

基于地图预处理及改进 A \* 算法的路径规划<sup>①</sup>余文凯<sup>②\*</sup> 章 政<sup>③\*</sup> 付雪画<sup>\*\*\*</sup> 王昭伟<sup>\*</sup>

(\* 武汉科技大学机器人与智能系统研究院 武汉 430081)

(\*\* 武汉科技大学信息科学与工程学院 武汉 430081)

(\*\*\*) 中山大学数学学院 广州 510970)

**摘 要** 针对移动机器人路径规划中栅格地图建模及 A \* 算法搜索效率问题,设计了一种基于地图分区预处理及改进 A \* 算法的路径规划。首先,基于 K-Means 聚类算法对栅格地图进行分区并量化各局部区域的复杂度;然后,改进 A \* 算法的评价函数和子节点选择方式并依据地图区域的复杂度生成有效的搜索空间;最后,改进 Floyd 算法对路径进行双向平滑度优化处理,并通过添加防碰撞安全距离系数,使路径与障碍物保持安全距离。实验仿真结果表明,本文所设计的算法可提高 A \* 算法的搜索效率和灵活性,增加路径的平滑度和安全性。

**关键词** 移动机器人; 路径规划; A \* 算法; K-Means 聚类算法; Floyd 算法

## 0 引 言

路径规划是实现移动机器人自主无碰撞移动的关键技术之一。机器人根据特定的工作要求,依据路径距离最短、能耗最少或者移动时间最少等优化指标,搜索一条从起始点到目标点的最优路径<sup>[1-4]</sup>。A \* 算法具有搜索效率高、规划速度快和克服了搜索过程中形成的早熟现象等特点,在机器人路径规划中得到了广泛应用<sup>[5,6]</sup>。

传统 A \* 算法在路径规划应用中,随着地图面积的增大其搜索空间会产生多余的搜索节点,导致算法效率降低、规划路径出现不必要的转折等问题。针对这些问题,许多学者从不同的角度提高了 A \* 算法的搜索效率。文献[7]在启发函数中增加了父节点信息,并修改了评价函数的权重,增加启发函数的权重,减少搜索空间来提高算法的搜索效率。文献[8]引入领域矩阵进行障碍搜索以提高路径的安

全性,并通过结合方向角度和位置距离对启发函数进行修改,提高了计算效率。文献[9]加权处理评价函数,并通过人工搜索标记,减少了搜索区域,让搜索的精度下降,来提高算法的效率。文献[10]针对复杂地形的路径规划问题,在 A \* 算法的评价函数中增加安全成本,并将评价函数转换成统一的时间成本,由此提高了机器人的安全性。然而,路径规划问题不仅与路径搜索算法有关,还与环境地图复杂度密切相关。经研究发现,路径规划搜索算法的效率与地图搜索空间的规模成反比,如 A \* 算法的时间复杂度为  $O(n^2)$  ( $n$  为节点数)。因此,通过缩减搜索空间的规模是提高 A \* 算法效率的有效途径之一。

基于上述分析,本文将局部环境信息识别与 A \* 算法相结合,提出了一种基于地图分区预处理及改进 A \* 算法的路径规划算法。一方面,通过对 A \* 算法本身进行改进来提高搜索效率;另一方面,通过预先分析环境地图,依据环境复杂度改变搜索

① 国家自然科学基金(61773298),教育部工程研究中心开放基金(MADT201603)和武汉科技大学国防预研基金(GF201706)资助项目。

② 男,1993年生,硕士生;研究方向:移动机器人智能控制;E-mail: 411816072@qq.com

③ 通信作者,E-mail: kitamizz@wust.edu.cn

(收稿日期:2019-04-01)

空间规模来提高搜索精度。首先,基于栅格法对地图建模,引入 K-Means 聚类算法对地图栅格模型进行分区预处理,根据障碍点数量确定地图区域的规模,并依据障碍物的数量和位置判断每个区域环境的复杂度。然后,依据地图聚类结果,改进 A\* 算法的评价函数和子节点选择方式在不同区域内进行灵活的搜索。最后,改进 Floyd 算法对路径进行防碰撞平滑度双向优化处理。实验结果表明,本文所设计的路径规划优化算法具有适应性强、搜索效率高、路径转折次数少和平滑度高等特点。

