

# 基于改进蚁群算法的水下无人机路径规划研究

杨海清, 芦斌

(浙江工业大学 信息工程学院, 杭州 310014)

**摘要:** 为实现水下无人机在水中自主作业的功能, 对其设计一套合理的路径规划方案是非常有必要的; 蚁群算法针对水下无人机路径规划方面有着非常好的效果, 拥有不错的鲁棒性, 但是传统的蚁群算法在解决路径规划问题时很容易出现局部最优解的问题; 以传统蚁群遗传算法理论为根据, 对其进行添加目标引导素、构建精英蚂蚁体系、更新信息素浓度这三方面的改进, 使用栅格法构建水下环境分析模型, 并以最短的路径为目的, 规划一条从初始状态到目标状态的无碰安全途径, 运用仿真的办法展开验证; 结果显示: 相较于传统算法, 改进后的算法在求解速度和全局求解能力上有较大的优势。

**关键词:** 蚁群算法; 路径规划; 水下无人机; 环境建模

## Research on Path Planning of Underwater UAV Based on Improved Ant Colony Algorithm

Yang Haiqing, Lu Bin

(College of Information Engineering, Zhejiang University of Technology, Hangzhou 310014, China)

**Abstract:** In order to realize the autonomous operation of underwater drones in waters, it is necessary to design a reasonable path planning scheme for them. The ant colony algorithm has very good effects on the path planning of underwater drones, and has good robustness, but the traditional ant colony algorithm is prone to the problem of local optimal solution when solving the path planning problem. Based on the traditional ant colony genetic algorithm theory, it is improved by adding target guides, constructing an elite ant system, and updating the pheromone concentration. The grid method is used to build an underwater environment analysis model. The purpose is to plan a collision-free safety path from the initial state to the target state and use simulation to carry out verification. The results show that compared with traditional algorithms, the improved algorithm has greater advantages in solving speed and global ability.

**Keywords:** ant colony algorithm; path planning; underwater drone; environmental modeling

### 0 引言

随着现代计算机信息科学的不断发展, 人工智能和海洋大数据的应用领域也在不断扩大, 无人机的最优路径选择和规划成了其重要的组成部分。我国拥有漫长的海岸线和包括内海在内的四百多平方公里的水域, 所以水下无人机与海洋的资源开发综合利用的问题也已经变得越来越亟待解决。水下无人机已经成为了对海洋资源的开发综合利用的重要技术工具, 在面对这种复杂多变的海洋生态环境时, 目前的水下无人机在最优路径规划上依然是个重要课题, 很多算法在路径规划上的效果并不令人满意, 很难得到较好的结果<sup>[1-2]</sup>。

蚁群算法针对水下无人机路径规划方面有着非常好的效果, 拥有不错的鲁棒性和全局性, 但在面对复杂的海洋环境时, 往往出现局部最优解、收敛速度慢的不足之处。本文在传统算法基础上进行研究, 针对传统算法的不足之处进行改进, 建立了基于正态分布的自适应蚁群算法, 对

添加目标引导素、精英蚂蚁体系、更新信息素浓度这 3 个方向进行改进。进而确定了改进蚁群算法在路径规划上的应用步骤, 并对其实验仿真。仿真结果表明, 相较于传统的蚁群算法, 经过改进的蚁群算法具有更好的收敛速度, 能够准确求得最优路径, 具有很强的可行性。

### 1 问题描述

#### 1.1 水下环境建模

面对复杂的水下环境, 水下无人机在水下航行过程中也面临的种种威胁, 包括来自洋流、海底火山地震、地质变化及水下动植物带来的问题<sup>[3]</sup>。要完成水下无人机的整个航行过程, 我们不仅仅需要机器本身拥有更优秀的结构和材料, 也需要研究人员设计一套更加完善可靠的规划方案。

要完成整个水下无人机的路径规划方案, 就需要先建立起一个高效的水下环境的模型, 将复杂的水下环境抽象成为计算机能够识别的地图模型, 抽象表达的方式能够使得计算机的计算效率就可以大幅度提升, 环境模型建立的好坏会直接导致路径寻优的成功与否, 通过不断优化水下环境的模型, 是整个系统朝着更加安全可靠的方向进行<sup>[4-5]</sup>。

