

# 控制与决策

Control and Decision

一种Halton序列的HDRRT移动机器人融合规划算法

钟华庚, 罗高生, 王芳, 周正权, 陈祎

引用本文:

钟华庚, 罗高生, 王芳, 周正权, 陈. 一种Halton序列的HDRRT移动机器人融合规划算法[J]. *控制与决策*, 2023, 38(6): 1551–1559.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2021.1709>

---

## 您可能感兴趣的其他文章

### Articles you may be interested in

[基于动态变采样区域RRT的无人车路径规划](#)

Path planning of unmanned vehicle based on dynamic variable sampling area RRT  
*控制与决策*. 2023, 38(6): 1721–1729 <https://doi.org/10.13195/j.kzyjc.2021.1555>

[一种基于改进RRT\\*的全局路径规划算法](#)

A global path planning algorithm based on improved RRT\*  
*控制与决策*. 2022, 37(4): 829–838 <https://doi.org/10.13195/j.kzyjc.2020.1354>

[机器人信息增益RRT环境探索算法](#)

Robot RRT based on information gain for environment exploration  
*控制与决策*. 2021, 36(11): 2683–2689 <https://doi.org/10.13195/j.kzyjc.2020.1007>

[基于改进RRT\\*FN算法的机器人路径规划](#)

Robot path planning based on improved RRT\*FN algorithm  
*控制与决策*. 2021, 36(8): 1834–1840 <https://doi.org/10.13195/j.kzyjc.2019.1713>

[一种基于免疫机理的确定性移动机器人路径规划算法](#)

A path planning algorithm of deterministic mobile robot based on immune mechanism  
*控制与决策*. 2021, 36(10): 2418–2426 <https://doi.org/10.13195/j.kzyjc.2020.0059>

# 一种Halton序列的HDRRT移动机器人融合规划算法

钟华庚<sup>1,2,3</sup>, 罗高生<sup>2,3†</sup>, 王芳<sup>1</sup>, 周正权<sup>4</sup>, 陈祎<sup>4</sup>

(1. 上海海洋大学 工程学院, 上海 201306; 2. 上海海洋大学 海洋科学学院, 上海 201306; 3. 上海海洋大学 上海深渊科学工程技术研究中心, 上海 201306; 4. 中海油能源发展装备技术有限公司, 天津 300452)

**摘要:** 针对标准快速扩展随机树(RRT)算法采用伪随机序列导致采样点分布不均、不合理,且移动机器人从起始点到目标点路径有冗余路段及冗余节点的问题,提出HDRRT(halton & dijkstra & rapidly exploring random tree)算法,该算法采用采样点分布均匀性好的Halton序列进行采样,并利用候选点集策略对节点进行筛选,以剔除冗余节点;同时该算法采用改进的Dijkstra算法提取原始路径关键节点,以减少路径冗余路段;在此基础上采用3次B样条曲线对路径作平滑处理.经Matlab联合ROS系统仿真结果表明,HDRRT算法相对于Bias-RRT和标准RRT算法具有快速性,稳定规划出最短以及平滑路径等优点.

**关键词:** 路径规划; RRT; Halton序列; 采样方法; 改进Dijkstra算法; 移动机器人; ROS系统

中图分类号: TP242.2 文献标志码: A

DOI: 10.13195/j.kzyjc.2021.1709

引用格式: 钟华庚,罗高生,王芳,等.一种Halton序列的HDRRT移动机器人融合规划算法[J].控制与决策,2023,38(6): 1551-1559.

## A Halton sequence fusion planning algorithm for HDRRT mobile robots

ZHONG Hua-geng<sup>1,2,3</sup>, LUO Gao-sheng<sup>2,3†</sup>, WANG Fang<sup>1</sup>, ZHOU Zheng-quan<sup>4</sup>, CHEN Yi<sup>4</sup>

(1. College of Engineering Science and Technology, Shanghai Ocean University, Shanghai 201306, China; 2. College of Marine Sciences, Shanghai Ocean University, Shanghai 201306, China; 3. Shanghai Engineering Research Center of Hadal Science and Technology, Shanghai Ocean University, Shanghai 201306, China; 4. CNOOC Energy Development Equipment Technology Co., Ltd, Tianjin 300452, China)

**Abstract:** Aiming at the problem that the standard RRT algorithm (rapidly exploring random tree) using the pseudo-random sequence leads to uneven and unreasonable distribution of sampling points, and there are redundant sections and redundant nodes in the path of mobile robots from the starting point to the target point, a halton & dijkstra & rapidly exploring random tree (HDRRT) algorithm is proposed, which uses the Halton sequence with good uniform distribution of sampling points for sampling, and uses the candidate point set strategy to filter nodes to eliminate redundant nodes. At the same time, the improved Dijkstra algorithm is used to extract the key nodes of the original path to reduce the redundant path sections. On this basis, the path is smoothed by the cubic B-spline curve. The simulation results of the ROS system combined with Matlab show that the HDRRT algorithm has the advantages of rapidity, stable planning and smooth path compared with the Bias-RRT and standard RRT algorithm.

**Keywords:** path planning; rapidly exploring random tree (RRT); Halton sequence; sampling method; improved Dijkstra algorithm; mobile robot; robot operating system

## 0 引言

移动机器人作为智能系统,可以提高生产率并提供各种便利,因而备受关注.路径规划模块引导机器人在障碍环境中安全移动方面起着关键作用.路径规划的目标是在移动机器人运动约束下,引导机器人从起始点移动到目标点.主要应用于移动机器人的算法有概率完备全局算法:快速扩展随机树

(rapidly-exploring random tree, RRT)和概率路图法(probabilistic roadmap, PRM)<sup>[1]</sup>;仿生算法:遗传算法、粒子群算法、蚁群算法<sup>[2-5]</sup>、A\*算法和人工势场算法等<sup>[6-8]</sup>.仿生算法具有比较好的鲁棒性,但易陷入局部最优,且收敛时间较长<sup>[5]</sup>;A\*算法所生成路径平滑度较差,转折较多,算法搜索效率不高<sup>[7]</sup>;人工势场法规划路径平滑,但存在局部最小值的缺点<sup>[8]</sup>;RRT算法

收稿日期: 2021-10-05; 录用日期: 2022-02-25.

基金项目: 国家自然科学基金项目(52071203); 上海市科技攻关计划项目(20dz1206500); 水下设施智能清洗巡检机器人技术研究项目(D-8006-21-0100).

†通讯作者. E-mail: gsluo@shou.edu.cn.

