

DOI:10.3969/j.issn.1671-024x.2026.02.011

## 基于二分思想的迷宫机器人智能搜索算法

袁臣虎<sup>1</sup>, 张习文<sup>2</sup>, 高超<sup>1</sup>, 谭子檬<sup>1</sup>

(1. 天津工业大学 电子与信息工程学院, 天津 300387; 2. 天津工业大学 电气工程学院, 天津 300387)

**摘要:**为解决迷宫机器人在复杂迷宫进行无效区域搜索导致的低效率问题,提出在智能迷宫搜索算法基础上加入二分查找方法对迷宫进行高效解算。利用迷宫单元顺序排列特征构建一个有序的迷宫坐标二维数组,设定初始边界元素集合并将机器人的搜索路径更新到边界集合;当边界将迷宫划分为2个部分时,对2个区域进行目标点查找,并据此划分有效和无效区域;机器人后续搜索将无效区域剔除,仅对有效区域进行智能搜索,迭代此过程实现迷宫目标的快速到达。将二分查找法与智能算法A和Flood-central算法结合,生成100幅迷宫地图,利用机器人开发研究平台(ROS)对机器人和迷宫进行建模,搭建交互环境进行两组算法应用测试。结果表明:在迷宫搜索路程分别超过100格、200格、300格的情况下,相较于单独使用A算法和Flood central算法,结合二分查找法后搜索效率平均提升了26%、28%与41%,证实了二分查找法与智能搜索算法相结合的策略可有效提升机器人在复杂迷宫中的搜索效率。

**关键词:** 迷宫机器人;二分查找法;智能搜索算法;迷宫搜索效率;机器人开发研究平台(ROS)

中图分类号: TP242.6; TP301.6

文献标志码: A

文章编号: 1671-024X(2026)02-0086-08

### Intelligent search algorithm for maze robots based on binary search concept

Yuan Chenhu<sup>1</sup>, Zhang Xiwen<sup>2</sup>, Gao Chao<sup>1</sup>, Tan Zimeng<sup>1</sup>

(1. School of Electronics and Information Engineering, Tiangong University, Tianjin 300387, China; 2. School of Electrical Engineering, Tiangong University, Tianjin 300387, China)

**Abstract:** In order to solve the low-efficiency problem caused by maze robot searching invalid area in complex maze, a binary search method is proposed to solve the maze efficiently based on intelligent maze search algorithm. An ordered two-dimensional array of maze coordinates is constructed by using the sequential arrangement characteristics of maze units, and the initial set of boundary elements is set and the search path of the robot is updated to the boundary set. When the boundary divides the maze into two parts, the target points of the two regions are searched, and the effective and invalid regions are divided accordingly. The robot will eliminate the invalid area in the subsequent search, and only carry out intelligent search for the effective area. This process is iterated to achieve the rapid arrival of the maze target. The binary search method is combined with intelligent algorithm A and Flood-central algorithm to generate 100 maze maps. The robot and maze are modeled using the robot development and research platform (ROS), and the interactive environment is built to test the application of two groups of algorithms. The test results show that: when the maze search distance is more than 100 grids, 200 grids, and 300 grids respectively, compared with the use of A algorithm and Flood-central algorithm alone, the search efficiency combined with binary search method is improved by 26%, 28%, and 41% on average, which proves that the strategy of combining binary search method with intelligent search algorithm can effectively improve the search efficiency of robot in complex maze.

**Key words:** maze robot; binary search method; intelligent search algorithm; maze search efficiency; Robot Operating System (ROS)

智能决策与控制技术的迅速发展,极大提升了机器人应对复杂实际问题的能力。轮式移动机器人作为

机器人的典型代表,因其操控灵活、成本可控、扩展性强等特点,广泛应用于工厂自动化、智慧服务、搜救等

收稿日期: 2024-03-25

基金项目: 国家自然科学基金面上项目(62072335)

通信作者: 袁臣虎(1979—),男,博士,副教授,主要研究方向为机器人运动控制与路径规划。E-mail: yuanchenhu@tiangong.edu.cn

场景。迷宫机器人作为轮式移动机器人的一种,其在未知复杂迷宫环境下的快速搜索导航对机器人导航研究有重要借鉴意义<sup>[1-6]</sup>。本文将机器人面对的环境抽象为迷宫,对迷宫机器人搜索算法展开研究,期望实现更效率的机器人搜索。

文献[7-8]提出基于强化学习的迷宫机器人搜索方式,实现了机器人在未知迷宫中的智能决策和快速搜索。但概率大模型算法的实现技术复杂,软件硬件维护成本高昂,目前不利于各类常见场景中的大规模推广使用。文献[9-13]在前人经验基础上,对迷宫机器人智能搜索算法进行改进。基于静态网路实现迷宫搜索和路径规划的洪水算法和 Dijkstra 算法,大大提高了迷宫搜索成功率。基于启发式搜索 A\* 算法的机器人迷宫搜索算法与中心指向搜索的 Flood-central 算法,提高了机器人迷宫搜索的效率<sup>[14-15]</sup>。基于智能仿生算法如遗传算法和模糊控制算法等的迷宫智能搜索算法,提高了迷宫解算效率和解的质量<sup>[16-18]</sup>。以上智能搜索算法仅强调机器人在迷宫中搜索方向上的改变,没有利用已有搜索信息对机器人需要搜索的迷宫区域进行分析简化,在搜索地形复杂迷宫时机器人大量遍历无效区域导致搜索效率降低。

