基于改进RRT*算法的智能车路径规划研究*

关志伟　路晓宇　赵宇

（天津中德应用技术大学汽车与轨道交通学院，天津 300350）

【摘要】针对传统RRT*算法在路径探索过程中存在的效率低、随机采样点质量差和路径质量不佳等问题，提出一种结合改进人工势场与自适应步长双向扩展机制的方法。通过自适应引力系数和双重斥力，优化路径引导与避障；采用动态步长调整策略和双向树协同搜索机制，减少冗余节点并加速收敛；利用贪婪算法去除冗余点，保证路径的平滑性和连续性；引入基于环境密度的自适应目标偏置策略，增强目标导向性和探索效率。试验结果表明：在复杂环境中，相较于传统RRT*及其他改进算法，ABF-RRT*平均路径长度缩短17.7%，平均节点数减少90.6%，运行时长缩短95.5%，具有稳定性与优越性。

主题词：路径规划　RRT*算法　双向扩展机制　改进人工势场　自适应步长　自适应目标偏置

中图分类号：TP391.9　　**文献标志码：**A　　**DOI:** 10.19620/j.cnki.1000-3703.20250142

Research on Path Planning for Intelligent Vehicles Based on Improved RRT* Algorithm

Guan Zhiwei, Lu Xiaoyu, Zhao Yu

(School of Automotive and Rail Transit, Tianjin Sino-German University of Applied Sciences, Tianjin 300350)

【Abstract】To address the issues of low efficiency, poor quality of random sampling points, and suboptimal path quality in the traditional RRT* algorithm during path exploration, this paper proposes a method that integrates an improved artificial potential field formulation with an adaptive step-size bidirectional expansion mechanism. By introducing an adaptive attraction coefficient and dual repulsion forces, the proposed method optimizes path guidance and obstacle avoidance. A dynamic step-size adjustment strategy and a bidirectional tree cooperative search mechanism are employed to reduce redundant nodes and accelerate convergence. A greedy algorithm is used to remove redundant nodes, ensuring path smoothness and continuity. Additionally, an adaptive goal-biasing strategy based on environmental density is introduced to enhance goal orientation and exploration efficiency. Experimental results demonstrate that, compared with the traditional RRT* algorithm and other improved algorithms, the proposed ABF-RRT* algorithm reduces the average path length by 17.7%, decreases the average number of nodes by 90.6%, and shortens the average running time by 95.5% in complex environments, verifying its stability and superiority.

Key words: Path planning, RRT* algorithm, Bidirectional expansion mechanism, Improved artificial potential field, Adaptive step size, Adaptive goal biasing

【引用格式】关志伟, 路晓宇, 赵宇. 基于改进RRT*算法的智能车路径规划研究[J]. 汽车技术, 2026(8): 1-11.

Guan Z W, Lu X Y, Zhao Y. Research on Path Planning for Intelligent Vehicles Based on Improved RRT* Algorithm[J]. Automobile Technology, 2026(8): 1-11.

1　前言

智能车辆路径规划的核心任务是结合道路环境与障碍物约束，为自车求解一条无碰撞、满足车辆运动特性的最优行驶路径。常见的路径规划算法包括A*[1]、Dijkstra[2]、人工势场法[3]、RRT等。其中，RRT类算法具有渐进最优性和良好扩展性，但其在搜索效率与路径质量方面仍存在一定局限性。

针对传统RRT类算法的不足，在优化采样方法方面，陈国军等[4]通过调整随机采样方法，提高了扑翼飞行器的采样效率，但该方法在障碍物密集区域仍易产生冗余点；Janjos等[5]使用4次样条插值优化路径平滑度，

*基金项目：天津市教委科研计划项目（2019ZD20）。

但计算复杂度较高；周秦源等[6]结合自适应目标导向策略、区域采样备选策略和贪婪剪枝策略，显著提高了路径搜索成功率，但路径平滑方面仍有待提升。

在提升计算与搜索效率方面，鲁宇明等[7]通过改进约束采样空间和引导策略，但该方法在非椭圆区域的改进效果有限；王昊等[8]结合深度强化学习与RRT*，增强了环境适应性，但训练成本较高；盛兆康等[9]利用综合评价函数减少无效采样；罗济雨等[10]采用分区采样拒绝策略减少冗余节点；左国玉等[11]使用启发式函数评估采样点成本，加速模型收敛；Fan等[12]通过目标偏向策略加速收敛；Ganesan等[13]引入非均匀与均匀采样提升效率。朱敏等[14]结合动态偏置策略、自适应人工势场权重和双向直连检测改进RRT-Connect，提升了路径规划的收敛速度、避障能力和路径平滑性；葛超等[15]通过4棵树扩展、动态步长和路径优化策略提高路径规划效率和质量。上述方法在一定程度上提升了算法性能，但模型仍会陷入局部最优和参数敏感性增加等问题。

因此，针对传统RRT*探索效率慢、采样点质量较低和路径质量差等问题，本文提出基于改进人工势场与自适应双向扩展机制的快速探索随机树（Adaptive Bi-Field Rapidly-exploring Random Tree Star，ABF-RRT*）算法。通过引入双重斥力解决因目标点附近障碍物过多导致斥力大于引力且无法到达目标点的局限性，使用自适应步长避免局部最优，同时，利用协同优化策略加速收敛，动态调整目标偏置概率增强目标导向性，从而实现高效的路径搜索和高质量的路径生成。