具有概率完备性,算法搜索速度快,对地图进行随机采样寻找目标点,适用于高低维路径规划<sup>[9]</sup>,但是,RRT算法随机性强,导致规划时间长,路径变化大,冗余节点多,路径不平滑,不符合移动机器人运动约束。

RRT算法的许多变体被提出,文献[10]提出直接采样的Informed-RRT\*,使用椭圆子集作为候选点集,减小了采样范围.文献[11]提出了MOD-RRT\*,该方法将Pareto理论、蚁群算法与RRT相融合,当路径与障碍物发生干涉时,算法通过Pareto理论选择最佳候选节点;蚁群算法对生成的路径进行寻优,去除冗余节点.文献[12]提出了基于势函数的RRT\*(P-RRT\*),将人工势场算法(artificial potential field, APF)与RRT\*算法相融合,APF使RRT\*的探索更加具有方向性,使P-RRT\*比RRT\*收敛更快.文献[13]提出了Quick-RRT\*算法,将RRT\*重选父节点的范围进一步扩大,使得相对于RRT\*算法优化区域更大.文献[14]提出了一种新的算法PQ-RRT\*,它结合了P-RRT\*和Quick-RRT\*的优点,PQ-RRT\*保证快速收敛到最优解,并生成更好的初始解.文献[15]提出了RRT-connect算法,该算法同时在起始点和目标点分别生成两颗随机树,大大提升了RRT算法的搜索速度,但是RRT-connect算法还是没有克服伪随机序列采样带来的随机性、路径变化大的缺点.文献[16]结合RRT\*和RRT-connect算法的优点,提出了RRT\*-connect算法,使得算法在快速收敛的同时,达到局部路径最优。

综上所述,造成RRT算法随机性强、路径变化大的原因之一是RRT算法采用了伪随机序列采样.RRT算法累积冗余节点与路段多,不满足移动机器人运动约束.一般的RRT算法采样方法有均匀采样<sup>[17]</sup>和随机采样<sup>[18]</sup>.均匀采样在几何对象上均匀分布,但是当采样面积较大时,采样点数量激增,导致路径规划算法效率低;随机采样在一定程度上减少了采样点的数量,但是随机采样是依托伪随机序列进行采样,会出现重复、不合理的采样点.本文针对以上RRT算法出现的问题,提出HDRRT算法.该算法是基于Halton序列采样的RRT算法(加入了候选点集策略和目标偏置策略),并采用改进的Dijkstra算法对路径关键节点二次提取以及利用三次B样条曲线对最终路径进行优化:1)相对于均匀采样和随机采样,Halton序列具有分布均匀性和覆盖性好、低差异性的优点,这种采样方法与分辨率取值没有关系<sup>[19]</sup>;2)采用候选点集对地图进行整体采样;3)改进的Dijkstra算法对路径关键节点进行提取,以剔除

不必要节点.

## 1 RRT算法定义

RRT算法拓展过程类似树叶生长, $X_{init}$ (初始点)通过固定的stepsize向终点拓展,在 $C_{free}$ 空间中使用伪随机序列进行采样得到采样点 $X_{rand}$ ,遍历随机树所有节点,找出 $d = \min[\text{distance}(\text{RRTTree}(i), X_{rand})]$ 的节点 $X_{near}$ , $\text{RRTTree}(i)$ 为随机树集合;通过 $X_{near}$ 与 $X_{rand}$ 两点计算出新节点的拓展方向, $X_{near}$ 在 $\overrightarrow{X_{rand}X_{near}}$ 方向拓展固定步长(stepsize)得到 $X_{new}$ 节点.如果 $X_{new}$ 在拓展的过程与障碍物干涉,则舍弃 $X_{rand}$ 采样节点;否则,将 $X_{new}$ 新节点加入随机树中.最后判断 $X_{new}$ 和 $X_{goal}$ 的距离是否小于误差值;如小于误差值则视为到达终点,停止搜索,否则继续在 $C_{free}$ 空间中采样.

RRT算法具体步骤如下:

初始化:起始点、目标点、随机树 $T$ 、固定步长stepsize、误差值.

step 1:对随机树进行初始化,将 $X_{init}$ 加入随机树 $T$ 中.

step 2:在 $C_{free}$ 中随机采样,得到采样点 $X_{rand}$ ;遍历随机树 $T$ 找到距离 $X_{rand}$ 最近的节点 $X_{near}$ 作为 $X_{new}$ 父节点.

step 3:计算 $\overrightarrow{X_{rand}X_{near}}$ 方向, $X_{near}$ 在 $\overrightarrow{X_{rand}X_{near}}$ 上拓展固定步长stepsize得到新节点 $X_{new}$ .

step 4:如果 $X_{new}$ 在拓展的过程与障碍物发生碰撞,则舍弃 $X_{rand}$ ,重新采样;否则将 $X_{new}$ 节点加入随机树 $T$ 中.

step 5:计算距离函数 $d = \text{distance}(X_{new}, X_{goal})$ ,如果 $d$ 小于误差值,则视为到达终点,否则继续采样.

step 6:  $\text{path} \leftarrow \text{rewire}(X_{near}, X_{new})$ ,通过 $X_{new}$ 找到 $X_{near}$ ,回溯完整路径.其中: $\text{rewire}(X_{near}, X_{new})$ 为回溯路径函数.

## 2 HDRRT算法

### 2.1 Halton 采样序列

一般的RRT算法,所采用的随机采样算法核心为伪随机序列算法,伪随机序列算法虽然较均匀采样法一定程度上减少了采样点的数量,但是伪随机序列采样点随机性强,容易产生聚集、不合理的采样点,导致节点数量增加,路径冗余.本文提出以Halton序列为采样方式的HDRRT算法,Halton序列的采样点具有分布均匀、空间覆盖性好、低差异性的优点.文献[20]中证明了依托Halton序列采样的概率路图法收敛时间是稳定的,而依托伪随机序列采样算法收敛时间波动幅度较大,RRT算法与概率路图法均是基于

采样的路径规划算法. Halton序列是基于一定的质数基实现的,一个维度对应一个质数,一般情况下按照质数的顺序取值,有助于不同维度之间结合后不同采样点之间距离的控制<sup>[21]</sup>.

Halton序列(质数 $k$ 为例):十进制 $1, 2, \dots, m, \dots$ ,  $M$ 用 $k$ 进制表示为

$$m = \sum_{i=0}^i b_i(m)k^i, \quad 0 \leq b_i(m) \leq k-1, k^i \leq m < k^{i+1}. \quad (1)$$