将机器人迷宫搜索过程类比为数组查找,其与数组查找不同之处在于:迷宫是二维平面,有 2 个维度 4 个方向供选择查找;查找元素必须是连续曲线且查找方向受迷宫地形影响。已有的迷宫智能搜索算法都在调整搜索权重、优化查找方向,本质上仍是逐格搜索查找,类似于顺序查找。鉴于此,本文提出了将二分查找与智能搜索算法相结合的迷宫搜索策略,将迷宫格作为数组元素建立二维迷宫数组,利用机器人迷宫搜索产生的连续曲线将迷宫数组划分为有效集和无效集,后续搜索过程将放弃对无效集进行查找,快速提高迷宫机器人的搜索效率。机器人查找方向受复杂迷宫地形桎梏,查找曲线不能完美满足数组二分查找条件;同时迷宫搜索以快速到达目标点为目的,错误选择会浪费大量时间,优先选择将迷宫数组二分的查找方向会显著降低搜索效率。因此以智能搜索算法为基础策略,保障搜索完成度与基本效率;以迷宫数组二分查找为辅助策略,快速缩小搜索范围。

## 1 迷宫搜索二分查找法的理论基础

### 1.1 二分查找法简介

二分查找,也被称为折半查找(binary search),是一种高效的查找技术。它要求数据集以顺序存储的方

式组织,并且数据元素需要根据关键字进行排序<sup>[19-21]</sup>。该技术的实现过程如图 1 所示,求解一维封闭且连续区间内的特定值(9),取中值(5)将区间进行二分,关键字比较后对包含目标值的子区间进行二分操作,迭代此过程直至精确定位到目标值。

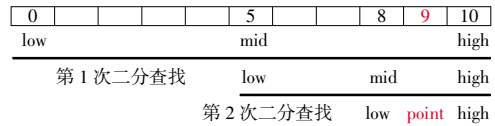


图 1 一维刻度上二分法求取特定值

Fig.1 Dichotomy search process on one-dimensional scale

### 1.2 二维平面上二分查找法的应用

将二分查找方法应用于二维平面中,过程如图 2 所示。

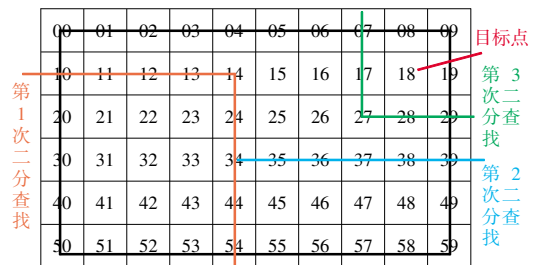


图 2 二维平面上二分法求取特定值

Fig.2 Dichotomy search process on two-dimensional plane

令二维平面中的关键字按照升序进行排列,求取目标点关键字(18)。取坐标连续变化的曲线(First)将整个区域不规则二分;将关键字与其中一个区域所有元素进行比较;对包含目标位置的子区域继续进行不规则二分;迭代上述过程直至精确定位到目标点。

### 1.3 迷宫二维数组的设定

在迷宫二维平面,建立迷宫坐标系,将水平轴定义为 X 轴,垂直轴定义为 Y 轴。构造一个升序排列的坐标二维数组来满足二分查找法的数据结构要求,如图 3 所示,迷宫格坐标值为二分法查找的关键字。迷宫的坐标二维数组 M 可以表示为式(1):

$$M = \begin{bmatrix} m_{1,1} & \cdots & m_{1,16} \\ \vdots & & \vdots \\ m_{16,1} & \cdots & m_{16,16} \end{bmatrix} \quad (1)$$

### 1.4 迷宫二分曲线与二分区域的设定

图 3 中,将迷宫坐标系中 X = 1、X = 16、Y = 1、Y = 16 的迷宫格(灰色背景标识的数组元素)设定初始边界,机器人搜索路径上的迷宫格(浅绿色标识的数组元素)补充为新边界。边界元素包含的连续曲线将迷宫不规则二分为 2 块新的封闭区域(橙色与蓝色标识的数组元素集合)。