## 1 地图分区预处理

在采用栅格法构建地图的模型中,对障碍栅格点的数量及位置信息进行分析,量化环境的复杂度,并将环境复杂度信息加入到 A\* 算法的搜索方式中,可有效提高算法的搜索效率。然而,全局地图的复杂度信息不能有效地描述局部环境的复杂度,需要识别地图的局部信息。在障碍率高的局部区域,需扩大搜索空间,选择穿过高复杂度区域的最小代价路径,在障碍率低的局部区域减少搜索空间,需快速搜索最短路径。

因此,本文采用栅格法建立地图模型后,用 K-Means 聚类算法根据障碍物数量和位置进行分区预处理,判断每个区域的复杂度,在每个区域内进行不同精度的搜索,提高算法局部搜索效率和灵活性,获得全局最优路径。

### 1.1 栅格法地图建模

基于栅格法对移动机器人工作环境进行地图建模。设机器人工作区域为  $m \times n$ , 单元栅格边长为  $l$ , 则 2 维建模地图大小为  $x_{\max} \times y_{\max}$ 。

$$\begin{cases} x_{\max} = \text{ceil}(m/l) \\ y_{\max} = \text{ceil}(n/l) \end{cases} \quad (1)$$

其中,  $\text{ceil}$  为进一取整函数,每个栅格都有对应的坐标表示,并映射一个值(1 或 0)。当该栅格内有障碍物时赋值为 1,用黑色表示;无障碍物时赋值为 0,用白色表示。路径规划时,只需遍历值为 0 的栅格。

### 1.2 地图环境分区预处理

#### 1.2.1 基于 K-Means 的分区处理

K-Means 聚类算法通过对象与聚类中心之间的距离进行聚类,其目标通常是将距离比较相近的对象组成簇,从而得到紧凑而独立的不同簇。此外,K-Means 聚类算法简单、收敛速度快,具有可伸缩性和高效性。本文利用 K-Means 聚类算法对环境地图进行聚类,可完成对环境地图的完全分割,避免剩下多余区域。同时,在对地图分区预处理中,将 2 维栅格地图中的每个障碍栅格点当作聚类样本,K-Means 聚类算法依据这些样本的位置坐标进行聚类处理。地图信息处理时,K-Means 算法通过预先设定的  $K$  值(即聚类簇群数目)进行处理。 $K$  值的设定由地图规模而定,设置过大会导致路径局部最优,设置过小则无法有效量化环境信息<sup>[11,12]</sup>。

#### 1.2.2 量化环境信息

通过 K-Means 聚类算法预处理后,可分别提取每个族群的环境信息。为量化地图的信息,设计障碍率  $P_o$  和直通率  $P_t$ 。

$$P_o = \frac{n_o}{N} \quad (2)$$

$$P_t = \frac{n_t}{L + D} \quad (3)$$

其中,  $n_o$  是值为 1 的栅格(障碍栅格)的数目,  $N$  是族群内栅格总数,  $n_t$  是一行或一列值都为 0 的栅格(无障碍栅格)条数,  $L$  和  $D$  是该族群的栅格行数和列数。

障碍率  $P_o$  描述了环境中障碍物数量的多少,  $P_o$  越大,障碍物越多。直通率  $P_t$  描述了环境中障碍物位置的混乱程度,  $P_t$  越大,混乱度越小。通过障碍率  $P_o$  和直通率  $P_t$  描述区域环境中障碍物的复杂情况,其数值大小影响改进 A\* 算法的评价函数,使得搜索空间随地图环境变化而改变,在不同的环境中更有更加有效的搜索方式,增加了 A\* 算法的灵活性,提高搜索效率。

#### 1.2.3 地图分区处理算法步骤

基于 K-Means 地图分区处理的算法步骤如下:

**步骤 1** 输入障碍栅格点样本  $D = \{x_1, x_2, \dots, x_m\}$ , 从数据  $D$  中随机选择  $k$  个质心向量  $\{u_1, u_2, \dots, u_k\}$ 。

**步骤 2** 设置最大迭代次数  $N$ 。

(1) 将簇划分  $k$  类,并初始化  $C_i (i = 1, 2, \dots,$

$k$ ) 为空集;

(2) 计算样本  $x_i (i = 1, 2, \dots, m)$  到各个质心向量  $u_j (j = 1, 2, \dots, k)$  的距离,  $d_{ij} = \|x_i - u_j\|_2^2$ , 将  $x_i$  标记最小的为  $d_{ij}$  所对应的类别  $C_i$ 。此时更新  $C_i = C_i \cup \{x_i\}$ ;

(3) 对  $C_i (i = 1, 2, \dots, k)$  中所有的样本点重新计算新的质心  $u_i = \frac{1}{|C_i|} \sum_{x \in C_i} x$ ;