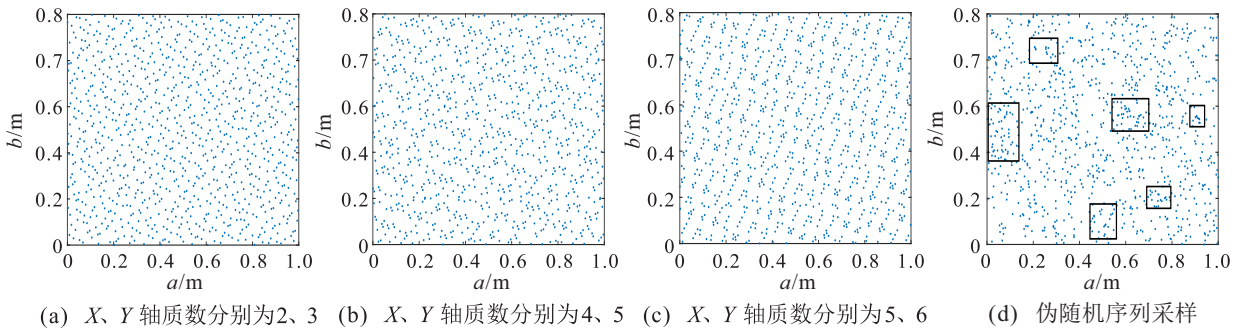


图1 Halton采样序列与伪随机序列分布情况

由图1(a)~图1(c)可知, Halton序列随着 $X$ 、 $Y$ 轴质数的增加,采样点相关性越高,相邻节点距离越小,同时采样点会出现规律性聚集和部分区域空缺.如图1(c)所示,采样点的聚集会使算法重复搜索某一区域,收敛时间变长.其次如目标点在空缺区域,同样会使收敛时间变长.文献[22]阐述了Halton序列采样点总数相同时,相邻节点距离越大其规划路径越平滑,收敛速度越快.所以,在RRT算法采样时选择 $X$ 轴质数为2、 $Y$ 轴质数为3的Halton采样序列.图1(d)为标准RRT算法使用的伪随机序列采样,随机采样点出现多处聚集、采样点分布不均等现象.图1(a)Halton序列与图1(d)伪随机序列相比,采样点分布均匀,采样点重复现象少,采样性能远优于伪随机序列采样.

### 2.2 目标偏置策略

本文的RRT算法引入了目标偏置策略,该策略为随机树的拓展提供了方向,降低了标准RRT算法在扩展中的随机性,并将目标偏置采样策略代替原来完全随机的采样方式.该方法定义:给定一控制常数 $\sigma$ , $\sigma$ 值取决于障碍物密集程度,若规划场景障碍物密集,则控制常数取较小值,算法探索未知环境能力强;如规划场景障碍物稀疏,则控制常数可取较大值,加快算法收敛速度,每次随机生成(0,1)的随机数 $p$ .

1) 如 $p < \sigma$ ,则令 $X_{\text{sample}} = X_{\text{goal}}$ ,  $X_{\text{new}}$ 沿 $\overrightarrow{X_{\text{near}} X_{\text{goal}}}$ 方向拓展固定步长 $\text{stepsize}$ .

反射可得到 $k$ 进制数

$$\varphi_k(m) = 0 \cdot b_0(m) b_1(m) \dots b_{i-1}(m) b_i(m). \quad (2)$$

最后将 $\varphi_k(m)$ 转化为十进制数,有

$$h_k(m) = \sum_{i=0}^i b_i(m)k^{-i-1}. \quad (3)$$

分别利用两种不同的序列在 $[0, 1] \times [0, 1]$ 地图上采样1000个点,图1(a)~图1(c)是Halton序列 $X$ 和 $Y$ 轴取不同质数为基数的采样表现;图1(d)是伪随机序列采样表现.

2) 如 $p \geq \sigma$ ,则令 $X_{\text{sample}} = X_{\text{rand}}$ ,根据随机采样函数在 $C_{\text{free}}$ 空间中采样,  $X_{\text{new}}$ 沿 $\overrightarrow{X_{\text{near}} X_{\text{rand}}}$ 方向拓展固定步长 $\text{stepsize}$ .

3) 如 $X_{\text{new}}$ 节点在拓展的过程没有与障碍物发生干涉,则将 $X_{\text{new}}$ 节点加入随机树 $T$ 中;否则舍弃 $X_{\text{new}}$ 节点,重新进行采点.

通过图2(a)与图2(b)的比较,标准RRT算法完全采用随机采样,目标导向性弱,随机性强,产生大量冗余节点,致使规划路径冗余路段多,规划时间长;而图2(b)所示Bias-RRT算法,加入了目标偏置策略,有更强的导向性,冗余节点与冗余路段较图2(a)的标准RRT算法有明显下降, Bias-RRT算法规划时间比标准RRT减少50%; Bias-RRT算法相比标准RRT算法的优势在于Bias-RRT可以一定概率向目标方向拓展,减少了大量无用节点,加快了Bias-RRT算法的搜索速度.

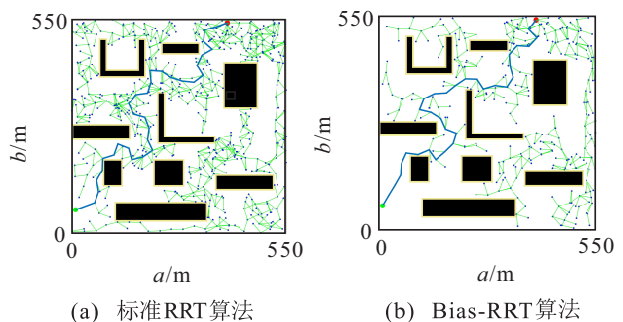


图2 标准RRT算法与Bias-RRT算法比较

2.3 候选点集策略

候选点集策略定义:为了达到筛选最优节点的目的,在采样阶段建立候选点集合,将一定数量的采样点加入候选点集中,当候选点集中采样点达到规定阈值 $\delta$ 时,则开始筛选节点,候选节点按照以下两种约束,并给予权重,按照 $Q$ 值大小进行排列:

1) 计算出候选点集合中每个采样点 $X_{\text{sample}}(i)$ 与 $X_{\text{goal}}$ 的欧氏距离,并对距离进行归一化处理,即

$$D_N = \text{distance}|X_{\text{sample}}(i) - X_{\text{goal}}|, \quad (4)$$

其中 $\text{distance}$ 为距离计算函数.

2) 计算出候选点集合中每个采样点转角角度 $\lambda_N$ ,并对 $\lambda_N$ 进行归一化处理,即

$$\lambda_N = \text{anglecalculation}(\text{RRTree}(i), X_{\text{near}}, \text{sample}(i)). \quad (5)$$

其中: $\text{anglecalculation}$ 为角度计算函数; $X_{\text{near}}$ 为随机树 $T$ 距离 $\text{sample}(i)$ 最近节点, $\text{RRTree}(i)$ 为 $X_{\text{near}}$ 父节点, $\text{sample}(i)$ 为随机采样点.