(1,16)	(2,16)	(3,16)	(4,16)	(5,16)	(6,16)	(7,16)	(8,16)	(9,16)	(10,16)	(11,16)	(12,16)	(13,16)	(14,16)	(15,16)	(16,16)
(1,15)	(2,15)	(3,15)	(4,15)	(5,15)	(6,15)	(7,15)	(8,15)	(9,15)	(10,15)	(11,15)	(12,15)	(13,15)	(14,15)	(15,15)	(16,15)
(1,14)	(2,14)	(3,14)	(4,14)	(5,14)	(6,14)	(7,14)	(8,14)	(9,14)	(10,14)	(11,14)	(12,14)	(13,14)	(14,14)	(15,14)	(16,14)
(1,13)	(2,13)	(3,13)	(4,13)	(5,13)	(6,13)	(7,13)	(8,13)	(9,13)	(10,13)	(11,13)	(12,13)	(13,13)	(14,13)	(15,13)	(16,13)
(1,12)	(2,12)	(3,12)	(4,12)	(5,12)	(6,12)	(7,12)	(8,12)	(9,12)	(10,12)	(11,12)	(12,12)	(13,12)	(14,12)	(15,12)	(16,12)
(1,11)	(2,11)	(3,11)	(4,11)	(5,11)	(6,11)	(7,11)	(8,11)	(9,11)	(10,11)	(11,11)	(12,11)	(13,11)	(14,11)	(15,11)	(16,11)
(1,10)	(2,10)	(3,10)	(4,10)	(5,10)	(6,10)	(7,10)	(8,10)	(9,10)	(10,10)	(11,10)	(12,10)	(13,10)	(14,10)	(15,10)	(16,10)
(1,9)	(2,9)	(3,9)	(4,9)	(5,9)	(6,9)	(7,9)	(8,9)	(9,9)	(10,9)	(11,9)	(12,9)	(13,9)	(14,9)	(15,9)	(16,9)
(1,8)	(2,8)	(3,8)	(4,8)	(5,8)	(6,8)	(7,8)	(8,8)	(9,8)	(10,8)	(11,8)	(12,8)	(13,8)	(14,8)	(15,8)	(16,8)
(1,7)	(2,7)	(3,7)	(4,7)	(5,7)	(6,7)	(7,7)	(8,7)	(9,7)	(10,7)	(11,7)	(12,7)	(13,7)	(14,7)	(15,7)	(16,7)
(1,6)	(2,6)	(3,6)	(4,6)	(5,6)	(6,6)	(7,6)	(8,6)	(9,6)	(10,6)	(11,6)	(12,6)	(13,6)	(14,6)	(15,6)	(16,6)
(1,5)	(2,5)	(3,5)	(4,5)	(5,5)	(6,5)	(7,5)	(8,5)	(9,5)	(10,5)	(11,5)	(12,5)	(13,5)	(14,5)	(15,5)	(16,5)
(1,4)	(2,4)	(3,4)	(4,4)	(5,4)	(6,4)	(7,4)	(8,4)	(9,4)	(10,4)	(11,4)	(12,4)	(13,4)	(14,4)	(15,4)	(16,4)
(1,3)	(2,3)	(3,3)	(4,3)	(5,3)	(6,3)	(7,3)	(8,3)	(9,3)	(10,3)	(11,3)	(12,3)	(13,3)	(14,3)	(15,3)	(16,3)
(1,2)	(2,2)	(3,2)	(4,2)	(5,2)	(6,2)	(7,2)	(8,2)	(9,2)	(10,2)	(11,2)	(12,2)	(13,2)	(14,2)	(15,2)	(16,2)
(1,1)	(2,1)	(3,1)	(4,1)	(5,1)	(6,1)	(7,1)	(8,1)	(9,1)	(10,1)	(11,1)	(12,1)	(13,1)	(14,1)	(15,1)	(16,1)

图 3 迷宫二维数组与边界二分图示

Fig.3 Binary diagram of maze 2D array and boundary

1.5 二分区域取舍的判定依据

图 3 中,目标点为(6,9),边界数组元素将迷宫分割成 2 个封闭区域,由橙色与蓝色背景进行标识。

目标点未被访问且位于蓝色标识区域,需要舍弃橙色区域。若橙色区域中的任一迷宫格与目标点无通路,该区域无搜索必要;或即使存在通路,该通路与机器人的搜索路径相交,交点到目标点的路径长度必定小于被舍弃区域内任意点到目标点的路径长度,从路径最短角度考虑被舍弃区域无需进一步搜索。

1.6 二分查找的数据集合定义

本文定义了以下数据集合,用于二分法中关键字的查找和二分计算:

(1) 集合  $S$  代表尚未搜索的迷宫格坐标集合,初始状态为整个迷宫  $M$ 。每次机器人搜索一格,就从集合  $S$  中剔除对应坐标元素。每进行一次二分查找操作,会将无效区域包含的坐标元素一次性从集合  $S$  中剔除。当机器人搜索到目标点时,集合  $S$  将被清零。

(2) 集合  $O$  代表已搜索区域的迷宫格坐标集合。每搜索一格,就会向集合  $O$  添加相应的坐标元素。集合  $O$  在迷宫数组  $M$  中表现为一条连续的曲线,当这条曲线将迷宫数组  $M$  划分为 2 个子数组时,机器人进行一次二分查找操作。

(3) 集合  $N_1$  和  $N_2$  用于存储被二分的 2 个区域的迷宫元素坐标。可以对迷宫数组  $M$  逐行扫描和逐列扫描,计算扫描交集以求出二分区域坐标集,本次二分查找操作后被清零。

(4) 集合  $W$  代表无效搜索区域的迷宫格坐标集

合。每次进行二分查找操作后,都会将舍弃的二分区域所包含的坐标元素添加到集合  $W$ 。

1.7 二分查找的数据运算分析

在迷宫搜索任务的初始状态下,这些集合满足式(2);

$$\begin{cases} S = M \\ O = N_1 = N_2 = W = \emptyset \end{cases} \quad (2)$$

当机器人成功到达终点时,即目标值查找成功,这些集合满足式(3):

$$\begin{cases} O \cup W \subseteq M \\ S = N_1 = N_2 = \emptyset \end{cases} \quad (3)$$

在搜索过程中,如果迷宫边界包含的连续曲线未将迷宫分为 2 部分,或者经过一次二分查找处理后还未产生新的二分区域,称该状态为“二分插入状态”。在这种状态下,迷宫数组集合( $M$ )、待搜索集合( $S$ )、已搜索集合( $O$ )、二分区域集合( $N_1$ )、( $N_2$ )以及无效搜索区域集合( $W$ )满足关系式(4):

$$\begin{cases} O \cup W \cup S = M \\ N_1 = N_2 = \emptyset \end{cases} \quad (4)$$

如果已搜索集合( $O$ )中的坐标点在迷宫数组( $M$ )上形成的连续曲线将迷宫分割成了 2 个封闭区域,则进入“目标查找状态”。在这个状态下,执行以下步骤进行目标关键字查找:

(1) 计算二分封闭区域的坐标点以形成集合( $N_1$ )和( $N_2$ );

(2) 将目标坐标值与 2 个集合中的坐标元素挨个进行比对,取不包含目标位置的那个集合(假设为( $N_1$ ))与待搜索集合( $S$ )的交集,将其添加到无效搜索

区域集合( $W$ )中,并从待搜索集合( $S$ )中移除;

(3) 将集合( $N_1$ )和( $N_2$ )清零。

上述 3 个步骤中的迷宫坐标集合满足关系式(5):

$$\begin{cases} N_1 \cup N_2 = M \\ W = (N_1 \cap S) \cup W \\ S = S - (N_1 \cap S) \\ N_1 = N_2 = \emptyset \end{cases} \quad (5)$$

## 2 二分查找法与智能算法的结合

### 2.1 算法结合思路

图 4 为使用二分查找与智能搜索算法结合进行迷宫搜索的流程。

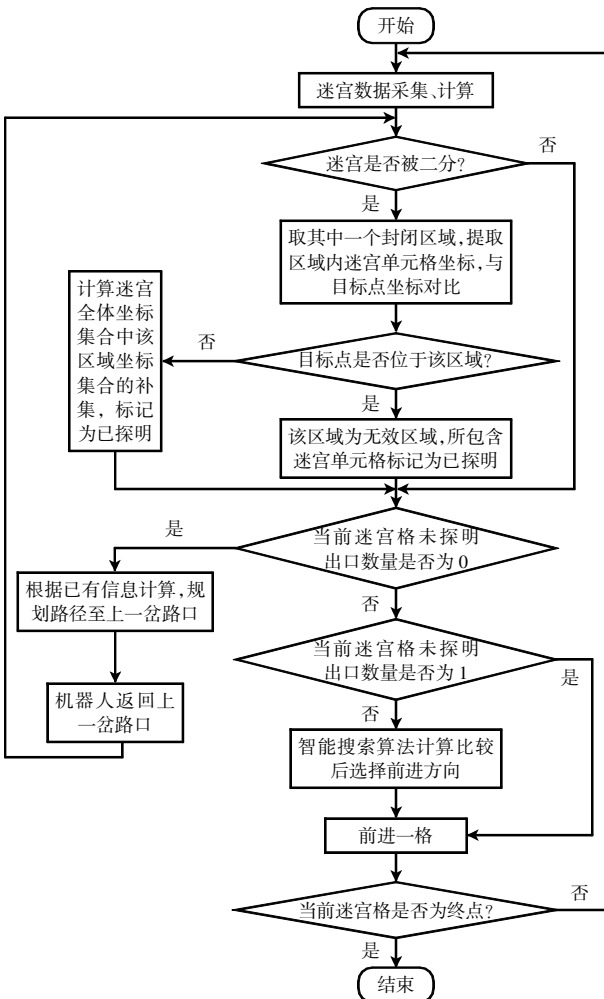


图 4 算法融合策略迷宫搜索流程

Fig.4 Maze search flow of fusion algorithm

设定二分查找法实施的数据结构,将其与所结合的智能搜索算法中相应数据作关联;仍使用原智能搜索算法的逻辑指导机器人搜索导航,保障迷宫搜索的

成功率和基本搜索效率;使用二分查找方式根据搜索信息对迷宫数组进行切分计算,甄别无效搜索区域,快速缩小待搜索范围,进一步提高迷宫机器人的搜索效率。

### 2.2 实体机器人迷宫搜索的时间复杂度分析

实体机器人迷宫搜索时间可以用来表示搜索效率的高低,迷宫搜索时间不仅涉及数据的计算处理时间,也涉及机器人在迷宫中的行进时间。机器人搜索迷宫过程的时间复杂度公式可由式(6)表示:

$$O(n) = T_1(n) + T_2(n) \quad (6)$$

式中: $T_1(n)$ 为机器人迷宫搜索行进时间; $T_2(n)$ 为算法计算时间; $n$ 为需要搜索的迷宫格数。当二分查找次数为 0 时,机器人使用融合算法搜索迷宫的路线与使用智能搜索算法的路线一致,使用融合算法搜索迷宫的时间复杂度公式总结为式(7):

$$O(n-w) = T_1(n-w) + T_2(n-w) \quad (7)$$

式中: $w$ 为二分查找法甄别出无效区域中被机器人主动绕行的格数。参考目前最普遍的 80 MHz 主频微处理器执行 1 条指令仅耗时几纳秒,算法计算时间  $T_2(n)$  相比机器人行进时间  $T_1(n)$  可忽略不计。迷宫组成格数由 2 格逐渐累加,机器人在迷宫中搜索的可行进最大路程统计归纳如表 1 所示。

表 1 不同格数迷宫中机器人的最大搜索路程

Tab.1 Maximum search distance of robot in maze with different lattice numbers

组成迷宫格数	最大路程长度/格
2	1
3	3
4	6
5	10
6	15
⋮	⋮
$n$	$n(n-1)/2$

迷宫搜索机器人的行进时间复杂度由式(8)表示,使用算法融合搜索策略的机器人的行进时间复杂度由式(9)表示:

$$T_1(n) = t_1 \frac{n(n-1)}{2} + t_2 m \quad (8)$$

$$\begin{cases} T_1(n-w) = t_1 \frac{(n-w)(n-w-1)}{2} + t_2 m \\ 0 \leq w < n \end{cases} \quad (9)$$