2 相关工作

2.1 RRT*算法

RRT*是一种渐进最优的路径规划算法，在原有RRT的基础上增加了重新连接步骤，使新加入的节点与其周围的节点建立连接关系，从而改善了最终路径的质量。当搜索到一条可达路径，从终点q_{goal}回溯到起点q_{start}，并记录路径中所有节点，形成最终路径。RRT*的扩展方式与潜在父节点如图1所示，其伪代码如图2所示。

2.2 APF-RRT*算法

APF-RRT*算法结合了人工势场（Artificial Potential Field，APF）法和RRT*算法的优点，旨在更高效地解决机器人路径规划问题。通过在随机树的扩展过程中引入人工势场的概念，APF-RRT*能够更好地避开障碍物，并且有效利用吸引力和排斥力来指导随机树的生长方向，从而避免局部最优解的问题。一旦找到一条可达路径，从终点q_{goal}开始回溯到起点q_{start}，记录路径中所有节点，形成最终路径。APF-RRT*的受力原理如图3所示，其伪代码如图4所示。

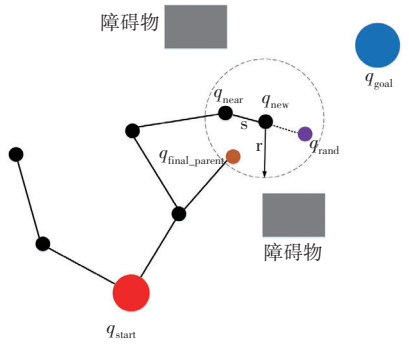


图1 RRT*的扩展方式与潜在父节点示意

1：Initialize tree T with q_{start}
//初始化起点树T，将起点q_{start}设为根节点
2：**for** i = 1 to MaxIterations **do**
//主循环迭代上限为MaxIterations
3：q_{rand}= RandomSample(M);
//在地图M中随机采样得到随机点q_{rand}
4：q_{near}= NearestNeighbor(T,q_{rand});
//在树T查找距离q_{rand}最近的节点q_{near}
5：q_{new}= Steer(q_{near},q_{rand});
//沿q_{rand}方向固定步长扩展生成新节点q_{new}
6：**if** CollisionFree(q_{near},q_{new}) **then**
//检测q_{near}与q_{new}间的连线是否与障碍物碰撞
7：q_{near_nodes}=NearNodes(T,q_{new},radius);
//在树T中搜索q_{new}周围半径radius内的邻近节点
8：q_{min}= ChooseParent(q_{near_nodes},q_{new});
//从邻近节点中选择代价最小节点作为q_{new}父节点
9：AddNode(T,q_{new},q_{min});
//将q_{new}及其选定父节点q_{min}加入树T中
10：Rewire(T,q_{new},q_{near_nodes}); //重布线优化
11：**if** Distance(q_{new},q_{goal})<threshoud **then**
//判断新节点q_{new}是否进入目标点q_{goal}的阈值范围
12：return Path(T,q_{start},q_{goal});
//从q_{start}回溯至q_{goal}，生成并返回最终路径
13：**end if**;
14：**end if**;
15：**end for**
16：return NoPathFound //未找到路径

图2 RRT*算法流程

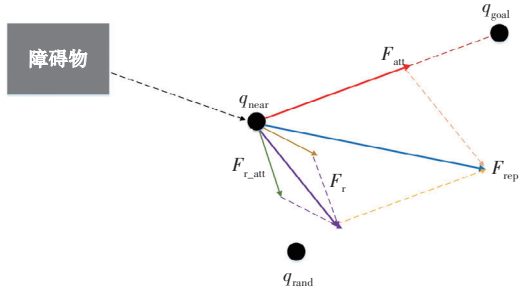


图3 APF-RRT*受力原理

```
Initialize tree T with q_start
//初始化起点树 T，将起点 q_start 设为根节点
for i = 1 to MaxIterations do
//主循环迭代上限为 MaxIterations
    q_rand = RandomSample(M);
    //在地图 M 中随机采样得到随机点 q_rand
    q_near = NearestNeighbor(T, q_rand);
    //在树 T 查找距离 q_rand 最近的节点 q_near
    F_att = CalculateAttraction(q_near, q_goal);
    //计算目标点 q_goal 对 q_near 的引力向量 F_att
    F_rep = CalculateRepulsion(q_near, obstacles);
    //计算障碍物对 q_near 的斥力向量 F_rep
    F_total = F_att + F_rep; //计算合力向量 F_total
    q_new = q_near + StepSize * F_total;
    //沿合力方向扩展生成新节点 q_new
    if CollisionFree(q_near, q_new) then
    //检测 q_near 与 q_new 间的连线是否与障碍物碰撞
        q_near_nodes = NearNodes(T, q_new, radius);
        //在树 T 中搜索 q_new 周围半径 radius 内的邻近节点
        q_min = ChooseParent(q_near_nodes, q_new);
        //从邻近节点中选择代价最小节点作为 q_new 父节点
        AddNode(T, q_new, q_min);
        //将 q_new 及其选定父节点 q_min 加入树 T 中
        Rewire(T, q_new, q_near_nodes) //重布线优化
        if Distance(q_new, q_goal) < threshoud then
        //判断新节点 q_new 是否进入目标点 q_goal 的阈值范围
            return Path(T, q_start, q_goal);
        //从 q_start 回溯至 q_goal，生成并返回最终路径
        end if;
    end if;
end for
return NoPathFound //未找到路径
```