(4) 如果所有的  $k$  个质心向量  $\{u_1, u_2, \dots, u_k\}$  都没有发生变化, 转到步骤(3); 否则, 转到步骤(2)重新计算。

**步骤 3** 输出簇划分  $C = \{C_1, C_2, \dots, C_k\}$ 。

**步骤 4** 由式(2)和式(3)计算  $k$  类簇群  $\{C_1, C_2, \dots, C_k\}$  区域内的障碍率  $P_{o_i}$  和直通率  $P_{t_i} (i = 1, 2, \dots, k)$ 。

**步骤 5** 改进 A\* 算法在  $C_i (i = 1, 2, \dots, k)$  区域搜索时, 输出该区域的障碍率  $P_o$  和直通率  $P_t$  到评价函数  $f(n)$  中。

## 2 融合环境信息的改进 A\* 算法

本文基于 K-Means 聚类算法对地图进行分区预处理后, 设计了 A\* 算法的评价函数  $f(n)$ , 将量化环境信息的数据加入评价函数中。然后, 改进子节点选择方式, 避免路径斜穿过障碍物顶点。最后, 改进 Floyd 算法对规划后的路径进行双向平滑度优化处理, 并设置安全距离防碰撞。

### 2.1 评价函数设计

A\* 算法是解决静态路网中最短路径的最有效的直接搜索方法, 也是解决许多搜索问题的有效算法<sup>[3]</sup>。算法中距离估算值与实际值越接近, 最终搜索速度越快。其评价函数为

$$f(n) = g(n) + h(n) \quad (4)$$

其中,  $f(n)$  为评价函数,  $g(n)$  为起始点到当前节点  $n$  实际路径代价的代价函数,  $h(n)$  为当前节点  $n$  到目标点估计路径代价的启发函数。本文的评价函数中引入了环境信息和机器人位置信息, 并设置了代价函数和估价函数的权重函数。

$$g(n) = \sqrt{(n_x - s_x)^2 + (n_y - s_y)^2} \quad (5)$$

$$h(n) = \sqrt{(n_x - g_x)^2 + (n_y - g_y)^2} \quad (6)$$

$$f(n) = r_1 g(n) + r_2 h(n) \quad (7)$$

其中,  $r_1 = P_t + d_{sn}/d_{st}$ ,  $r_2 = (1 - P_o) + e^{d_{nt}/d_{st}}$ ,  $d_{sn}$  表示起始点到当前点的距离,  $d_{nt}$  表示当前点到目标点的距离,  $d_{st}$  表示起始点到目标点的距离。

将预处理中式(2)和式(3)得到的障碍率  $P_o$  和直通率  $P_t$  数据代入式(7)中, 评价函数  $f(n)$  直接影响了路径规划的好坏和效率。代价函数  $g(n)$  越大, 评价函数  $f(n)$  就会越大于实际的代价, 导致算法效率降低, 路径距离变长, 从而使移动机器人运动代价过高。启发函数  $h(n)$  的权重越大, 算法的运算速度就越快, 但路径会更加趋近代价最小路径, 陷入局部最优, 造成转弯次数较多的路线。

因此, 本文采用 K-Means 聚类算法对地图进行分区处理, 判断区域内的环境复杂度。当障碍物较少时, 障碍率  $P_o$  值较小, 则让启发函数  $h(n)$  权重增大, 从而减少搜索空间, 提高搜索效率。当障碍物越多时, 障碍率  $P_o$  值越大, 则让启发函数  $h(n)$  权重减少, 增大搜索空间, 提高搜索精度。当障碍物越多, 摆放很整齐时, 直通率  $P_t$  增加, 使得代价函数  $g(n)$  权重增大, 增加路径搜索空间, 以获得全局最优的路径, 避免出现多曲折转弯的路径。

将环境信息和移动机器人位置信息都融入到评价函数中。在不同的区域内, 权重函数进行自适应调整, 产生不同的有效搜索空间, 不仅保证算法的快速性和灵活性, 还保证路径的全局最优。

### 2.2 优化子节点的选择

传统的 A\* 算法中, 判断下一个子节点时, 选择条件只考虑了子节点是否存在障碍物, 没有考虑子节点与障碍物的位置关系。所以, 规划出的路径斜穿过障碍物的顶点, 存在发生碰撞的风险。针对此问题, 本文增加了子节点选择规则, 避免路径斜穿过障碍物顶点。