式中: $t_1$ 为机器人直行 1 格耗时; $t_2$ 为机器人转弯 1 次耗时; $t_1$ 与  $t_2$ 为常量; $m$ 为最大转弯次数。

使用基于二分查找的融合算法进行迷宫搜索和

仅使用智能搜索算法进行迷宫搜索的时间差值可表示为机器人迷宫搜索行进时间差值与算法计算时间差值之和,如式(10)所示:

$$\Delta O = O(n) - O(n - w) = \Delta T_1 + \Delta T_2 \quad (10)$$

机器人转弯动作分布在行进过程中。与仅使用智能搜索算法进行迷宫查找相比,基于二分的融合算法缩小机器人迷宫的待搜索范围,在减少行进路程的同时同步减少转弯次数,行进时间差值由式(11)表示:

$$\Delta T_1 = \begin{cases} 0, w = 0 \\ t_1 \frac{(w^2 + nw + w)}{2} + t_2 \Delta m, w > 0 \end{cases} \quad (11)$$

使用融合算法和仅使用智能搜索算法进行迷宫搜索的时间差值表示为式(12):

$$\Delta O = \begin{cases} \Delta T_2, w = 0 \\ t_1 \frac{(w^2 + nw + w)}{2} + t_2 \Delta m + \Delta T_2, w > 0 \end{cases} \quad (12)$$

$w = 0$  时,机器人使用融合算法进行迷宫搜索所花费时间,与仅使用智能搜索算法进行迷宫搜索的时间大致相同; $w > 0$  时,机器人使用融合算法进行迷宫搜索所花费时间,应小于使用智能搜索算法进行迷宫搜索的时间。

### 3 实验验证

#### 3.1 算法验证平台

Robot Operating System(ROS)是一个面向机器人开发的开源操作系统,提供丰富的功能和庞大的生态用于机器人开发和技术研究<sup>[22-23]</sup>。利用 ROS 提供的功能对迷宫机器人的传感器、执行器、控制器等部件进行抽象处理,在软件中模拟出实体硬件的功能,可实现与实体机器人一样的运动控制效果,如图 5 所示。

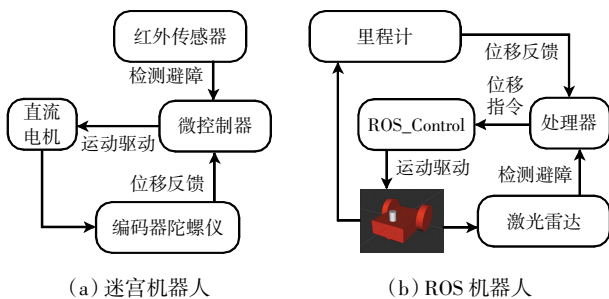


图 5 迷宫机器人与 ROS 机器人运动控制

Fig.5 Motion control of maze robots and ROS robots

通过 URDF(unified robot description format)文件实现机器人和迷宫 3D 建模,在开源机器人仿真器

(gazebo)中添加机器人和实体迷宫的三维物理模型,复现机器人与环境的交互过程。基于 ROS 搭建机器人迷宫搜索实验平台,可用于迷宫搜索、路径规划、轨迹优化、机器人控制等算法验证,并将算法直接移植到实体机器人中,实验平台如图 6 所示。

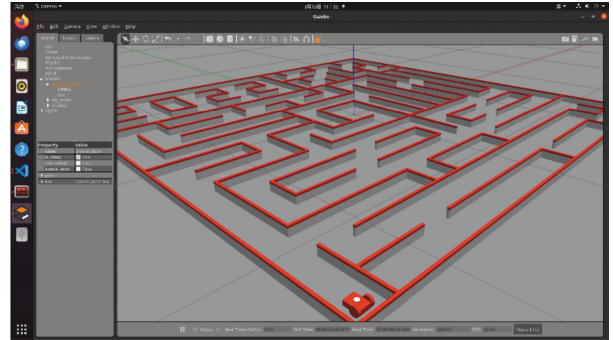


图 6 基于 ROS 的机器人走迷宫实验平台

Fig.6 Maze search experimental platform based on ROS

#### 3.2 算法对比分析

为了验证所提出算法的有效性,选择 A\* 算法、Flood-central 2 种迷宫搜索算法与二分查找法结合,将融合智能搜索算法与原算法作对比,进行各自 50 组迷宫搜索实验。记录机器人搜索路线并输出迷宫搜索用时和步长,机器人前行 1 格步长加 1,搜索时长默认 3 min,超时搜索失败输出空白。基于二分查找的智能搜索算法与原算法对比实验的输出结果如表 2 和图 7 所示。

由实验数据可知,结合二分查找法的智能搜索算法相较于原智能搜索算法,在 40%以上的迷宫搜索实验中提高了搜索效率,说明二分查找方法对提高机器人迷宫搜索是有效的。

物理仿真平台界面渲染的快慢、卡顿对实际搜索时间有一定影响,而机器人搜索行进步长的数值不受影响,机器人搜索速度为定值,迷宫搜索时间与行进步长成正比,实验数据如图 8 所示,可用机器人搜索行进步长来衡量搜索效率的高低。

以机器人搜索步长 150 为分界线,将搜索效率提高和未提高的实验数据进行对比,如图 9 所示。由图 9 可以看出,在迷宫地形越复杂、搜索行进步长越长的迷宫中,更能满足二分查找的前置条件,使机器人迷宫搜索效率得到提高。