3) 按照 $X_{\text{sample}}(i)$ 所对应的 $Q$ 值从小到大进行排列, $Q = C_1\lambda_N + C_2D_N$ , $C_1$ 、 $C_2$ 分别为随机采样点到目标点角度 $\lambda_N$ 和距离 $D_N$ 的权重值,本文取 $C_1 = 0.3$ 、 $C_2 = 0.7$ ;按照 $X_{\text{sample}}(i)$ 所对应的 $Q$ 值从小到大进行排列;选取候选点集合 $M$ 筛选出 $M[i, 3] = 1$ 和 $Q_{\text{min}}$ 所对应的 $X_{\text{sample}}(i)$ 作为采样点.

$$M = \begin{bmatrix} \text{sample}(x_1) & \text{sample}(y_1) & 0 \text{ or } 1 & \lambda_1 & D_1 \\ \text{sample}(x_2) & \text{sample}(y_2) & 0 \text{ or } 1 & \lambda_2 & D_2 \\ \text{sample}(x_3) & \text{sample}(y_3) & 0 \text{ or } 1 & \lambda_3 & D_3 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \text{sample}(x_N) & \text{sample}(y_N) & 0 \text{ or } 1 & \lambda_N & D_N \end{bmatrix}. \quad (6)$$

其中: $M$ 为存储随机采样点信息的矩阵; $\text{sample}(x_N)$ 、 $\text{sample}(y_N)$ 代表随机采样点 $(x, y)$ 坐标值; $M[i, 3] = 0$ 表示新节点( $X_{\text{new}}$ )在该采样点方向上扩展时与障碍物发生干涉, $M[i, 3] = 1$ 代表该采样点方向可以拓展,为有效节点; $\lambda_N$ 表示归一化后的 $X_{\text{sample}}(i)$ 转角角度, $D_N$ 表示归一化后 $X_{\text{sample}}(i)$ 与 $X_{\text{goal}}$ 的距离.图3为候选点集示意图,绿色代表根节点,黑色代表随机树 $T$ 已有节点,水滴中灰色节点代表采集 $N$ 个候选节点 $X_{\text{sample}}(i)$ .

每次候选点集的样本量都会影响规划时间、路径长度、随机树节点等,选定图4(c)环境,对HRRRT算法进行50次测试,比较HRRRT的规划时间、路径长度、随机树 $T$ 节点数量、平均转角来选取最佳候选点集最佳样本量.如表1所示,HRRRT的候选点集最

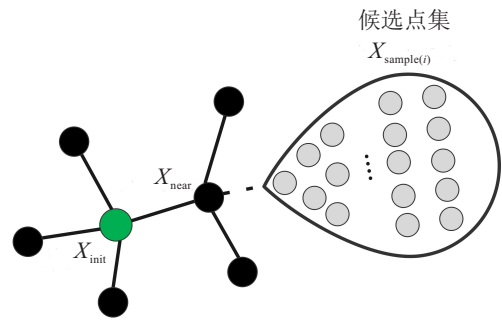
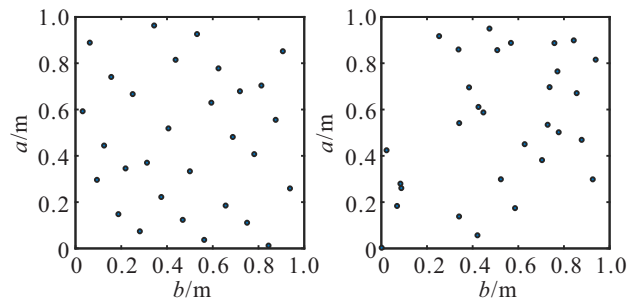
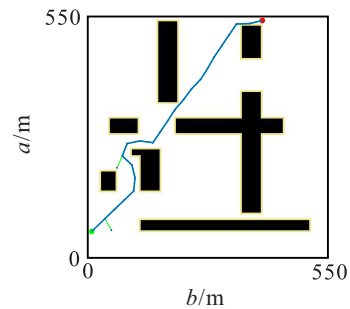


图3 候选点集示意图



(a) Halton序列(30个点) (b) 伪随机序列(30个点)



(c) 最佳候选点集规划路径

图4 不同序列采点分布及最佳候选点集规划路径

表1 候选点集最佳样本量测试

候选点集采样点 (个)	规划时间 (s)	路径长度 (m)	随机树节点 (个)	平均转角 (°)
$N = 5$	2.432	801.973	82.500	30.120
$N = 10$	2.474	929.866	83.200	39.827
$N = 15$	2.162	773.832	71.820	32.420
$N = 20$	1.765	815.117	60.200	32.925
$N = 25$	1.792	788.059	60.780	33.588
$N = 30$	1.072	744.619	36.620	27.842
$N = 35$	1.367	802.194	46.500	27.392
$N = 40$	1.250	761.858	42.440	27.202
$N = 45$	1.222	776.913	40.900	31.140
$N = 50$	1.268	764.212	41.260	25.997
$N = 55$	1.315	763.584	42.460	28.962

佳样本量为 $N = 30$ 个,此时HRRRT算法平均规划时间为1.07 s,路径长度为744.62 m,随机树节点总数为36.62个,平均转角为27.84°,综合数据表现最优;图4(a)为Halton序列采集30个点的分布情况;图4(b)为伪随机序列采集30个点的分布情况.伪随机序列采样点明显分布不均匀,存在部分区域空缺;Halton序列采样点分布均匀,无采样点聚集现象,所以HRRRT

算法选择  $N = 30$  作为最佳候选点集样本量. 图4(c)为HDRRT算法在最佳候选点集样本量下所规划出的路径.

图4中黑色以及浅黄色均为环境障碍物,浅黄色为膨胀区域(扩大1.5倍安全系数);蓝色为HDRRT所规划路径;

### 2.4 改进的Dijkstra算法

改进的Dijkstra算法加入了碰撞策略和角度限制策略. 碰撞策略在Dijkstra算法提取路径关键点时,避免路径与障碍物发生干涉;角度限制改善了路径质量,使得路径转角变小. 利用改进后的Dijkstra算法对原来生成的路径进行关键节点的提取,以剔除路径冗余节点,使得规划路径更加平滑,路径更短.

改进Dijkstra算法定义如下:

初始化: 节点编号  $0, 1, \dots, n - 1$ , 起点  $0 \leq s < n$ ; 起点  $d[s] = 0$ ; 其他节点  $d[i] = \inf[0 \leq i < n, i \neq s]$ , 邻接矩阵  $W = \emptyset$ , 节点集合  $V = \emptyset$ , 角度限制为  $45^\circ$ . 改进Dijkstra算法的邻接矩阵  $W$  中, 节点  $i$  至其余  $N - 1$  个节点通路, 如果符合碰撞策略和角度限制约束, 则将其放入邻接矩阵  $W$  中, 有效节点置为两点通路距离, 无效节点置为  $\inf$ . 例: 节点1至节点2和节点3的通路符合要求, 加入邻接矩阵  $W$ .

step 1: 建立邻接矩阵  $W$ , 假设总的路径点为  $N$  个, 检查第  $i$  个路径点与剩余  $N - 1$  个路径点路径是否与障碍物干涉; 角度是否符合角度限制, 如果符合则将第  $i$  个路径点与该点的路径代价放入邻接矩阵中, 如不符合则将该位置设置为  $\inf$ ; 如没有符合角度限制的节点, 则将最小转角角度且不与障碍物发生干涉的采样点放入邻接矩阵  $W$  中.

step 2: 循环  $N$  次.

step 2.1: 从邻接矩阵  $W$  中找到节点  $i \notin V$ , 且起始节点距离节点  $i$  最近;

step 2.2: 将  $i$  节点加入  $V$  集合中,  $V = V + 1$ ;

step 2.3: 对于所有满足  $j \notin V$  的边  $(i, j)$ , 节点  $j$  不属于  $V$  集合, 如有  $d[j] > d[i] + w(i, j)$ , 则更新  $d[j] = \min\{d[j], d[i] + W(i, j)\}$ ,  $d[j]$  表示起始节点与节点  $j$  的距离,  $W(i, j)$  表示节点  $i$  和节点  $j$  产生的边.