图4　APF-RRT*算法流程

3　改进RRT*算法

3.1　改进人工势场法

为解决传统的人工势场（Artificial Potential Field，APF）法中，固定引力可能会使车辆与目标点较远时移动过慢，接近目标点时移动过快，从而难以精确停靠在目标点的问题，本文引入自适应引力系数$k(x)$，可根据车辆与目标点的距离动态调整，从而实现更合理的路径规划：

$$k(x) = k_{\max} \cdot \mathrm{e}^{-\frac{d(x,x_{\text{goal}})}{d_{\text{scale}}}} \tag{1}$$

式中：$k_{\max}$为最大吸引力系数，x、x_{goal}分别为当前点和目标点的位置坐标，$d(x,x_{\text{goal}})$为x与目标点x_{goal}间的距离，d_{scale}为缩放因子。

当$d(x,x_{\text{goal}})$较大时，车辆会向目标点运动更快。引入自适应引力系数后，总引力$\boldsymbol{F}_{\text{att}}$为目标点$\boldsymbol{F}_{\text{att_goal}}$与随机点$\boldsymbol{F}_{\text{att_rand}}$的向量和：

$$\begin{cases} \boldsymbol{F}_{\text{att}} = \boldsymbol{F}_{\text{att_goal}} + \boldsymbol{F}_{\text{att_rand}} \\ \boldsymbol{F}_{\text{att_goal}} = -k_{\text{att}} \cdot (x_{\text{goal}} - x) \\ \boldsymbol{F}_{\text{att_rand}} = -k_{\text{rand}} \cdot (x_{\text{rand}} - x) \end{cases} \tag{2}$$

式中：k_{att}、k_{rand}为引力强度系数，x_{rand}为随机采样点的位置坐标。

鉴于车辆接近障碍物时，传统单一斥力模型可能存在紧贴障碍物的情况，为此，本文引入传统斥力$\boldsymbol{F}_{\text{rep1}}$和附加斥力$\boldsymbol{F}_{\text{rep2}}$，确保生成的路径与障碍物保持一定的安全距离。则优化后的总斥力为：

$$\begin{cases} \boldsymbol{F}_{\text{rep}} = \boldsymbol{F}_{\text{rep1}} + \boldsymbol{F}_{\text{rep2}} \\ \boldsymbol{F}_{\text{rep1}} = \sum_i \left(k_{\text{rep}} \cdot \left(\dfrac{1}{d} - \dfrac{1}{r} \right) \cdot \dfrac{\rho_{g^2}}{d^2} \cdot \dfrac{x-p}{d} \right) \\ \boldsymbol{F}_{\text{rep2}} = \dfrac{n}{2} \cdot k_{\text{rep}} \cdot \left(\dfrac{1}{d} - \dfrac{1}{r} \right) \cdot \dfrac{1}{\rho_{\text{g}}^n} \cdot \dfrac{x-p}{d} \end{cases} \tag{3}$$

式中：k_{rep}为最大斥力系数，d为x与障碍物的距离，r为斥力的影响范围，ρ_{g}为x与目标点的距离，p为矩形上的最近点，n为影响斥力的指数参数。

3.2　自适应步长双向扩展机制

传统单向搜索方法的路径生成时间周期长，且传统双向扩展机制的步骤较简单，仅从起点q_{start}和目标点q_{goal}同时向对方扩展。在复杂环境中的扩展步长和引导方式容易陷入局部最优，无法灵活适应复杂的环境。因此，结合前述的改进人工势场法，引入自适应步长调整及双向树间协同优化策略。

在双向扩展中，步长根据树间距离d_{tree}和障碍物密度c调整，其中，障碍物密度由蒙特卡罗方法计算得到：

$$\begin{cases} S = S_{\min} + S_{\max} \cdot \mathrm{e}^{-\varepsilon c} \cdot \left(1 - \dfrac{d_{\text{tree}}}{d_{\max}} \right) \\ S_{\max},\ S_{\min} \in (0.5, 3.0) \\ \varepsilon \in (0.1, 1.0) \end{cases} \tag{4}$$

式中：$S_{\max}$、$S_{\min}$为步长的最大值和最小值，ε为步长调节参数，$d_{\max}$为最大树间距离。

在本文环境下，当步长小于0.5时，节点激增，搜索时间显著增加；当步长大于3.0时，避障成功率明显下降。ε使步长在障碍物密集区域能够平滑收缩至最小值附近、在空旷区域保持较大值。

扩展步长如图5所示，每次扩展时需考虑距离最近节点的引力，还需考虑树间的距离、障碍物位置以及当前搜索状态。假定目标引力和障碍物斥力合力为$\boldsymbol{F}_{\text{total}}$，当前扩展节点为：

$$q_{\text{new}} = q_{\text{current}} + S \cdot \boldsymbol{F}_{\text{total}} \tag{5}$$

式中：S为步长，q_{current}为当前节点的位置坐标。

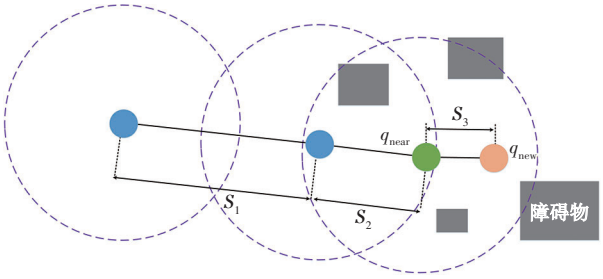


图5　扩展步长的调整

在扩展中，为了加快两棵树的收敛，本文引入双向树协同优化策略，其原理如图6所示。当$d_{tree}<\delta$时，其中，δ为设定阈值，可直接连接两棵树并生成路径；反之，可根据目标点的引力和障碍物的斥力进行动态调整。协同优化策略为：

$$\boldsymbol{F}_{total}=\omega_1\boldsymbol{F}_{att}+\omega_2\boldsymbol{F}_{rep}+\omega_3\boldsymbol{F}_{tree} \tag{6}$$