在 2 组共 41 例搜索效率提高的实验中,平均搜索效率提高的程度与机器人迷宫搜索行进步长关系如表 3 所示。由表 3 可以看出,搜索行进步长越大,迷宫越复杂,使用基于二分查找的智能搜索算法的搜索效率提高平均值越大。

表 2 对比实验数据

Tab.2 Comparative experimental data

迷宫	A* 算法		基于二分查找 A*		迷宫	Flood-central		基于二分查找 Flood-central	
	搜索时间/s	搜索步数	搜索时间/s	搜索步数		搜索时间/s	搜索步数	搜索时间/s	搜索步数
demo1	50.591 3	118	36.568 4	96	demo51	—	—	—	—
demo2	93.354 0	184	65.426 0	134	demo52	151.871 7	314	111.423 1	226
demo3	14.371 3	32	14.092 1	32	demo53	90.648 3	181	37.265 1	79
demo4	63.121 3	169	62.356 5	169	demo54	52.526 9	118	53.057 5	118
demo5	27.481 3	69	27.143 1	69	demo55	55.543 4	118	54.633 3	118
demo6	22.518 3	52	20.597 6	46	demo56	40.486 2	90	39.762 6	88
demo7	85.168 5	171	86.802 0	171	demo57	42.186 6	90	41.216 5	88
demo8	95.263 3	199	81.743 4	167	demo58	105.753 1	232	107.744 8	217
demo9	14.974 3	33	14.458 5	33	demo59	132.345 9	310	64.973 0	140
demo10	100.868 0	231	94.154 9	189	demo60	35.073 2	77	36.483 2	77
demo11	7.356 9	17	7.395 5	17	demo61	—	—	—	—
demo12	40.833 3	84	39.286 1	80	demo62	34.909 0	77	35.460 7	77
demo13	60.969 3	155	60.158 1	155	demo63	49.289 0	117	42.251 7	107
demo14	15.839 3	35	15.290 4	35	demo64	11.225 0	23	11.531 6	23
demo15	89.255 8	197	56.169 0	148	demo65	38.263 9	94	39.556 2	94
demo16	66.373 7	158	66.538 9	158	demo66	61.558 4	144	61.677 4	144
demo17	113.841 3	278	92.129 8	202	demo67	39.012 5	94	39.688 3	94
demo18	134.540 0	312	95.129 8	202	demo68	42.341 7	100	43.337 9	100
demo19	115.761 3	272	115.690 1	272	demo69	28.461 8	72	28.516 2	72
demo20	98.943 5	202	62.903 2	169	demo70	51.441 1	112	52.816 7	112
demo21	31.751 3	70	31.260 6	70	demo71	40.340 0	96	36.930 2	88
demo22	—	—	—	—	demo72	68.601 2	161	59.265 9	131
demo23	81.361 6	180	81.803 2	180	demo73	141.650 3	333	62.258 0	157
demo24	101.527 5	236	95.916 8	202	demo74	60.074 5	134	59.928 5	134
demo25	34.380 1	81	34.653 0	81	demo75	114.738 5	253	94.376 7	217
demo26	89.259 3	208	54.289 0	143	demo76	14.794 0	31	14.436 8	31
demo27	73.576 3	191	73.941 7	191	demo77	42.241 4	94	43.157 1	94
demo28	—	—	—	—	demo78	41.862 7	94	41.862 7	94
demo29	40.204 3	81	40.158 0	81	demo79	53.429 7	125	48.922 5	119
demo30	42.860 7	90	42.732 7	90	demo80	120.032 2	288	108.006 1	256
demo31	112.155 6	269	104.267 1	216	demo81	22.431 6	54	14.558 9	36
demo32	89.420 8	221	89.160 3	221	demo82	25.095 0	66	25.095 0	66
demo33	25.214 0	58	25.925 8	58	demo83	112.625 9	288	112.625 9	288
demo34	120.199 3	294	83.385 6	189	demo84	75.410 4	198	67.631 8	182
demo35	71.911 3	168	71.270 5	168	demo85	41.005 0	92	42.062 7	90
demo36	42.631 3	97	42.019 5	97	demo86	94.051 8	184	33.764 9	66
demo37	55.706 3	133	55.203 0	133	demo87	67.631 7	163	69.602 1	161
demo38	145.796 8	354	55.197 7	147	demo88	59.276 8	145	56.811 5	145
demo39	126.628 3	315	59.240 2	160	demo89	60.412 0	128	52.937 9	99
demo40	139.671 3	319	113.063 1	275	demo90	98.675 7	200	40.466 6	82
demo41	57.702 6	133	57.029 3	133	demo91	6.325 1	15	6.359 3	15
demo42	23.401 3	54	23.905 7	54	demo92	38.388 9	84	38.253 0	84
demo43	100.366 3	233	85.160 7	202	demo93	55.898 9	123	57.706 1	123
demo44	71.813 1	178	56.654 0	148	demo94	137.163 5	320	137.163 5	320
demo45	—	—	—	—	demo95	41.339 3	86	41.081 5	86
demo46	55.276 8	145	55.343 7	145	demo96	82.454 6	190	82.454 6	190
demo47	77.750 2	192	77.025 7	192	demo97	40.598 4	99	42.891 2	99
demo48	111.561 3	263	102.563 8	249	demo98	43.592 9	101	42.556 5	101
demo49	—	—	—	—	demo99	42.932 2	104	42.932 2	104
demo50	83.361 3	194	83.194 2	194	demo100	72.257 3	160	76.066 9	160