### 2.5 平滑处理

由于HDRRT算法规划出的路径在转折处呈现‘凸起’状, 移动机器人在转弯时可能会发生急转弯或者抖动, 对机器的预期寿命和整体结构会有影响. 所以, 对HDRRT算法规划出的路径进行平滑处理, 采用3次B样条曲线对路径进行平滑处理. 样条

定义: 共  $(N + 1)$  个点, 包含  $P_i(t)$  段, 路径每一段都受  $P_{i-1}, P_i, P_{i+1}, P_{i+2}$  控制; 其次, 3次样条为了美观与更加圆滑, 需满足  $C^2$  连续与曲线和坐标选择无关<sup>[23]</sup>. 图5为原始路径与优化后的路径对比, 未优化路径转折处曲率大, 优化后路径曲率明显减小, 移动机器人运动平稳, 不易抖动.

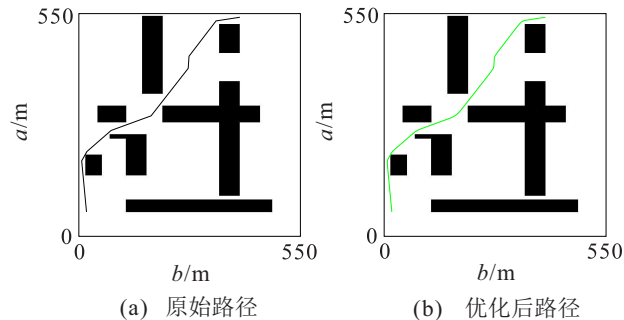


图5 3次B样条曲线优化示意图

$N$  次B样条曲线用多项式表示为

$$P(t) = \sum_{i=0}^n P_i F_{i,k}(t). \tag{7}$$

其中基函数

$$F_{i,k}(t) = \frac{1}{k!} \sum_{m=0}^{k-i} (-1)^m C_{k+1}^m (t + k - i - m)^k, \tag{8}$$

$$0 \leq t \leq 1, i = 0, 1, \dots, m.$$

$N = 3$  时, 式(7)可以表示成如下形式:

$$P(t) = \frac{1}{6} \begin{pmatrix} t^3 & t^2 & t & 1 \end{pmatrix} \begin{bmatrix} -1 & 3 & -3 & -1 \\ 3 & 6 & -3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix}. \tag{9}$$

### 2.6 HDRRT算法流程

初始化: 随机树  $T$ , 起始点  $X_{init}$ , 目标点  $X_{goal}$ , 固定步长  $stepsize$ , 误差值.

step 1: 随机树  $T$  初始化, 将起始点  $X_{init}$  加入随机树中.

step 2: 给定一控制常数  $\sigma$ , 每次随机生成  $(0, 1)$  的随机数  $p$ , 如  $p < \sigma$ , 则令  $X_{sample} = X_{goal}$ , 遍历随机树中的节点, 选出与  $X_{goal}$  最近节点  $X_{near}$  作为拓展的父节点.

step 3:  $X_{new}$  沿  $\overrightarrow{X_{near} X_{goal}}$  方向拓展固定步长  $stepsize$ .

step 4: 如果  $X_{new}$  节点在拓展的过程没有与障碍物发生干涉, 则将  $X_{new}$  节点加入随机树  $T$  中; 否则舍弃  $X_{new}$  节点, 重新进行采样.

step 5: 如果  $p \geq \sigma$ , 则利用Halton采样序列在  $C_{free}$  中一次性采集  $N (N = 30)$  个采样点.

step 6: 对采集的  $N$  个节点进行筛选, 其中与障碍

物发生干涉的采样点直接舍弃, 剩余节点按照对应的  $Q$  值从小到大进行排列, 选出  $Q$  值最小的  $X_{\text{sample}}(i)$  作为采样点;  $Q = C_1 \lambda_N + C_2 D_N$ .

step 7: 遍历随机树  $T$ , 筛选出距离  $X_{\text{sample}}(i)$  最近的节点  $X_{\text{near}}$  作为  $X_{\text{new}}$  的父节点.

step 8:  $X_{\text{near}}$  沿  $\overrightarrow{X_{\text{near}} X_{\text{sample}}(i)}$  方向拓展固定步长  $\text{stepsize}$ , 得到  $X_{\text{new}}$ .

$$X_{\text{new}} = X_{\text{near}} + \frac{\overrightarrow{X_{\text{near}} X_{\text{sample}}(i)}}{|\overrightarrow{X_{\text{near}} X_{\text{sample}}(i)}|} \text{stepsize}.$$

step 9: 计算  $d = \text{distance}(X_{\text{new}}, X_{\text{goal}})$ , 如果  $d <$  误差值, 则视为到达终点; 否则继续迭代.

step 10: 根据  $X_{\text{new}}$  与  $X_{\text{near}}$  的父子关系, 对路径进行回溯.

step 11: 利用改进的 Dijkstra 算法对路径关键节点进行二次提取, 并连接路线.

step 12: 利用 3 次 B 样条曲线对路径进行平滑处理.

### 3 HDRRT 性能验证

#### 3.1 HDRRT 算法 Matlab 仿真

算法仿真配置为: Matlab R2018a, 处理器 Intel(R) Core(TM) i7-10875H, CPU 主频 2.3 GHz, 内存 16GB. 为了在真实环境中 HDRRT 算法能够有更好的避障效果, 对障碍物进行膨化处理, 膨胀距离为移动机器人直径  $\times 1.5$  倍安全系数. Matlab 仿真环境中将移动机器人视为一个质点. 分别在地图 1 和地图 2 环境下对 HDRRT 算法进行 50 次测试, 测试数据取 50 次之平均值. 地图 1 和地图 2 分辨率均为  $550 \text{ m} \times 550 \text{ m}$ , 起始点均为  $[400 \text{ m}, 10 \text{ m}]$ , 目标点均为  $[10 \text{ m}, 490 \text{ m}]$ . 为了测试 HDRRT 算法性能, 加入了标准 RRT 算法和 Bias-RRT 算法进行对比实验. 图 6 和图 7 为不同算法性能比较, 黑色为障碍物, 黄色为膨胀体积, 蓝色为原始规划路径, 绿色主线为改进的 Dijkstra 算法提取关键节点的路径 (且 3 次 B 样条优化). 图 6(c)、图 7(c) 透明箭头处为提取的路径关键节点.