根据 8 个子节点与父节点的坐标关系, 如图 1 子节点图。将在父节点的上下 2 个子节点(子节点 2, 6)分为 I 类, 将在父节点的左右 2 个子节点(子节点 4, 8)分为 II 类, 进行优化子节点的选择。

(1) 如果障碍物为 I 类, 删除该障碍点的左右

2 个可选子节点(如子节点 1,3 或者子节点 5,7)。

(2) 如果障碍物为 II 类,删除该障碍点的上下 2 个可选子节点(如子节点 1,7 或者子节点 3,5)。

(3) 如果障碍点不是 I 类和 II 类,则不做处理。

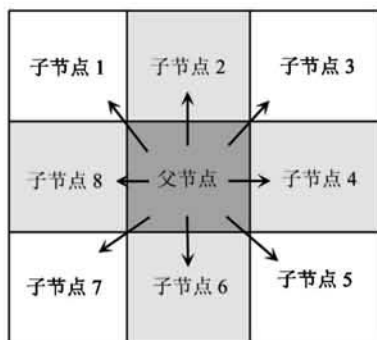


图1 子节点图

改进后,规划后的路径不再穿过障碍物的顶点,避免移动机器人在运动中与障碍物发生碰撞。

### 2.3 基于改进 Floyd 算法优化路径平滑度

Floyd 算法<sup>[13]</sup>又称为插点法,是一种利用动态规划的思想寻找给定的加权图中多源点之间最短路径的算法。基于栅格法地图建模所规划出的路径,存在多次转折、路径节点只能在栅格中心的问题,导致路径不是最短且平滑度较差。针对这些问题,依据 Floyd 算法,设计了双向优化处理过程,常规 Floyd 算法从起始点到目标点正方向进行平滑度优化,本文改进 Floyd 算法,增加从目标点到起始点的反方向优化,实现双向平滑度优化,可有效减少路径长度,并增加安全距离判断,避免与障碍物发生碰撞。

在改进 Floyd 算法双向优化中,增加防碰撞隔离距离,通过障碍点到连线的垂直距离与设置的安全距离的关系,判断路径是否安全,实现防碰撞。障碍隔离关系如图 2 所示,路径节点 A 的坐标为  $(x_a, y_a)$ , B 的坐标为  $(x_b, y_b)$ , 障碍点 C 的坐标为  $(x_c, y_c)$ , 隔离距离为点 C 与节点 AB 之间线段的垂直距离  $d$ , 到节点 AB 之间线段纵轴的距离 CD 为  $l$ , 节点 AB 之间线段与横轴之间的夹角为  $\theta$ 。

由 A、B 及 C 3 点坐标可求出障碍点到路径的距离为

$$d = l \cos \theta \quad (8)$$

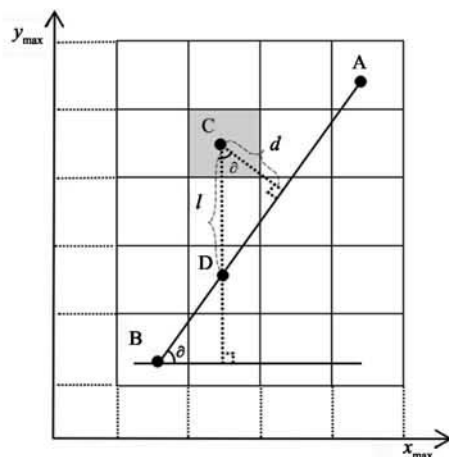


图2 障碍隔离图

其中,  $l = |y_c - (\frac{y_a - y_b}{x_a - x_b}(x_c - x_a) + y_a)|$ ,  $\theta = |\arctan \frac{y_a - y_b}{x_a - x_b}|$ 。

因此,设置安全距离  $D$ , 进行碰撞隔离处理。其大小与单元栅格的边长  $d_{cell}$  有关,满足  $D > \frac{d_{cell}}{\sqrt{2}}$ 。计算障碍点到路径的距离  $d$  与安全距离  $D$  的大小关系,若  $d \leq D$ , 则该路径不能选择;若  $d > D$ , 则该路径可以选择。

改进 Floyd 算法将规划出的路径节点进行平滑度优化,增加从目标点到起始点方向的优化,进行双向平滑度优化,设置 Floyd 算法中取点步长  $k$  的数值,并设置安全距离  $D$  的参数大小。以图 3 中所示的路径为例说明双向优化路径平滑度步骤,图中 S 为起始点, G 为目标点。