式中：$\boldsymbol{F}_{tree}=\dfrac{(q_{goal}-q_{start})}{\|q_{goal}-q_{start}\|}$为两棵树的距离引导力，$\omega_1$、$\omega_2$为根据当前环境动态调整的权重系数，$\omega_3$为根据引导力动态调整的权重系数。

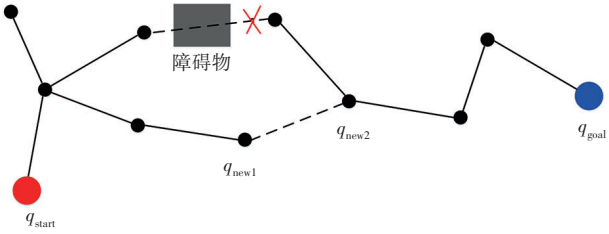


图6　协同优化原理

将改进人工势场与自适应步长双向扩展机制与RRT*算法融合，形成自适应双向APF-RRT*算法，其伪代码如图7所示。

当两棵树成功连接或达到预设的迭代次数时，终止算法，输出最终的路径作为结果。自适应双向APF-RRT*算法中$q_{nearest}$的受力分析如图8所示，流程如图9所示。

3.3　自适应目标偏置策略

为了解决探索树缺乏目标导向性的问题，本文提出一种基于环境密度的自适应目标偏置策略，根据环境中的障碍物密度和算法的进展状态，动态调整目标偏置概率，以优化路径规划中的树扩展行为。假定当前环境中障碍物密度为c，路径的扩展进度为$p_{progress}$，则目标偏置概率为：

$$\begin{cases} p_{goal\ bias}=\alpha\cdot\left(1-\dfrac{C}{C_{max}}\right)\cdot(1-p_{progress})+\beta\cdot\dfrac{p_{progress}}{L_{max}} \\ p_{progress}=\dfrac{\|q_{current}-q_{start}\|}{\|q_{goal}-q_{start}\|},\quad p_{progress}\in[0,1] \end{cases} \tag{7}$$

式中：α、β分别为控制环境密度和路径进展对目标偏置影响的调节因子，C_{max}为障碍物密度的最大值，L_{max}为路径的最大扩展距离，$p_{progress}$=0为当前节点处于起点位置。

```
Initialize two trees T_start and T_goal with q_start and q_goal
//初始化双树：T_start以q_start为根节点，T_goal以q_goal为根节点
while not MaxIterations reached do //主循环迭代
    q_rand=RandomSample(M)
    //在地图M中随机采样得到随机点q_rand
    if rand( ) < goal_bisa_prob then
    //以概率goal_bisa_prob决定是否将采样点设为目标点
        q_rand=q_goal //采样点直接设为目标点q_goal
    end if
    if iter % 2 == 0 then //交替扩展双树
        q_near= NearestNeighbor(T_start,q_rand)
        //在起点树T_start查找最近节点q_rand
    else
        q_near= NearestNeighbor(T_goal,q_rand)
        //在目标树T_goal查找最近节点q_near
    end if
    F_att= CalculateAttraction(q_near,q_goal)
    //计算目标点q_goal对q_near的引力
    F_rep= CalculateRepulsion(q_near,obstacles)
    //计算障碍物obstacles对q_near的斥力
    F_total=F_att+F_rep //计算合力向量F_total
    StepSize = CalculateAdaptiveStepSize(q_near,obstacles)
    //根据当前节点附近障碍物密度计算步长StepSize
    q_new= q_near+StepSize*F_total
    //沿合力方向扩展生成新节点q_new
    if CollisionFree(q_near,q_new) then
    //检查q_near到q_new的连线是否避开障碍物
        if iter % 2 == 0 then //扩展起点树
            AddNode(T_start,q_start,q_goal) //加入起点树T_start
        else
            AddNode(T_goal,q_new,q_min) //将q_new加入目标树T_goal
        end if
        if  Distance(T_start,T_goal)< threshoud then //连通检测
            return Path(T_start,T_goal) //返回完整路径
        end if
    end if
end while
return NoPathFound //未找到路径
```