根据水下环境的图片运用栅格法对数据进行处理, 把水下环境比作成一个二维的平面, 然后将这个二维的平面进行

收稿日期: 2020-03-31; 修回日期: 2020-04-16。

**作者简介:** 杨海清(1971-), 男, 浙江温岭人, 博士, 副教授, 主要从事无线传感与网络控制技术及应用方向的研究。

**通讯作者:** 芦斌(1994-), 男, 山东滨州人, 硕士研究生, 主要从事无线传感与网络控制技术及应用方向的研究。

分割, 划分成  $m \times n$  个相同面积的方块作为小栅格, 这样我们对每个栅格进行了赋值, 从而就将复杂的水下环境用简单的栅格表达出来<sup>[6-7]</sup>。用白色栅格代表可自由行进的空间, 用黑色栅格代表不可行进和障碍物空间, 如图 1 所示。

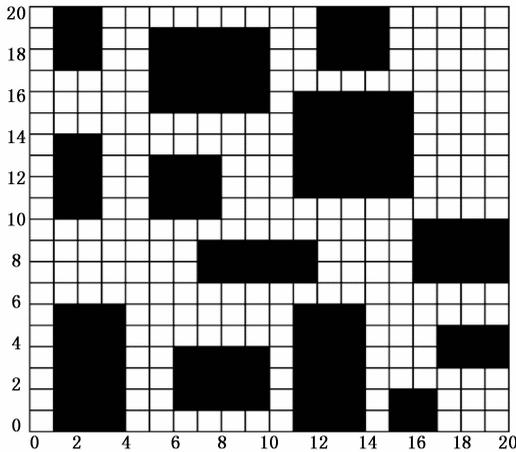


图 1 二维环境栅格模型

在进行建立二维环境的模型时, 使用这种建模方法最终能否精确获得所求解的重要因素就是划分出的栅格的大小, 若栅格面积大, 整个环境模型的信息保留少, 计算机处理速度加快, 能有效地避免干扰, 能够快速求出最优路径, 但是这样会使得环境中的信息不完整, 构建的环境模型模糊, 无法准确地进行规划, 容易造成结果错误; 相反地, 若栅格面积分割的特别小, 会使得构建的环境模型清晰, 但这样就使得计算机处理速度缓慢, 虽然会增大获得最优路径的几率, 寻得最优解, 但其实时效性较差, 无法快速地计算出最优的路径。

### 1.2 蚁群算法基本原理

蚂蚁之间进行信息交流的媒介就是自身分泌的气味, 在蚂蚁群运动的过程中, 它们往往能够在其要走过的路上留下分泌物, 这种分泌物就能够引导其他的蚂蚁也在这个路径上行走。当蚂蚁在路上遇到障碍物的时候, 蚂蚁就会以相同概率选择一个方向, 久而久之, 每一个通往目标处的路上都会存在着信息素<sup>[8-9]</sup>。但由于蚂蚁在经过的短路径上, 信息素的浓度就会比其他路上的浓度高, 当其他蚂蚁再进行选择时, 就会更加倾向于信息素浓度大的路线, 这样就会使得后面的蚂蚁更多的通过这条路, 而其它路上的蚂蚁就越来越少<sup>[10]</sup>。蚂蚁觅食的原理如图 2 所示。

蚁群算法在寻找最短路径时能够运用正反馈的原理, 在最短的路径上不断地增大信息素的浓度, 这种随时间连续增大的浓度, 可以加快系统的运算速度。而负反馈的加入, 就尽可能地避免出现局部最优解, 使得整个算法得到一个正确解。

根据蚁群觅食的这些规则, 蚂蚁在觅食路线上会留下一定量的信息素, 后面蚂蚁将会根据留下来的信息素的浓度对下一步路线进行选择, 这个状态可用概率公式表示为:

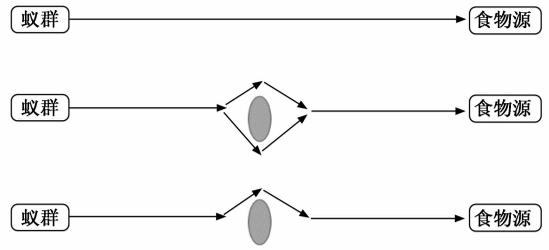


图 2 蚁群觅食路径选择原理图

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

其中:  $p_{ij}^k$  表示  $t$  蚂蚁在  $t$  时刻时, 从节点  $i$  移动到节点  $j$  的概率;  $\alpha$  为信息启发式因子;  $\tau_{ij}$  表示在  $t$  时刻, 蚂蚁在路径留下信息素的量;  $\mu_{ij}$  表示启发函数,  $\beta$  表示期望启发式因子。信息启发式因子  $\alpha$  的值越大, 信息素对蚁群的影响就会越大; 反之, 信息素对蚁群的影响就越小。  $d_{ij}$  表示两个节点  $i$  和  $j$  之间的距离, 它与启发函数的值成反比, 随着其取值的变小, 启发函数值也就越大, 此时蚂蚁从  $i$  到达  $j$  的概率就越大。

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

蚂蚁每次到达一个节点时, 就将这个节点排除在以后的前进目标中, 这样就保证了每个节点只能被选取一次, 当所有的节点都被蚂蚁排除的时候, 蚂蚁就相当于对环境地图的所有能到达的节点都经过, 这时就要对这个排除单的目录重新刷新一次, 后续的蚂蚁就将继续进行。每一只蚂蚁在所经过的路上也将会留下一定量的信息素, 这些信息素将随着时间慢慢地消失, 这就要求合理的控制信息素的浓度, 保证蚂蚁达到的概率。经过时间  $n$  秒后, 信息素浓度更新公式如下:

$$\tau_{ij}(t+n) = (1-\rho) * \tau_{ij}(t) + \Delta\tau_{ij}(t), \rho \in (0,1) \quad (3)$$

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

其中:  $\rho$  表示信息挥发系数;  $\Delta\tau_{ij}$  表示路径上留下来的信息素总量; 当  $t=0$  时, 路径上留下的信息素为 0。

信息素的更新有不同的方式, 根据方式的不同, 下面提供 3 种计算信息素总量的方法。

蚂蚁循环模型:

$$\Delta_{kij}(t) = \begin{cases} \frac{Q}{L_k}, & \text{当第 } k \text{ 蚂蚁在这个循环中经过路径 } i \text{ 到 } j \text{ 时} \\ 0, & \text{其他} \end{cases}$$

蚂蚁数量模型:

$$\Delta_{kij}(t) = \begin{cases} \frac{Q}{d_{ij}}, & \text{当第 } k \text{ 蚂蚁在 } (t, t+1) \text{ 中经过路径 } i \text{ 到 } j \text{ 时} \\ 0, & \text{其他} \end{cases}$$

蚂蚁密度模型:

$$\Delta_{kij}(t) = \begin{cases} Q, & \text{当第 } k \text{ 蚂蚁在这个循环中经过路径 } i \text{ 到 } j \text{ 时} \\ 0, & \text{其他} \end{cases}$$

通过对比这 3 种模型的公式，可以看出它们对信息素浓度的计算方法有差别，3 种公式的  $Q$  表示路径上信息素的强度， $L_k$  表示蚂蚁经过的路线长度<sup>[11-12]</sup>。蚂蚁循环模型中，信息素浓度跟蚂蚁在觅食中走过的路线的长度有关；蚂蚁数量模型中，信息素浓度与两节点之间的距离有关系；蚂蚁密度模型中，信息素浓度完全取决于信息素的强度这个常量。分析公式可以看出，当信息素强度变大时，得到的信息素浓度也就变大，这样就会使得蚁群过早地找到一条信息素含量高的路线，这种情况下容易出现局部最优解。由此可见，蚂蚁循环模型在解决路径规划问题上有更好的优势，能够更方便地计算。

## 2 改进蚁群算法

针对传统蚁群算法的不足，我们将从以下几个方面做出改进：

首先是在算法中加入目标引导素。蚂蚁在节点移动时更加地具有目的性，更好地提高算法的求解效率，蚂蚁能够在初始阶段就开始确定搜索范围，这样就可以直接朝着更接近最终目的地的方向寻找最优解，改进后的算法将会显著提高收敛的速度，减少了不必要的资源消耗。

其次确立精英蚂蚁体系。蚁群在路径寻优时，动态的调整每只蚂蚁所带信息素的量，经过路径短的蚂蚁携带信息素的量增大，经过路径长的蚂蚁携带信息素的量降低。这样就能对后面蚂蚁起到正反馈作用，增加求得解的数量，有效地避免出现局部最优解。