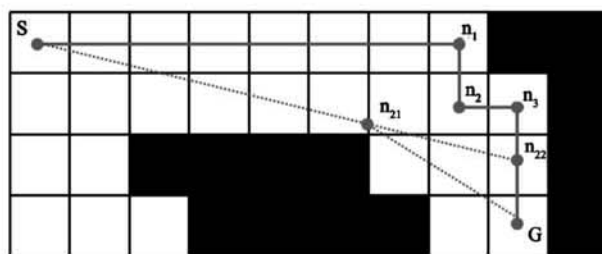


图3 路径平滑优化例图

**步骤 1** 删除路径节点同一线上的中间节点,只保留起点、拐点和终点。如图 3 所示,处理后路径为  $S \rightarrow n_1 \rightarrow n_2 \rightarrow n_3 \rightarrow G$ 。



**步骤 2** 从起始点 S 方向,在保留的 2 路径节点之间每隔  $k$  步长取点,每取一个节点与前端路径节点,判断 2 点之间路径是否有障碍物,并由式(8)计算路径旁边障碍物到路径的距离  $d$ ,判断与安全距离  $D$  的大小。若障碍物在安全距离之外,满足要求则选择当前节点为路径节点,否则不选择。如图 3 所示,处理后路径为  $S \rightarrow n_{22} \rightarrow G$ 。

**步骤 3** 从目标点 G 方向,反向取点判断(重复步骤 2 的判断方式),如图 3 所示,处理后路径为  $S \rightarrow n_{21} \rightarrow G$ 。

依据改进的 Floyd 算法对规划后的路径进行双向优化,并增加防碰撞隔离,使得路径节点选择不再局限于栅格中心点,减少了转折度数和路径长度,减少了运动代价,增加了路径的平滑度。

### 3 仿真实验及结果分析

首先建立  $30 \times 30$  的移动机器人工作环境地图,将本文算法与地图未预处理的改进 A\* 算法和传统 A\* 算法进行对比仿真测试,验证基于 K-Means 地图分区预处理及改进 A\* 算法的路径规划算法优点。再扩大地图规模,建立  $50 \times 50$  的环境地图模型,将本文算法与传统 A\* 算法对比仿真测试,验证本文算法的优点及灵活性。

为方便对比研究,假定移动机器人工作环境栅格边长为单位 1 m,设安全距离  $D = 0.8$  m,取点步长  $k = 0.1$  m。地图中 S 表示起始点, T 表示目标点。黑色区域为障碍物,灰色区域表示该算法的搜索空间。在基于 K-Means 地图分区预处理及改进 A\* 算法中,以不同形状的符号表示各个区域的障碍物,区别不同的分区,以圆圈符号表示基于 K-Means 聚类处理出该区域的中心质点。

同时,为综合评价算法的性能,设定了评判函数  $T$ ,  $T$  是机器人按算法规划的路径进行移动所需要消耗的时间。

$$T = L/v + N + \theta/45 \quad (9)$$

其中,  $L$  表示规划路径的长度,  $v$  表示移动机器人运动速度,假设速度为 1 m/s,  $N$  表示转弯次数,  $\theta$  表示转角总度数,假设每次转弯消耗加减速的时间为

1 s,每转  $45^\circ$  消耗的时间为 1 s。

#### 3.1 实验 1

参考自动导引小车和其他室内移动机器人的工作环境,设计了  $30 \times 30$  的仿真环境地图。基于传统 A\* 算法的路径规划如图 4 所示。基于改进 A\* 算法,在未分区处理的地图及基于 K-Means 预处理的地图环境中(聚类簇群数目  $K = 4$ ),路径规划分别如图 5 和图 6 所示。其中,对于未分区处理的算法中直接计算整个地图环境的障碍率和直通率。3 种算法的实验路径参数如表 1 所示。