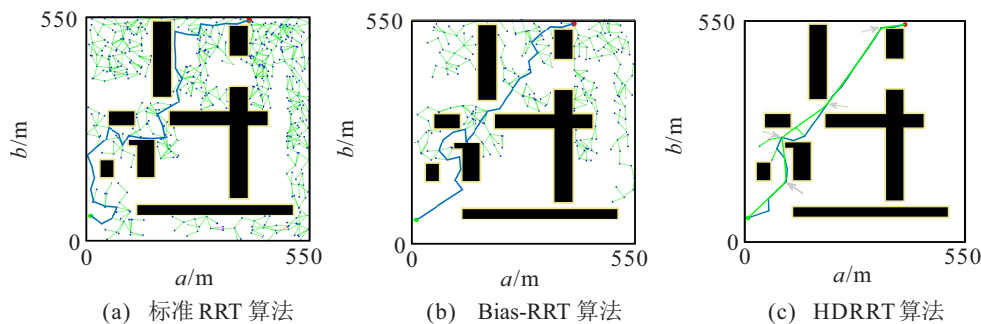


图 6 不同算法的性能比较 (地图 1)

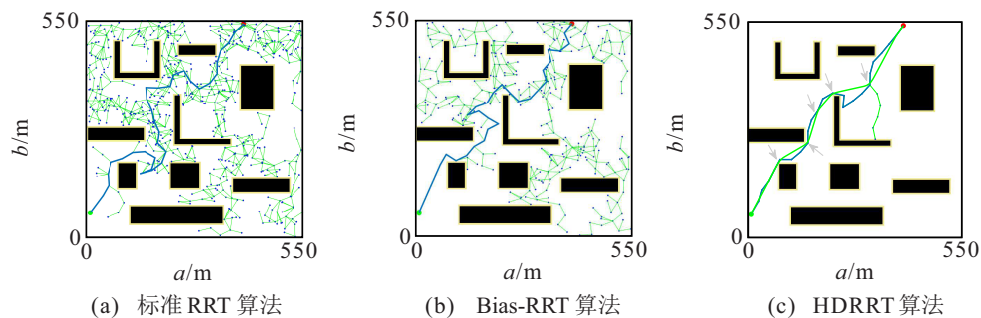


图 7 不同算法的性能比较 (地图 2)

由图 8、图 9 和表 2 分析可知, HDRRT 算法在地图 1 和地图 2 的规划时间分别为 0.905 s 和 1.092 s, 较标准 RRT 算法降低 91.30% 和 89.60%, 较 Bias-RRT 算法降低了 82.92% 和 81.51%; HDRRT 算法路径长度分别为 682.815 m 和 674.593 m, 较标准 RRT 算法降低 24.86% 和 29.36%, 较 Bias-RRT 算法降低 21.45% 和 22.70%; HDRRT 算法随机树节点数分别为 32.500 个和 38.420 个, 较标准 RRT 算法降低 87.86% 和 86.16%, 较 Bias-RRT 算法降低 79.06% 和 78.16%;

HDRRT 算法在地图 1 和地图 2 的候选点集策略中, 加入了角度权重, 较标准 RRT 算法, 路径平均转角减小了 34.68% 和 27.92%, 较 Bias-RRT 算法, 路径平均转角减小了 17.91% 和 14.94%. 据图 8(d) 和图 9(d) 分析, 标准 RRT 算法的转角角度在  $35^\circ \sim 55^\circ$  度之间波动, Bias-RRT 算法在  $25^\circ \sim 50^\circ$  之间波动, HDRRT 算法分别在  $20^\circ \sim 35^\circ$  和  $25^\circ \sim 35^\circ$  之间波动, 波动值与波动范围均小于标准 RRT 算法和 Bias-RRT 算法.

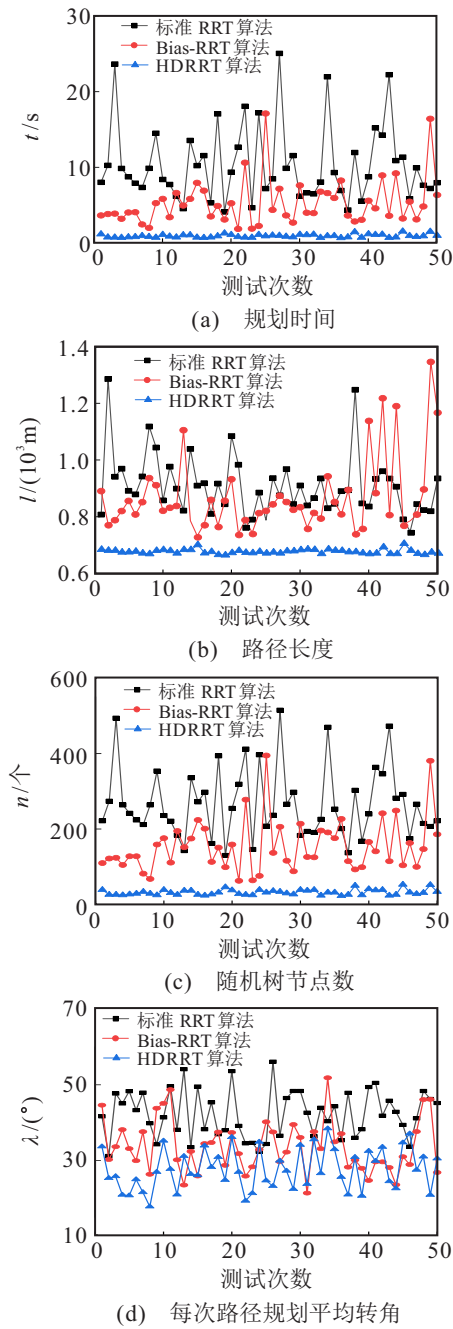


图8 不同算法的指标比较(地图1)

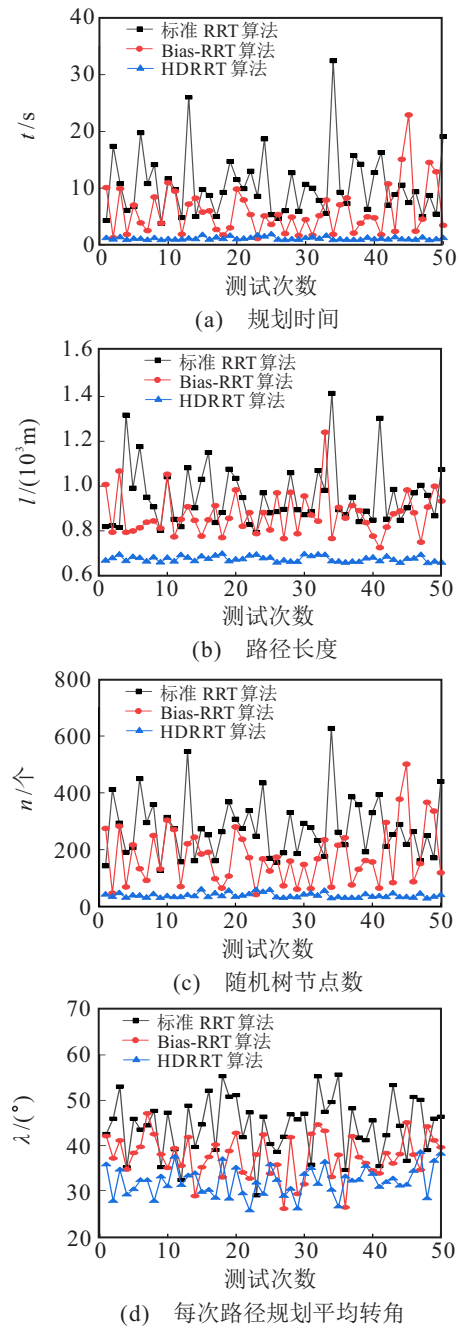


图9 不同算法的指标比较(地图2)