图7　自适应双向APF-RRT*算法流程

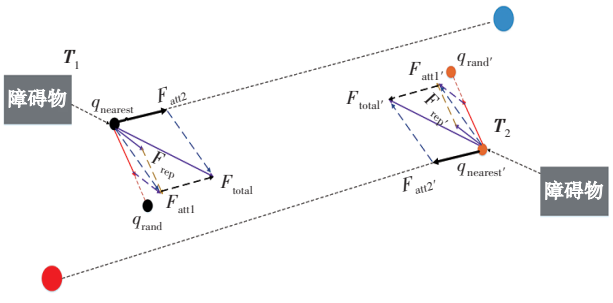


图8　$q_{nearest}$受力分析

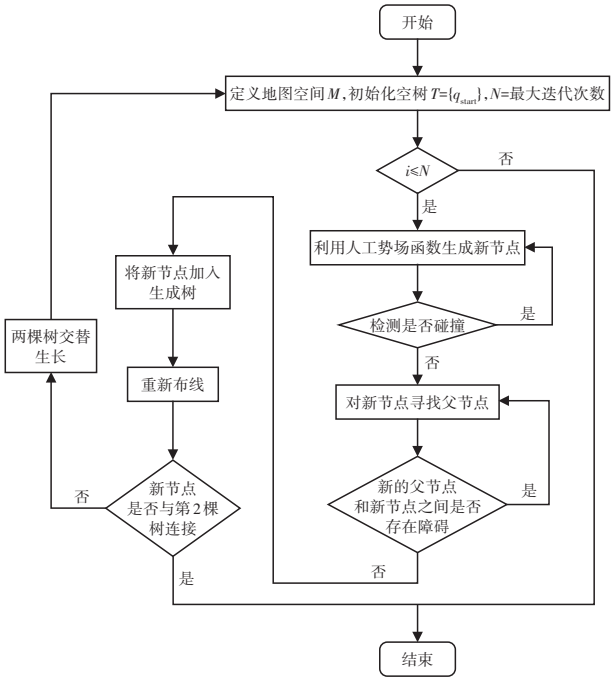


图9　自适应双向APF-RRT*流程

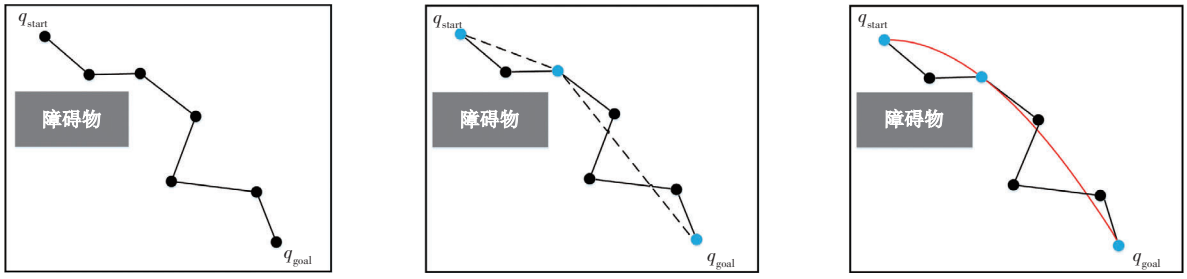
3.4　路径平滑策略

鉴于贪婪算法可通过逐步选择距离目标点最近的节点，去除冗余节点，从而缩短路径长度并减少转折次数，故在评估路径质量时，引入成本函数，得到：

$$Cost(q_i,q_j) = u_1 \cdot Dist(q_i,q_j) + u_2 \cdot Turn(q_i,q_j) + u_3 \cdot Safe(q_i,q_j) \tag{8}$$

式中：$Dist(q_i,q_j)$、$Turn(q_i,q_j)$、$Safe(q_i,q_j)$分别为节点q_i与q_j间的欧氏距离、转折角度、路径与障碍物的最小距离，u_1、u_2和u_3分别为路径长度、路径平滑度和安全距离的权重系数。

在本文算法ABF-RRT*中引入贪婪算法，结合成本函数对路径进行优化，生成一条平滑且符合智能车运动学约束的路径，如图10所示。处理前的路径包含许多非必要转折点，增加了行驶距离和时间成本。通过贪婪算法去除冗余节点，使用成本函数进一步优化路径平滑度和安全性，保证了智能车路径规划的安全性和行驶流畅性。

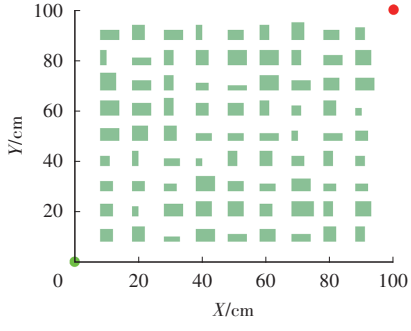


(a)原始路径　(b)贪婪算法　(c)贪婪算法引入成本函数路径

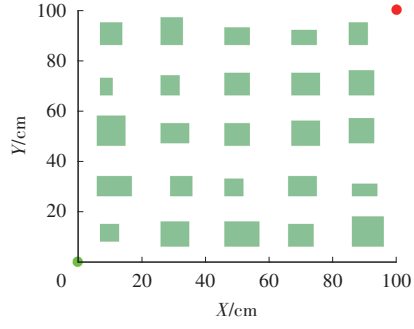
图10　平滑路径结果

4　仿真试验与结果分析

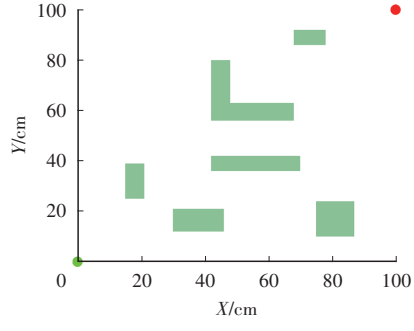
为了验证本文算法性能，使用MATLAB进行仿真测试，并与无路径平滑策略的ABF-RRT*、APF-RRT*、RRT-Connect和RRT*进行对比。本文构建4种不同障碍物拥挤程度的试验环境，长、宽均为100 cm，如图11所示。其中，环境1和环境2中障碍物较为复杂，环境3为L型，环境4为简单通道型。4种场景的起始点坐标均为(0,0)，目标点坐标均为(100,100)。