由表 1 的实验数据可知,传统 A\* 算法搜索空间数量为 359 个,远大于其他 2 种算法,改进后 A\*

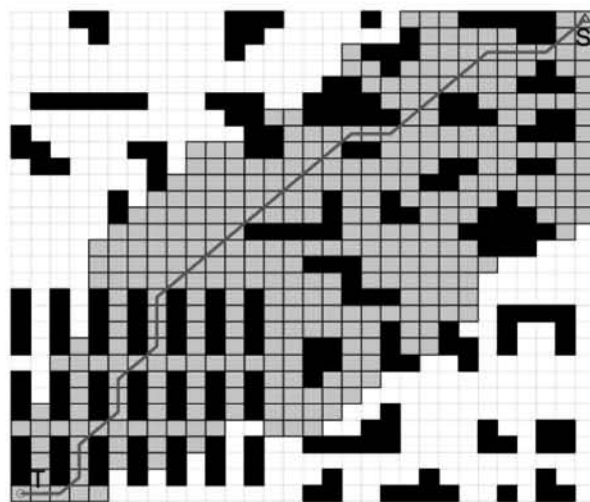


图 4 传统 A\* 算法规划路径

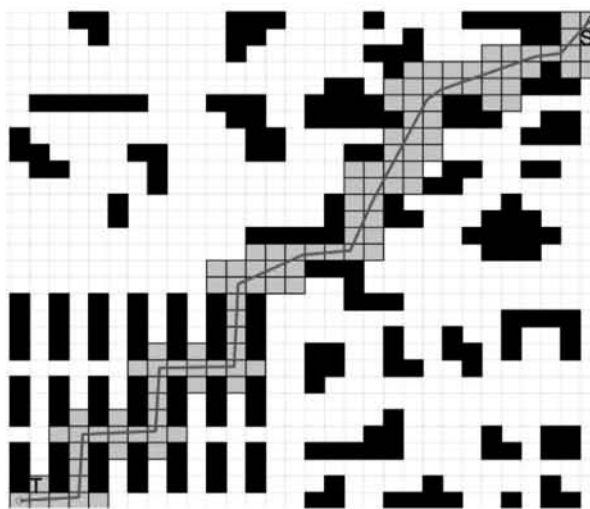


图 5 未分区处理改进 A\* 算法规划路径

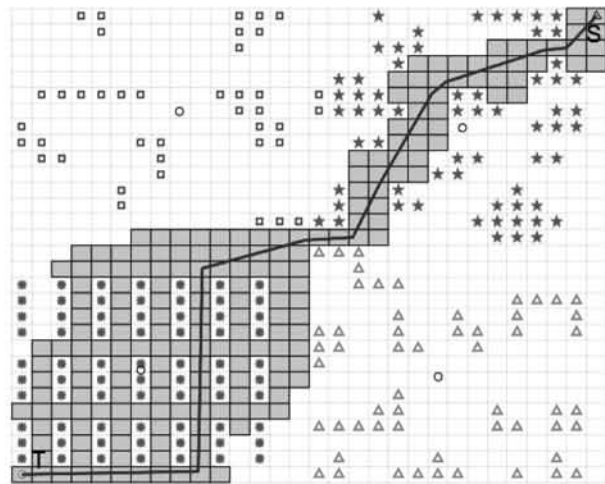


图 6 基于 K-Means 分区预处理改进 A\* 算法规划路径

表 1 实验路径参数表

| 路径参数               | 传统 A* 算法 | 未分区处理改进 A* 算法 | 分区预处理改进 A* 算法 |
|--------------------|----------|---------------|---------------|
| 遍历节点(个)            | 359      | 108           | 228           |
| 路径长度(m)            | 45.11    | 49.1          | 49.6          |
| 转角总和( $^{\circ}$ ) | 495      | 673           | 342.2         |
| 转角次数(次)            | 11       | 13            | 9             |
| 移动时间 $T$ (s)       | 67.1     | 77            | 66            |

算法效率明显提高。未作地图预处理的改进 A\* 算法没能有效识别局部信息,陷入了局部最优解,导致路径频繁转弯,消耗了机器人的移动时间。而地图分区预处理的改进 A\* 算法对局部复杂区域扩大了搜索空间,转弯次数减少了 46%,转角总和减少了 49%。实验表明,结合了 K-Means 地图分区预处理的改进 A\* 算法灵活性更高,能依据环境信息的不同,有效改变搜索空间,获得全局上的最优路径。在长度差不多的情况下,转角次数减少,转角总和减少,提高了路径的平滑度。

在此地图内进行 5 次仿真测试,选择不同起始点和不同目标点进行实验。由式(9)计算 5 组实验的消耗时间,对比移动机器人消耗时间  $T$ ,如图 7 所示。

通过 5 次仿真测试,由图 7 中数据可得,基于 K-Means 地图分区域的改进 A\* 算法比其他 2 种算法所消耗移动时间  $T$  都少。在实际应用中,减少移动时间,提高了移动机器人的工作效率。