最后，更新信息素的浓度。传统蚁群算法时，蚂蚁在每条路径上留下的信息素的量是相同的，这样就会导致出现多个路径信息素总浓度相似，容易出现非最优解。这时我们对信息素挥发系数进行动态调整，增加蚂蚁所经过路径的数量，避免陷入局部解的误区，改进后的算法将会求得更多的解，面对数量大的蚁群时也会求得最优解。

### 2.1 添加目标引导素

算法在初始搜索阶段，由于每条路径经过的蚂蚁数量较少，使得后面蚂蚁没法根据信息素的浓度来判断下一个节点，这就使得蚂蚁随机的前往其余所有节点，这种盲目的搜索，极大地增加了算法的运算时间，也将会占用更多的资源。我们在改进的算法中添加目标引导素：

$$\lambda_{iD} = \frac{m - m_k}{m} * \frac{N_{cmax} - N_c}{N_{cmax}} * \frac{1}{d_{iD} + d_{ij}} \quad (5)$$

式中， $m$  为蚁群总数量； $m_k$  为当前蚁群数量； $N_{cmax}$  为最大迭代次数； $N_c$  为当前迭代次数； $d_{iD}$  为节点  $i$  与终点的长度； $d_{ij}$  为节点  $i$  与节点  $j$  的长度。

添加上目标引导素之后，节点之间的移动概率就变为：

$$p_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}^\alpha(t) \mu_{ij}^\beta(t) \lambda_{iD}^\gamma(t)}{\sum_{s \in allowed_k} \tau_{is}^\alpha(t) \mu_{is}^\beta(t) \lambda_{iD}^\gamma(t)}, & j \in allowed_k \\ 0, & otherwise \end{cases} \quad (6)$$

其中： $jD$  和  $sD$  为当前节点的引导素和下一个节点的引导素。蚁群搜索初始阶段时，各条路线上信息素浓度接

近，此时的引导素较小，移动概率就较低，此时蚁群可以不断地进行搜索，随着距离终点位置的接近，路线上信息素的浓度相应地增大，引导素也不断增大，此时蚂蚁移动到目标节点的概率增大，降低了蚁群搜索的盲目性。加入引导素的蚁群算法使得蚂蚁在节点之间移动时，概率出现差别，离终点越近的节点被蚂蚁选择的可能性更大，这种做法使得算法运行更加高效地、更快速地获得最优解。

### 2.2 精英蚂蚁体系

蚂蚁在节点之间进行移动时，由于开始搜索时路径经过蚂蚁数量较少，留下来的信息素特别低，如果这是后面蚂蚁朝着一条不是最优路线的节点移动时，这条非最优路径的信息素浓度就变大，在正反馈的作用之下，后续蚂蚁将会更多的经过这个路线，这时就会出现非最优解。面对这一问题，本文做出精英蚂蚁体系，对完成搜索的蚂蚁按照其走过路线的长度排序，对走过路线最小的蚂蚁所带有的信息素进行更新，增大其所带有的信息素量，而这种能够取得最短路径的蚂蚁被看成为精英蚂蚁，越是经过的路线越短，更新后蚂蚁所携带的信息素的量就越高。更新信息素的公式为：

$$\tau_{ij}(t+n) = (1-p)\tau_{ij}(t) + \Delta\tau_{ij}(t, t+n) \quad (7)$$

精英蚂蚁信息素的增加量为：

$$\Delta\tau_{ij}(t, t+n) = \sum_{k=1}^m \Delta\tau_{i,j}^k(t, t+n) \quad (8)$$

并对节点信息素进行更新为：

$$\Delta\tau_{ij}(t, t+n) = \begin{cases} (m-k) \frac{Q}{L_k}, & \text{若第 } k \text{ 只蚂蚁从 } i \text{ 到 } j \\ 0, & else \end{cases} \quad (9)$$

