANG B，WANG X H，LIU F. Path planning of search robot based on improved ant colony algorithm［J］. Journal of Jilin University（Engineering Science），2014，44（4）：1062-1068（in Chinese）.

[12] 李根，李航，张帅阳，等．基于蚁群算法的最优路径规划及参数研究［J］．中国科技论文，2018，13（16）：1909-1914.
LI G，LI H，ZHANG S Y，et al. Optimal path planning and parameter research based on ant colony algorithm［J］. China Sciencepaper，2018，13（16）：1909-1914（in Chinese）.

[13] ZHU W Y，ZHU S N，ZHANG X Q，et al. Path planning of lunar robot based on an adaptive ant colony algorithm［C］// 2019 5th International Conference on Robotics and Artificial Intelligence. Singapore：ACM press，2019：24-29.