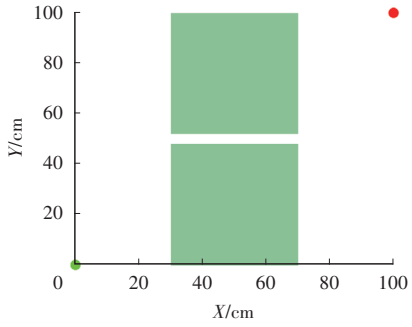
(a)环境1



(b)环境2



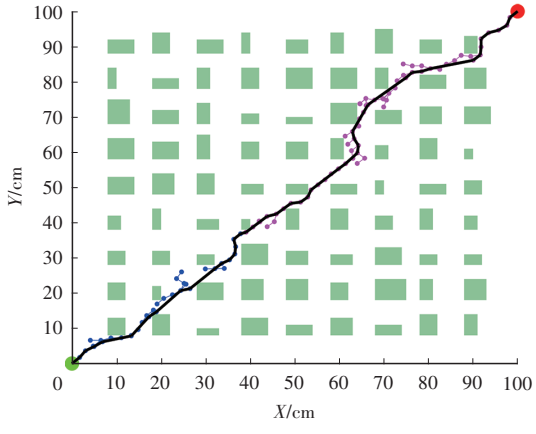
(c)环境3



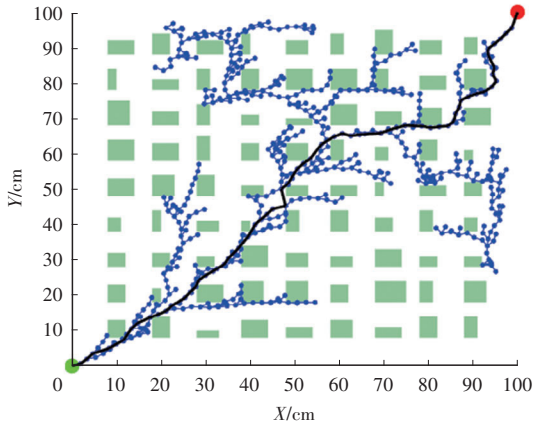
（d）环境4

图11　试验环境

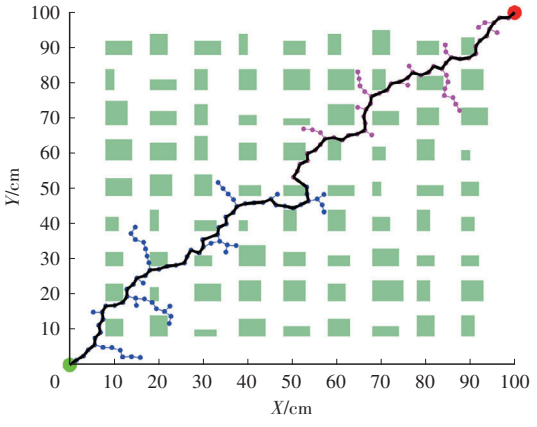
在初始条件下，对4种算法进行性能评估，在不同环境中各自执行15次测试，每种算法的最大迭代次数为3 000次，结果如图12所示，本文算法的收敛曲线变化趋势如图13所示，各算法的仿真数据如表1所示。在15次测试中，ABF-RRT*的路径规划更接近最优解，同时减少了对最优路径贡献较小的节点的选取。APF-RRT*利用障碍物、起点和目标点的信息引导探索树避开障碍并指向目标点，虽然减少了无用点生成，但因障碍物群的斥力作用，该算法会偏离理想路径或陷入障碍物密集区域，从而影响其收敛速度和路径质量。相较于APF-RRT*，RRT-Connect的探索速度有所提升，但其无目的探索会产生非必须节点，从而导致路径质量不佳。