蚁群搜索初始，蚂蚁在两节点  $i$  和  $j$  之间进行移动时，由于开始搜索时路径经过蚂蚁数量较少，留下来的信息素特别低，如果这是后面蚂蚁朝着一条不是最优路线的节点移动时，这条非最优路径的信息素浓度就变大，在正反馈的作用之下，后续蚂蚁将会更多地经过这个路线，这时就会出现非最优解。而加入了精英蚂蚁体系，对完成搜索的蚂蚁按照其走过路线的长度排序，走过路线最短的蚂蚁被看作为精英蚂蚁，对精英蚂蚁所带有的信息素进行更新，增大它们所带有的信息素量，越是经过的路线越短，更新后蚂蚁所携带的信息素的量就越高，这种搜索方式将会极大地提高算法运算速度，快速得到算法的最优解。

### 2.3 更新信息素浓度

运用蚁群算法求解时，每只蚂蚁会根据每条路径信息素浓度的正反馈来决定它们的移动路线，信息素的改变将会直接影响整个算法的结果，而每条路径上信息素的浓度与它们的挥发系数有着直接的联系。当信息素的挥发系数大时，每条路径上的信息素浓度低，此时算法能够迅速收敛，但路径上信息素消失速度太快，无法求得解，当信息素挥发系数小时，每条路径留下的信息素浓度较大，这样会陷入局部最优解的误区。为了能够平衡这个关系，对信息素挥发系数进行改造，使其服从正太分布，即蚁群在刚

开始搜索时, 挥发系数较小, 此时能够在路径上留下更多的信息素, 便于提高初始搜索效率, 蚂蚁能够通过正反馈的指引来获得比较强寻优能力; 随着时间的增加, 信息素挥发系数越来越大, 这时路径上所留下来的信息素能够快速消失, 这样就可以增大最优解的个数, 很好地避免了出现局部最优解; 当搜索快要结束时, 挥发系数降低, 这样路径上的信息素浓度就会增加, 能够更好地引导后续的蚁群, 起到正反馈作用。

信息素挥发系数满足正态分布如下:

$$\rho(x = k) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (10)$$

其中:  $k$  为精英蚂蚁的序号。

相较于传统蚁群算法, 利用服从正态分布的信息素挥发系数能够更好地利用取值的不同大小来实现改进蚁群算法求得更多最优解的目的。利用正态分布的特性, 在蚂蚁搜索最优解的初始阶段, 信息素挥发系数小, 蚂蚁在经过每条路径时便可以留下更多的信息素, 对后面蚂蚁做到了更好的引导, 随着搜索的不断进行, 挥发系数慢慢增大, 蚂蚁在通过每条路径上的信息素将会快速挥发, 这样蚂蚁就能够达到更多的节点, 有效地避免局部最优的现象, 能够增大最终获得最优解的数量, 提高了系统的性能, 随着蚂蚁搜索的不断进行, 信息素的挥发系数慢慢减小, 蚂蚁经过的每条路径上的信息素浓度就不断增大, 便于蚂蚁能够到达目标, 整个路径寻优就此完成。由此可见, 用于服从正态分布的挥发系数具有传统固定挥发系统所不具备的多种优势, 这种改进的蚁群算法更能够快速、高效地求出更多的最优解, 进一步增加了蚁群算法的鲁棒性。

#### 2.4 改进蚁群算法的路径规划步骤

通过对蚁群算法的改进, 路径规划过程也发生了相应的改进, 具体步骤如下:

Step1: 数学参数初始化。给定路径的起点坐标和终点坐标, 设置蚁群的个数、信息启发因子、最大迭代的次数、信息素挥发系数等一系列参数

Step2: 构建环境模型。运用栅格法将空间模型抽象处理, 根据环境信息完成环境模型的构建。

Step3: 信息参数初始化。将蚁群的排除表及模型的长度等信息进行初始化, 蚂蚁从路径的起点坐标出开始向前行进, 根据蚂蚁从一个节点到达另一个节点的概率公式来进行寻找, 每次到达一处节点都要进行记录并将此处节点加入排除表, 当环境模型中的所有节点都出现在排除表时, 蚁群就完成了整个寻优过程。

Step4: 信息素更新。蚁群没完成一次迭代, 都要对信息素浓度等参数进行更新。

Step5: 判断是否是局部最优解。若是, 则将信息素挥发系数按照正态分布进行改进; 否则, 继续进行迭代。

Step6: 完成遍历。将蚁群寻优已完成的次数与最大迭代次数比较, 当已完成次数小于最大迭代次数时, 继续完成下一次迭代; 反之, 迭代完成, 蚁群结束下一步寻优。