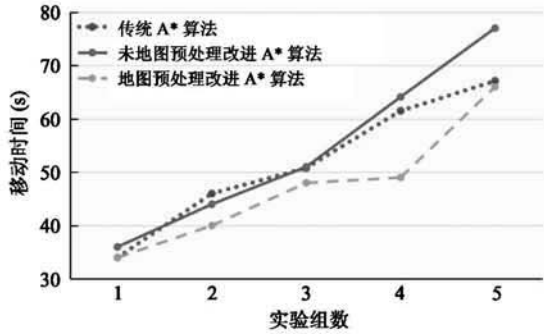


图 7 路径消耗时间  $T$  值对比图

### 3.2 实验 2

为了进一步验证本文算法的搜索效率和环境适应性,将地图规模由  $30 \times 30$  扩大为  $50 \times 50$ ,环境随机产生障碍物。在基于 K-Means 聚类地图预处理算法中,  $K = 5$ 。基于传统 A\* 算法和本文所设计算法的路径搜索如图 8 和图 9 所示。

从规划出的路径图 8 和图 9 可知,本文算法搜索空间减少,搜索效率提高,规划出的路径始终与障碍物保持安全距离,避免了移动机器人与障碍物发生碰撞的危险。由表 2 路径数据表可知,本算法规划出的路径长度和转角次数少于传统 A\* 算法规划出的路径,转角总和比传统 A\* 算法减少 63.3%,平滑度提高。在算法的效率上,改进 A\* 算法搜索空间数减少了 54.8%,算法的搜索效率也得到了提高,减少了消耗的移动时间  $T$ ,提高了移动机器人的工作效率。

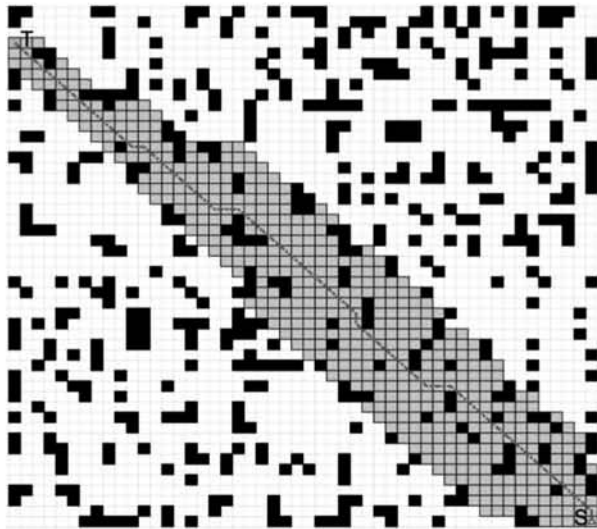


图 8 传统 A\* 算法规划路径

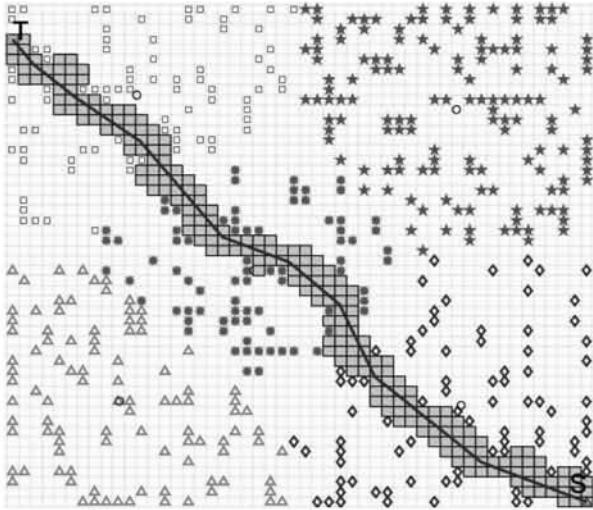


图9 基于 K-Means 分区预处理改进 A\* 算法规划路径

表2 实验路径参数表

| 路径参数               | 传统 A* 算法 | 分区预处理<br>改进 A* 算法 |
|--------------------|----------|-------------------|
| 遍历节点(个)            | 484      | 219               |
| 路径长度(m)            | 69.2     | 69                |
| 转角总和( $^{\circ}$ ) | 405      | 148.7             |
| 转角次数(次)            | 9        | 8                 |
| 移动时间 $T(s)$        | 87.3     | 80.4              |

实验表明,本文算法比传统的 A\* 算法不仅减少了搜索空间,提高了算法的搜索效率,还提高了路径的平滑度和安全性。

## 4 结 论

本文针对地图外部的环境因素及 A\* 算法内部的运算因素对移动机器人路径规划搜索效率的影响问题,设计了一种基于地图分区预处理及改进 A\* 算法的路径规划。基于 K-Means 算法对地图进行聚类分区预处理,并以此对 A\* 算法的评价函数和子节点选择方式进行改进,能在不同区域依据环境信息采用不同搜索方式,并改进 Floyd 算法进行双向平滑度优化,防碰撞处理。仿真结果表明,本文算法在根据地图环境进行有效构建路径的搜索空间基础上,改进 A\* 算法的搜索机制和灵活度,从而保证路径规划效率。地图分区预处理可以提前对地图分析并量化环境的复杂度,平均有效减少 54.8% 的搜索