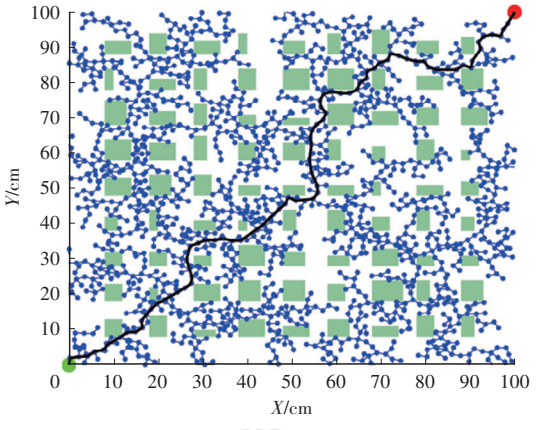
ABF-RRT*



APF-RRT*

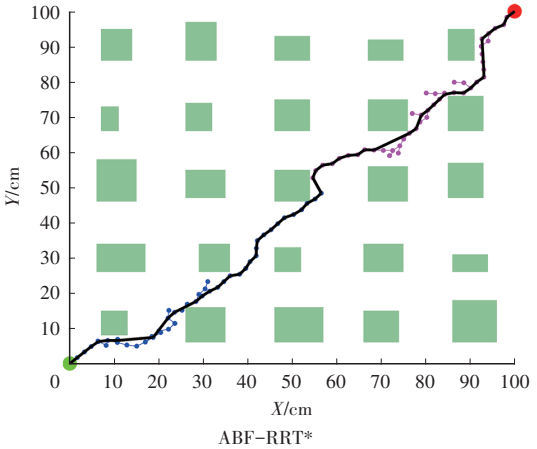


RRT-Connect

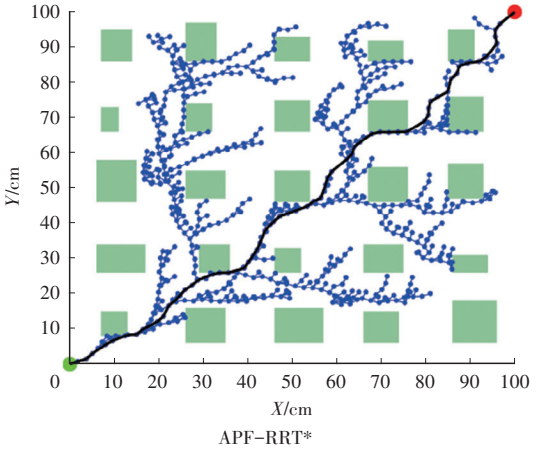


RRT*

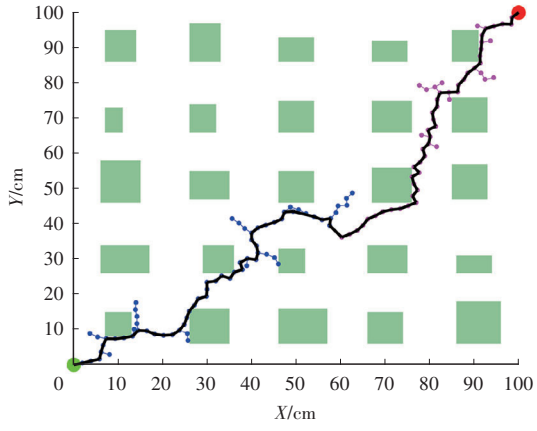
（a）环境1



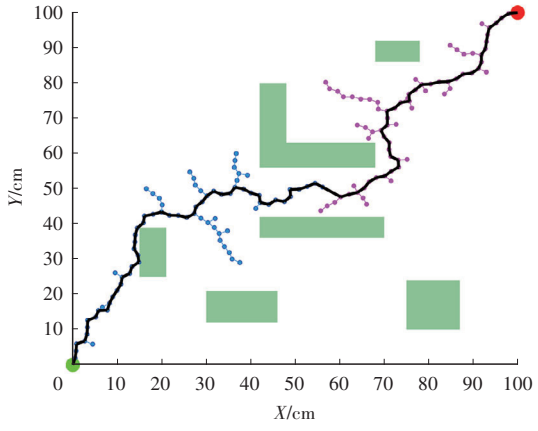
ABF-RRT*



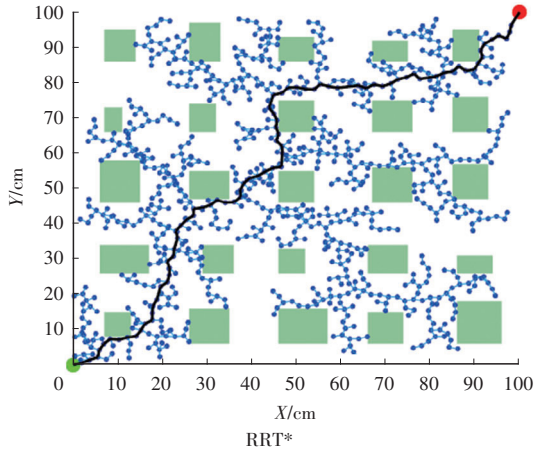
APF-RRT*



RRT-Connect

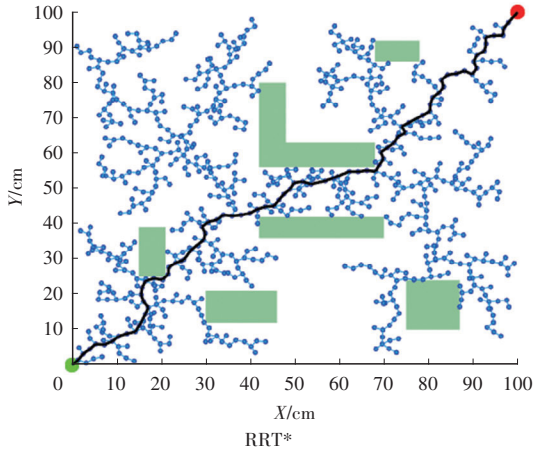


RRT-Connect