Step7: 输出。将蚁群算法得到的最优路径保存。

改进的蚁群算法处理路径规划问题的流程如图 3 所示。

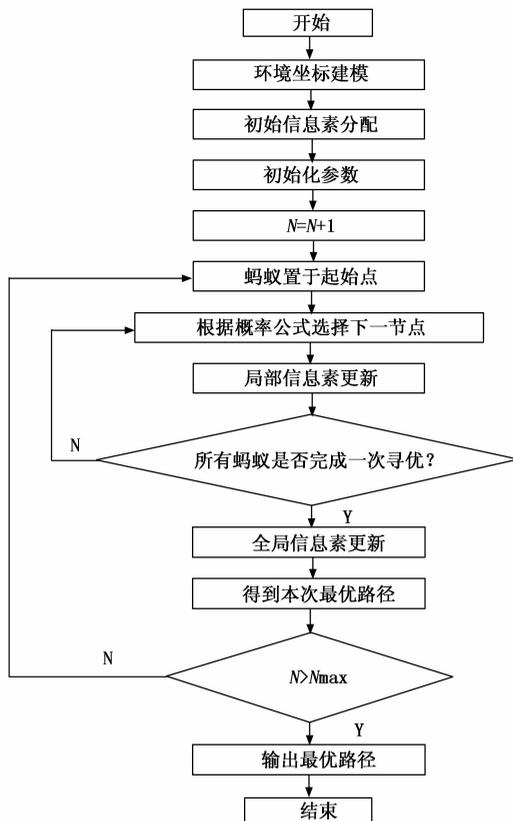


图 3 改进后的蚁群算法寻优流程图

算法在处理路径规划问题上的编程思路如下:

Begin

建立环境模型的 01 矩阵。(0 表示可行进空间, 1 表示障碍物空间)

参数初始化, 将  $m$  只蚂蚁放到起点上。

loop

for  $k=1$  to  $m$  do//从第 1 只蚂蚁开始, 依次置于起点

按概率计算公式(6)计算选择下一个节点  $j$ ;

按公式(10)更新节点间的信息素浓度;

if 节点  $j$  是终点  $D$  then

全局信息素更新;

记录起点到终点  $D$  之间的距离和路径;

else 根据公式(6)选择下一节点

更新全局信息素

if  $N \geq$  then

输出最短路径和距离

else exit loop

end

### 3 仿真实验

为了验证改良过后的蚁群遗传算法的全局路径规划的性能, 将水下环境空间栅格划分为  $20 \times 20$  的栅格坐标系, 对传统蚁群算法和改进后的蚁群算法进行仿真实验, 此次仿真计算机为 Intel (R) .Core (TM) i5-7300HQ 的处理器, Win-

dows10 家庭中文版操作系统，仿真软件为 Matlab2016b。

仿真的实验参数如表 1 所示。

表 1 实验参数值

参数	信息素常量	信息启发因子	期望启发因子	信息素挥发系数	最大迭代次数
传统算法数值	100	1	3	0.1	200
改进算法数值	100	1	3	(0.2,0.9)	200