表2 图8及图9算法仿真数据

算法类型	平均时间/s	时间方差/s <sup>2</sup>	路径长度/m	长度方差/m <sup>2</sup>	随机树节点数/个	节点数方差	平均转角/(°)	转角方差
标准RRT(地图1)	10.410	25.332	908.721	11 346.080	267.740	8 719.872	42.360	38.100
Bias-RRT(地图1)	5.299	9.490	869.294	17 612.420	155.220	4 806.132	33.698	47.353
HDRRT(地图1)	0.905	0.045	682.815	64.007	32.500	55.010	27.662	28.553
标准RRT(地图2)	10.503	31.835	954.986	18 363.900	277.760	10 987.380	44.280	38.695
Bias-RRT(地图2)	5.907	18.389	872.774	9 114.700	175.920	9 923.234	37.526	22.018
HDRRT(地图2)	1.092	0.063	674.593	151.524	38.420	68.724	31.918	10.022

由标准RRT算法的规划时间、路径长度、随机树节点指标可知:标准RRT算法指标波动幅度最大,表明标准RRT算法随机性强,冗余路径长,冗余节点多;Bias-RRT算法增加目标偏置策略,在规划时间、路径长度、随机树节点数等指标较标准RRT算法

有所降低,但还是存在规划时间长、冗余节点和冗余路径的问题,波动幅度较标准RRT算法有所下降;HDRRT算法在地图1和地图2环境中,规划时间、路径长度、随机树节点数等指标远低于标准RRT算法和Bias-RRT算法,且HDRRT算法在每次路径规划时,

规划时间、路径长度、随机树节点数等指标波动幅度较小,方差小,表明HDRRT算法稳定性好,算法目的性强,剔除冗余节点减少了路径长度,规划时间短.

### 3.2 HDRRT算法ROS Gazebo-Rviz仿真

为了验证HDRRT算法在仿真场景中的避障性能,在ROS系统建立 $15\text{ m}\times 10\text{ m}\times 2.5\text{ m}$ 三维仿真环境,棕色及白色为障碍物,灰色区域为可行区域,如图10(b)所示;将障碍物区域进行膨化处理,膨胀距离为 $0.32\text{ m}$ ,是移动机器人直径的1.5倍,如图10(c)所

示.图10(c)通过激光雷达对指定环境建模,图11(a)~图11(c)分别为RRT算法、Bias-RRT算法和HDRRT算法规划路径.移动机器人为Turtlebot3-Burger,ROS版本melodic,ubuntu18.04操作系统,运行内存2GB,如图10(a)所示.参数设置:起始点位置 $(5.10\text{ m}, -5.05\text{ m}, 0\text{ m})$ ,目标点位置 $(-6.75\text{ m}, -44.15\text{ m}, 0\text{ m})$ ,地图分辨率 $0.05\text{ m}$ ,设定最大线速度 $0.55\text{ m/s}$ ,最大角速度 $57.0^\circ/\text{s}^2$ ,最大线加速度 $1\text{ m/s}^2$ ,最大角加速度 $20.0^\circ/\text{s}^2$ .

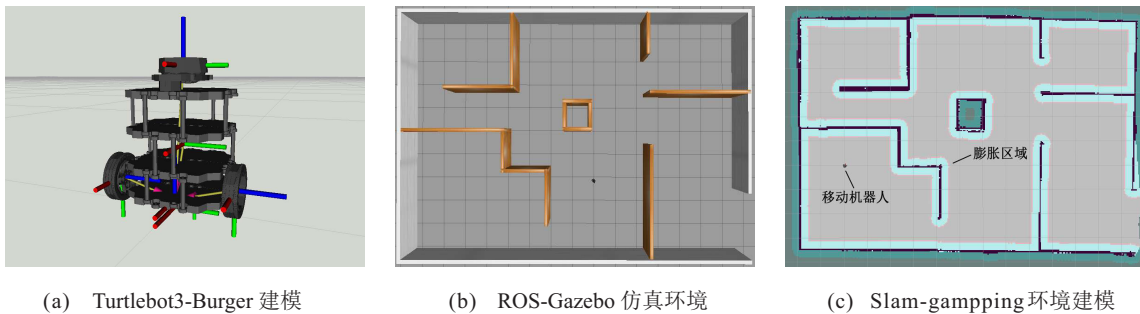


图10 Gazebo-Rviz环境建模

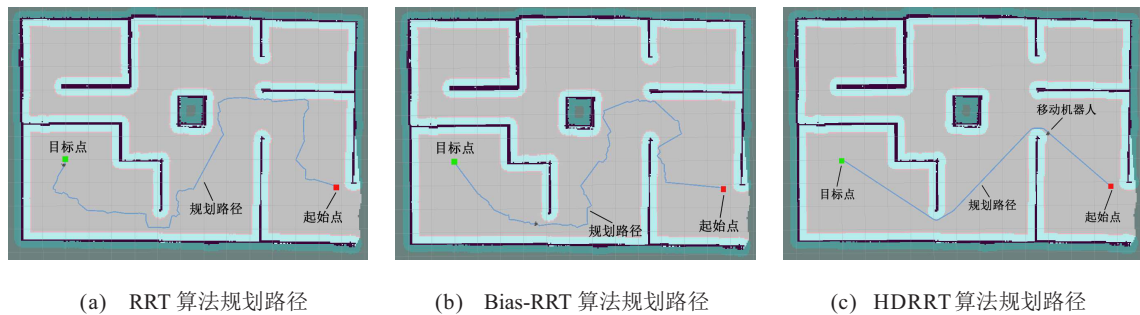


图11 Gazebo-Rviz仿真

表3数据为ROS仿真环境中对不同算法进行50次性能测试的平均数值.由表3数据可知,HDRRT算法在ROS仿真环境下,规划时间较标准RRT算法和Bias-RRT算法分别降低89.50%和80.69%,规划长度较标准RRT算法和Bias-RRT算法分别缩短29.77%和26.50%,仿真时间较标准RRT算法和Bias-RRT算法分别降低21.65%和18.04%.结合图11(a)~图11(c),HDRRT算法规划路径明显有所优化,消除了标准RRT算法冗余路段及冗余转角,HDRRT算法路径更加平滑,路径更短.综上所述,表明HDRRT算法能够在仿真环境中避开障碍物,能规划出全局最优路径,解决移动机器人在路径规划中的问题.