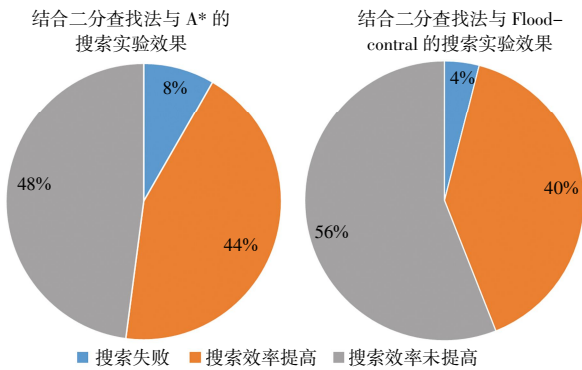


图 7 搜索算法统计与效果对比

Fig.7 Statistics and effect comparison of search algorithm

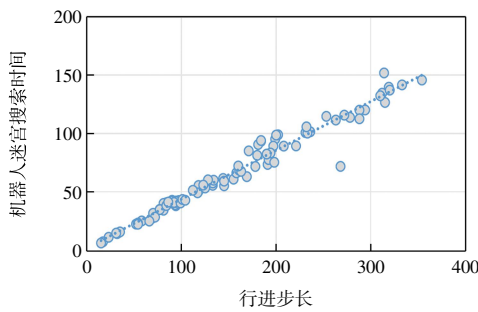


图 8 机器人迷宫搜索时长与行进步长关系

Fig.8 Relationship between search time and walking step size

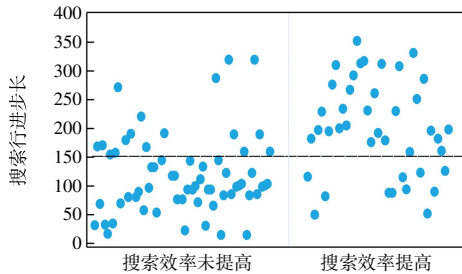


图 9 二分查找实现与迷宫复杂程度的关系

Fig.9 Relationship between binary search implementation and maze complexity

表 3 搜索效率提高程度与迷宫复杂度关系

Tab.3 Relationship between improvement degree search efficiency and maze complexity

迷宫搜索行进步长	搜索效率提高平均值 /%
>0	23
≥100	26
≥200	28
≥300	41

### 4 结 论

本文从机器人在迷宫中的搜索区域入手,提出将二分查找方法应用到迷宫二维模型,实现机器人对迷宫区域化的高效处理;并对应用二分查找进行迷宫搜

索的具体实现方法和搜索效果进行研究。传统迷宫搜索算法只计算搜索方向,结合二分查找的智能搜索算法通过对迷宫进行区域划分、目标查找计算,在机器人搜索之前甄别并剔除无效区域,实现搜索效率的提高。研究结果显示:

(1) 二分查找方法实现了对迷宫无效搜索区域的甄别和剔除,缩小了机器人在迷宫中的实际搜索范围。

(2) 结合二分查找方法的迷宫智能搜索算法在保证成功率的同时提高了搜索效率,在迷宫搜索路程分别超过 100 格、200 格、300 格的情况下,相较于单独使用 A 算法和 Flood-central 算法,结合二分查找法后搜索效率平均提升了 26%、28%与 41%,尤其是在较为复杂的迷宫地形中搜索效果较好,验证了以缩小迷宫搜索区域的方法来提提高机器人迷宫搜索效率的可行性。

(3) 在少部分迷宫地形中,机器人实际搜索路线未能将迷宫区域二分,本文提出的二分查找方法对此类迷宫处理效果不佳,后续将进一步研究将其他数据查找方法与搜索算法结合,或可实现更高效的迷宫搜索算法。

### 参考文献:

[1] 袁臣虎,董佳丽,周佳文,等. 迷宫机器人外置磁编码器最优位置的确定[J]. 天津工业大学学报, 2022, 41(6): 70-75, 82.  
Yuan Chenhu, Dong Jiali, Zhou Jiawen, et al. Determination of optimal position of external magnetic encoder for labyrinth robot[J]. Journal of Tiangong University, 2022, 41(6): 70-75, 82(in Chinese).

[2] 李靖,杨帆. 分布式协同多机器人多任务目标遍历路径规划[J]. 天津工业大学学报, 2020, 39(6): 68-75.  
Li Jing, Yang Fan. Distributed cooperative multi-robot traversing multi-task target path planning[J]. Journal of Tiangong University, 2020, 39(6): 68-75(in Chinese).

[3] Nabil H, Nasution S M, Astuti Nugrahaeni R. Comparison of real time iterative deepening best first search algorithm and A\* algorithm on maze chase game NPC[C]//2019 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI). Yogyakarta, Indonesia: IEEE, 2019: 85-90.

[4] Youssefi K A, Rouhani M. Swarm intelligence based robotic search in unknown maze-like environments[J]. Expert Systems with Applications, 2021, 178: 114907.