仿真结果如图 4、图 5 所示。

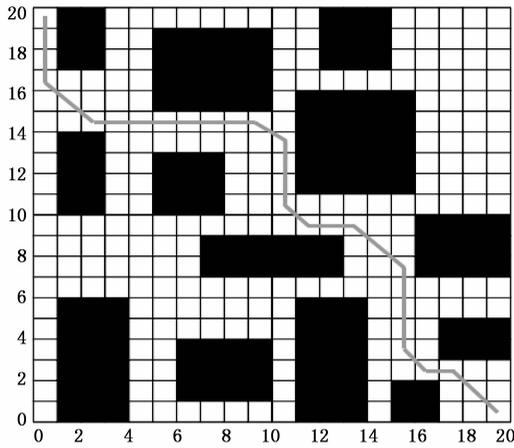


图 4 传统蚁群算法路径图

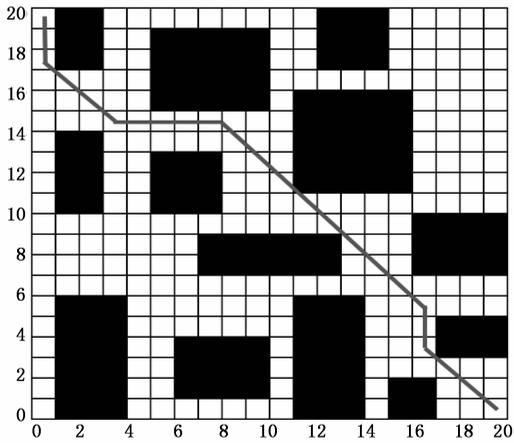


图 5 改进后的蚁群算法路径图

表 2 两种算法数据对比表

算法	平均迭代次数	最大迭代次数	最优路径长度	平均路径长度
常规蚁群算法	43	61	26.373 9	27.79
改进蚁群算法	24	34	25.384 8	25.61

根据图 4 可以看出，传统的蚁群算法在解决路径规划问题上有着不错的优势，蚁群从起点开始能够有效躲避路径上的障碍物到达终点，但在最优解上并不完美。而从图 5 可以看到，改进后的算法可以有效地找到最短距离，得到的路径更短。由表 2 的数据可看出，改进后的蚁群算法平

均迭代次数为 24 次，比传统的蚁群算法的 43 次更少，而且得到的最短路径也比传统蚁群算法短了接近一个单位长度，算法响应更加快速。

改良过的蚁群遗传算法在最短路径长度、迭代次数及运算时间方面都优于传统的蚁群算法，有效提升了遗传算法的收敛速度及管理效率。从以上最优途径对比可见，传统式蚁群遗传算法在搜索初期陷于了局部最优，纵然终究还找到了一条最优途径，但是途径品质不及改良后的蚁群遗传算法。

### 4 结束语

本文在传统的蚁群遗传算法分析模型剖析的基础上，给出一种改良的蚁群算法，加入目标引导素，将几个因素综合考虑，减少了随机性，构建多参数的优化分析模型。仿真试验结果显示，改良的蚁群遗传算法具备良好的鲁棒性，且其收敛到最优解的速度还较快，求解成本低。在下一步的研究中，经不断改进添加引导素和更新信息素浓度等参数的值，进一步提高算法的收敛速度，保证水下无人机的环境适应能力。

### 参考文献：

[1] 王钦钊, 程金勇, 李小龙. 复杂环境下机器人路径规划方法研究 [J]. 计算机仿真, 2017 (10): 296-300.

[2] 马国兵. 灾后矿井环境探测机器人的路径规划研究 [D]. 中国矿业大学, 2019.

[3] 张爽爽. 环境监测的无人机群路径规划方法研究 [D]. 哈尔滨: 哈尔滨理工大学, 2019.

[4] 李晓静, 余东满. 改进蚁群算法在果蔬采摘机器人路径规划中的应用 [J]. 江苏农业科学, 2018 (23): 253-258.

[5] 庄丽阳, 陈树林, 朱龙彪, 等. 基于改进蚁群算法的农用喷药机器人路径规划 [J]. 机床与液压, 2018 (21): 15-19.

[6] 郭秀娟, 张坤鹏. 基于蚁群混合遗传算法的组卷问题研究 [J]. 吉林建筑大学学报, 2017 (4): 79-83.

[7] 梁嘉俊, 曾 碧, 何元烈. 基于改进势场栅格法的清洁机器人路径规划算法研究 [J]. 广东工业大学学报, 2016 (4): 30-36.

[8] 董 晔, 吴丽娟. 基于混合人工势场-蚁群算法的机器人避障 [J]. 辽宁科技大学学报, 2015 (3): 212-216.

[9] 吴 建, 李 康, 庞 宇, 等. 基于差分阈值与模板匹配的心电 R 波提取算法 [J]. 重庆邮电大学学报 (自然科学版), 2015 (3): 372-376.

[10] 王 芳, 李昆鹏, 袁明新. 一种人工势场导向的蚁群路径规划算法 [J]. 计算机科学, 2014 (s2): 47-50.

[11] Li H, Yang S X, Seto M L. Neural-network-based path planning for a multirobot system with moving obstacles [J]. IEEE Transactions on Systems, Man & Cybernetics Part C, 2009, 39 (4): 410-419.

[12] Yuan X B, Yang S X. Virtual assembly with biologically inspired intelligence [J]. IEEE Transactions on Systems, Man & Cybernetics Part C Applications & Reviews, 2003, 33 (2): 159-167.