表3 不同算法性能测试数据

算法类型	规划时间/s	仿真时间/s	规划长度/m
标准RRT算法	8.720	41.340	22.189
Bias-RRT算法	4.744	39.517	21.202
HDRRT算法	0.916	32.390	15.584

## 4 结论

针对标准RRT算法所采用的伪随机序列算法造成采样点分布不均匀、不合理和冗余路径以及冗余节点问题,本文提出了 $X$ 轴质数=2, $Y$ 轴质数=3的Halton采样序列,利用候选点集策略和目标偏置策略对节点和路径进行优化;利用改进的Dijkstra算法对原始路径关键节点进行二次提取,有效地缩短了移动机器人运动路线;Matlab联合ROS系统,以Turtlebot3-Burger机器人路径规划为例,验证了HDRRT算法的优越性以及可行性,能够在复杂环境中快速、稳定地规划出最短路径;采用的Halton采样序列有效解决了标准RRT算法采用伪随机序列采样的缺点.

本文提出的HDRRT算法能够使得移动机器人快速、稳定地从起始点达到目标点,完成任务规划;本算法亦可推广至清洗机器人、无人船、陆上机器人等领域的路径规划问题.

## 参考文献(References)

- [1] Kavraki L, Latombe J C. Randomized preprocessing of configuration space for path planning: Articulated robots[C]. Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'94). Munich, 1994: 1764-1771.
- [2] Tsai C C, Huang H C, Chan C K. Parallel elite genetic algorithm and its application to global path planning for autonomous robot navigation[J]. IEEE Transactions on Industrial Electronics, 2011, 58(10): 4813-4821.
- [3] Guo X H, Ji M J, Zhao Z W, et al. Global path planning and multi-objective path control for unmanned surface vehicle based on modified particle swarm optimization (PSO) algorithm[J]. Ocean Engineering, 2020, 216: 107693.
- [4] Qing D S, Deng Q L, Li J J, et al. Split vehicle route planning with full load demand based on particle swarm optimization[J]. Control and Decision, 2021, 36(6): 1397-1406.
- [5] Wang X Y, Yang L, Zhang Y, et al. Robot path planning based on improved ant colony algorithm with potential field heuristic[J]. Control and Decision, 2018, 33(10): 1775-1781.
- [6] Zhong X Y, Tian J, Hu H S, et al. Hybrid path planning based on safe A\* algorithm and adaptive window approach for mobile robot in large-scale dynamic environment[J]. Journal of Intelligent & Robotic Systems, 2020, 99(1): 65-77.
- [7] 陈光荣, 郭盛, 王军政, 等. 凸优化与A\*算法结合的路径避障算法[J]. 控制与决策, 2020, 35(12): 2907-2914. (Chen G R, Guo S, Wang J Z, et al. Convex optimization and A-star algorithm combined path planning and obstacle avoidance algorithm[J]. Control and Decision, 2020, 35(12): 2907-2914.)
- [8] Rasekhipour Y, Khajepour A, Chen S K, et al. A potential field-based model predictive path-planning controller for autonomous road vehicles[J]. IEEE Transactions on Intelligent Transportation Systems, 2017, 18(5): 1255-1267.
- [9] LaValle S M. Rapidly-exploring random trees: A new tool for path planning[Z]. 1998.
- [10] Gammell J D, Srinivasa S S, Barfoot T D. Informed RRT: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic[C]. 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems. Chicago, 2014: 2997-3004.
- [11] Qi J, Yang H, Sun H X. MOD-RRT\*: A sampling-based algorithm for robot path planning in dynamic environment[J]. IEEE Transactions on Industrial Electronics, 2021, 68(8): 7244-7251.
- [12] Qureshi A H, Ayaz Y. Potential functions based sampling heuristic for optimal path planning[J]. Autonomous Robots, 2016, 40(6): 1079-1093.
- [13] Jeong I B, Lee S J, Kim J H. Quick-RRT\*: Triangular inequality-based implementation of RRT\* with improved initial solution and convergence rate[J]. Expert Systems with Applications, 2019, 123: 82-90.
- [14] Li Y J, Wei W, Gao Y, et al. PQ-RRT\*: An improved path planning algorithm for mobile robots[J]. Expert Systems with Applications, 2020, 152: 113425.
- [15] Kuffner J J, LaValle S M. RRT-connect: An efficient approach to single-query path planning[C]. IEEE International Conference on Robotics and Automation. San Francisco, 2000: 995-1001.
- [16] Mashayekhi R, Idris M Y I, Anisi M H, et al. Informed RRT\*-connect: An asymptotically optimal single-query path planning method[J]. IEEE Access, 2020, 8: 19842-19852.
- [17] 郑惠江. 在机检测中曲面拓扑特征重建和检测点分布关键技术研究[D]. 天津: 天津大学, 2010. (Zheng H J. Research on the key technology of the topological feature reconstruction and sampling point distribution in OMV[D]. Tianjin: Tianjin University, 2010.)
- [18] Barraquand J, Kavraki L, Latombe J C, et al. A random sampling scheme for path planning[J]. The International Journal of Robotics Research, 1997, 16(6): 759-774.
- [19] Park B, Chung W K. Efficient environment representation for mobile robot path planning using CVT-PRM with Halton sampling[J]. Electronics Letters, 2012, 48(22): 1397-1399.
- [20] Geraerts R, Overmars M H. A comparative study of probabilistic roadmap planners[C]. Algorithmic Foundations of Robotics. Berlin, Heidelberg, 2004: 43-57.
- [21] Bhat C R. Simulation estimation of mixed discrete choice models using randomized and scrambled Halton sequences[J]. Transportation Research—Part B: Methodological, 2003, 37(9): 837-855.
- [22] Velagi J, Delimustafi D, Osmankovi D. Mobile robot navigation system based on Probabilistic Road Map (PRM) with Halton sampling of configuration space[C]. 2014 IEEE 23rd International Symposium on Industrial Electronics. Istanbul, 2014: 1227-1232.
- [23] Juhász I, Hoffmann M. Constrained shape modification of cubic B-spline curves by means of knots[J]. Computer-Aided Design, 2004, 36(5): 437-445.

## 作者简介

钟华庚(1996—), 男, 硕士生, 从事移动机器人路径规划、机器人控制的研究, E-mail: zhuageng@foxmail.com;  
 罗高生(1975—), 男, 讲师, 博士, 从事水下机器人及其运动控制的研究, E-mail: gsluo@shou.edu.cn;  
 王芳(1979—), 女, 副研究员, 从事智能装备、海洋技术等研究, E-mail: wangfang@shou.edu.cn;  
 周正权(1982—), 男, 工程师, 硕士, 从事智能装备、海洋工程的研究, E-mail: zhoushq@cnooc.com.cn;  
 陈祎(1989—), 女, 工程师, 从事智能装备、海洋工程的研究, E-mail: chenyl7@cnooc.com.cn.