[5] Pame Y G, Kottawar V G, Mahajan Y V. A novel approach to maze solving algorithm[C]//2023 International Conference on Emerging Smart Computing and Informatics (ESCI). Pune,

- India: IEEE, 2023: 1-6.
- [6] Juang L H. Humanoid robot runs maze mode using depth-first traversal algorithm[J]. *Multimedia Tools and Applications*, 2023, 82(8): 11847-11871.
- [7] 戴盛华, 曹景铭. 基于强化学习的电脑鼠优化方法研究[D]. 北京: 北京交通大学, 2021.  
Dai Shenghua, Cao Jingming. Research on computer mouse optimization method based on reinforcement learning[D]. Beijing: Beijing Jiaotong University, 2021 (in Chinese).
- [8] Zhang X P, Li K, Wang Li, et al. A cognitive model of maze robot with emotion and memory mechanism[J]. *Control and Decision*, 2023, 38(10): 2850-2858.
- [9] Mishra S, Bande P. Maze solving algorithms for micro mouse [C]/2008 IEEE International Conference on Signal Image Technology and Internet Based Systems. Bali, Indonesia: IEEE, 2008: 86-93.
- [10] 袁臣虎, 路亮, 王岁, 等. 基于概率距离的电脑鼠走迷宫融合算法研究[J]. *计算机工程*, 2018, 44(9): 9-14.  
Yuan Chenhu, Lu Liang, Wang Sui, et al. Research on micromouse walking maze fusion algorithm based on probability distance[J]. *Computer Engineering*, 2018, 44(9): 9-14 (in Chinese).
- [11] Bansal A, Agrawal S. A robust maze solving algorithm for a micromouse robot[J]. *Advanced Materials Research*, 2011, 403/404/405/406/407/408: 4762-4768.
- [12] Christiansen D. The amazing micromouse roars on spectral lines[J]. *IEEE Spectrum*, 2014, 51(5): 8.
- [13] Lu L, Liu X X, You P Y, et al. Research on micromouse maze search algorithm based on centripetal algorithm[C]/7th International Conference on Computing, Control and Industrial Engineering (CCIE 2023). Singapore: Springer Nature Singapore, 2023: 937-947.
- [14] 高源, 凌翌, 吕鹏, 等. 基于 A\* 算法的迷宫搜索向心法法则[J]. *电子设计工程*, 2019, 27(9): 31-34.  
Gao Yuan, Ling Yi, Lü Peng, et al. Maze searching centripetal rule which based on A\* algorithm[J]. *Electronic Design Engineering*, 2019, 27(9): 31-34 (in Chinese).
- [15] 姚得鑫, 伞红军, 王雅如, 等. 移动机器人路径规划中 A\* 算法的改进研究[J]. *系统仿真学报*, 2024, 36(11): 2684-2698.  
Yao Dexin, San Hongjun, Wang Yaru, et al. Improvement of A\* algorithm in path planning of mobile robot[J]. *Journal of System Simulation*, 2024, 36(11): 2684-2698 (in Chinese).
- [16] Batti H, Ben Jabeur C, Seddik H. Autonomous smart robot for path predicting and finding in maze based on fuzzy and neuro-fuzzy approaches[J]. *Asian Journal of Control*, 2021, 23(1): 3-12.
- [17] Kamalova A, Kim K D, Lee S G. Waypoint mobile robot exploration based on biologically inspired algorithms[J]. *IEEE Access*, 2020, 8: 190342-190355.
- [18] 汤伟, 赵静, 古婵. 基于自动机的迷宫路径规划求解算法优化[J]. *南京邮电大学学报(自然科学版)*, 2021, 41(5): 92-100.  
Tang Wei, Zhao Jing, Gu Chan. Optimization algorithm for maze path planning and solving based on automata[J]. *Journal of Nanjing University of Posts and Telecommunications (Natural Science)*, 2021, 41(5): 92-100 (in Chinese).
- [19] 张肖如, 张福泉, 刘梦. 基于二分查找算法的弱电网电压稳定性动态控制系统[J]. *电子设计工程*, 2020, 28(6): 140-143, 148.  
Zhang Xiaoru, Zhang Fuquan, Liu Meng. Dynamic control system for voltage stability of weak power grid based on binary search algorithms[J]. *Electronic Design Engineering*, 2020, 28(6): 140-143, 148 (in Chinese).
- [20] 成亚萍, 傅德胜. 基于位的二分查找算法[J]. *南京气象学院学报*, 2005, 28(6): 862-866.  
Cheng Yaping, Fu Desheng. A binary search algorithm based on bit[J]. *Journal of Nanjing Institute of Meteorology*, 2005, 28(6): 862-866 (in Chinese).
- [21] 龙舰涵, 李平. 5G 通信中基于二分图匹配的链路均衡负载方法研究[J]. *计算机应用与软件*, 2022, 39(8): 139-144.  
Long Jianhan, Li Ping. Load balancing method for link based on bipartite graph matching in 5G communication[J]. *Computer Applications and Software*, 2022, 39(8): 139-144 (in Chinese).
- [22] 从永正. 基于 ROS 的七自由度双臂协作机器人控制系统研究[D]. 合肥: 合肥工业大学, 2021.  
Cong Yongzheng. Research on control system of seven-degree-of-freedom dual-arm cooperative robot based on ROS[D]. Hefei: Hefei University of Technology, 2021 (in Chinese).
- [23] 乔雅桐. 基于 ROS 平台的仓储 AGV 系统设计及路径规划研究[D]. 太原: 中北大学, 2022.  
Qiao Yatong. Research on storage AGV system design and path planning based on ROS platform[D]. Taiyuan: North University of China, 2022 (in Chinese).

#### 本文引文格式:

- 袁臣虎, 张习文, 高超, 等. 基于二分思想的迷宫机器人智能搜索算法[J]. *天津工业大学学报*, 2026, 45(2): 86-93.  
Yuan Chenhu, Zhang Xiwen, Gao Chao, et al. Intelligent search algorithm for maze robots based on binary search concept[J]. *Journal of Tiangong University*, 2026, 45(2): 86-93 (in Chinese).

(责任编辑:程晓英)