空间,提高了算法搜索效率、精度和灵活性,规划出的路径始终与障碍物保持安全距离,避免发生碰撞,而且转折度数减少了 63.3%,平滑度得到了较大提升。路径消耗机器人的移动时间更少,提高了移动机器人的工作效率。

## 参考文献

- [1] 王春颖,刘平,秦洪政. 移动机器人的智能路径规划算法综述[J]. 传感器与微系统, 2018,37(8):5-8
- [2] 皇甫淑云,唐守峰,童紫原. 自主移动机器人路径规划方法研究综述[J]. 软件导刊, 2018,17(10):1-5
- [3] 姜明浩,陈洋,李威凌. 基于动态运动基元的移动机器人路径规划[J]. 高技术通讯, 2016,26(12):997-1005
- [4] Joonwoo L. Heterogeneous-ants-based path planner for global path planning of mobile robot applications[J]. *International Journal of Control, Automation and Systems*, 2017, 15(4): 1754-1769
- [5] 张红梅,李明龙,杨乐. 基于改进 A\* 算法的移动机器人安全路径规划[J]. 计算机仿真, 2018,35(4):319-324
- [6] Tan J, Zhao L, Wang Y, et al. The 3D path planning based on A\* algorithm and artificial potential field for the Rotary-Wing flying robot[C]// International Conference on Intelligent Human-Machine Systems and Cybernetics, Hangzhou, China, 2016: 551-556
- [7] 王维,裴东,冯璋. 改进 A\* 算法的移动机器人最短路径规划[J]. 计算机应用, 2018,38(5):1523-1526
- [8] 陈若男,文聪聪,彭玲. 改进 A\* 算法及其在室内机器人路径规划中的应用[J]. 计算机应用, 2018,26(11):1-9
- [9] Fu B, Chen L, Zhou Y, et al. An improved A\* algorithm for the industrial robot path planning with high success rate and short length[J]. *Robotics & Autonomous Systems*, 2018, 106: 26-37
- [10] Zhang H M, Li M L, Yang L. Safe path planning of mobile robot based on improved A\* algorithm in complex terrains[J]. *Algorithms*, 2018, 11(4):44-50
- [11] 王建仁,马鑫,段钢龙. 改进的 K-means 聚类 k 值选择算法[J]. 计算机工程与应用, 2018, 23(12):1-8
- [12] 海沫. 大数据聚类算法综述[J]. 计算机科学, 2016, 43(1):380-383

- [13] 张岩, 杨龙. 最短路问题的 Floyd 算法优化及分析  
[J]. 信息技术, 2017, 25(10):30-32

## Path planning based on map partition preprocessing and improved A \* algorithm

Yu Wenkai<sup>\*\*\*</sup>, Zhang Zheng<sup>\*\*\*</sup>, Fu Xuehua<sup>\*\*\*</sup>, Wang Zhaowei<sup>\*\*\*</sup>

(<sup>\*</sup> Institute of Robotics and Intelligent Systems, Wuhan University of Science and Technology, Wuhan 430081)

(<sup>\*\*</sup> School of Information Science and Engineering, Wuhan University of Science and Technology, Wuhan 430081)

(<sup>\*\*\*</sup> School of Mathematics, Sun Yat-sen University, Guangzhou 510970)

### Abstract

An path planning algorithm based on map partition preprocessing and improved A \* algorithm is proposed for the problem of the raster map modeling and searching efficiency of A \* algorithm for mobile robot. Firstly, the K-Means algorithm is applied to realize the partition of maps and quantify the complexity of each local area in the map. Then, the improved A \* algorithm evaluation function and the selection mode of each sub-nodes are designed to generate the effective search space based on the complexity of region decomposition. Finally, the Floyd algorithm is introduced to solve the bi-directional smoothness optimization of search path, meanwhile, the path and the obstacle are kept at a safe distance by adding the anti-collision safety distance coefficient. Simulation experiment results show that the proposed algorithm can improve the search efficiency and flexibility of A \* algorithm and increases the smoothness and security of the path.

**Key words:** mobile robot, path planning, A \* algorithm, K-Means clustering algorithm, Floyd algorithm