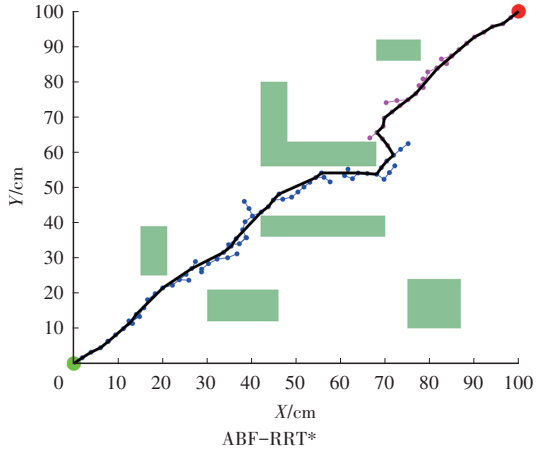
RRT*

(b)环境2

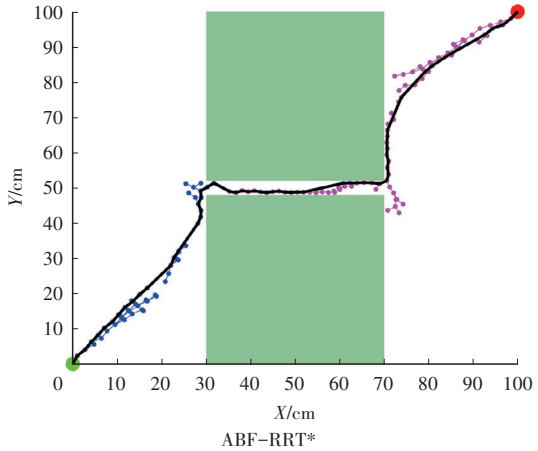


RRT*

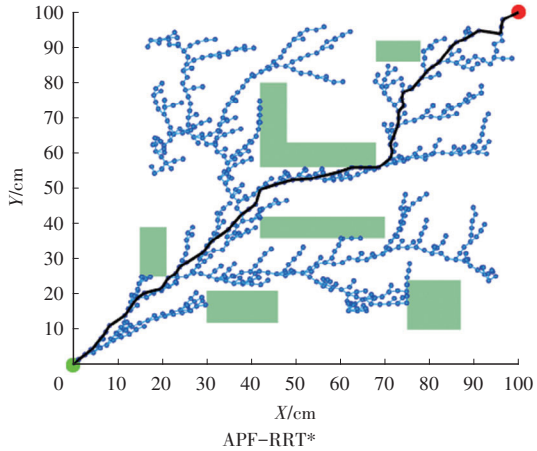
(c)环境3



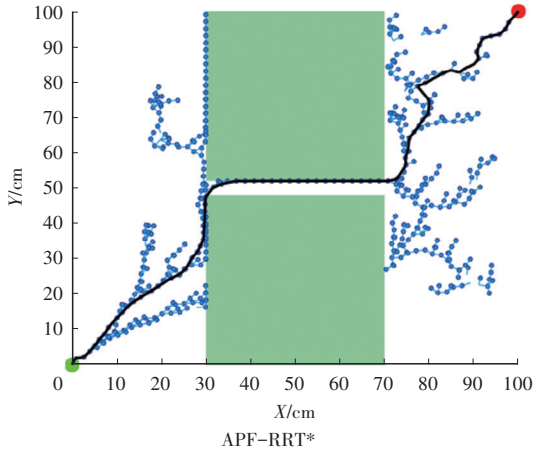
ABF-RRT*



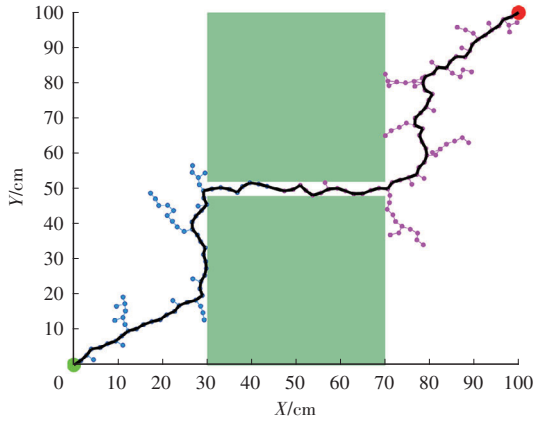
ABF-RRT*

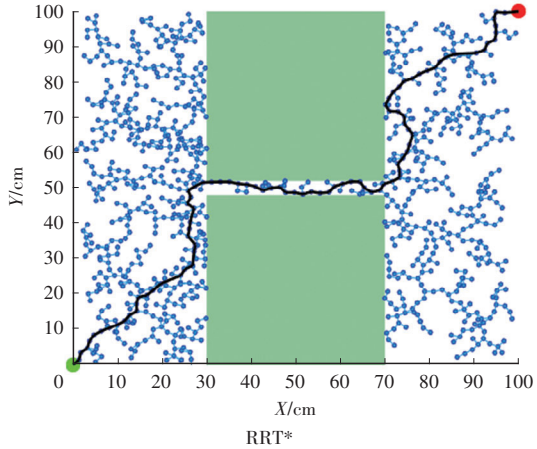


APF-RRT*



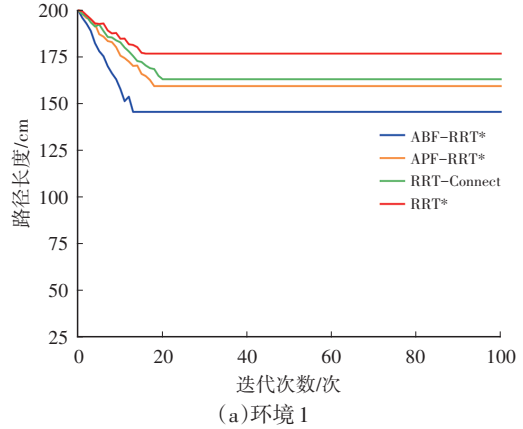
APF-RRT*



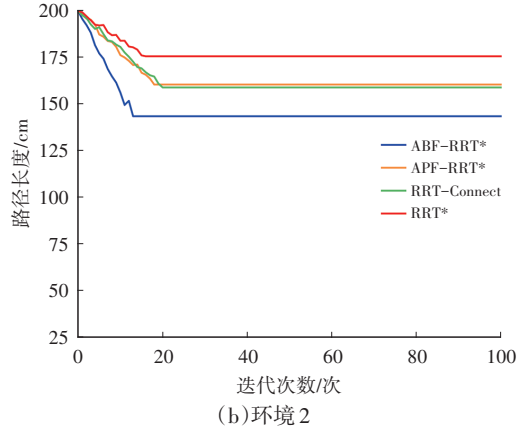


(d)环境4

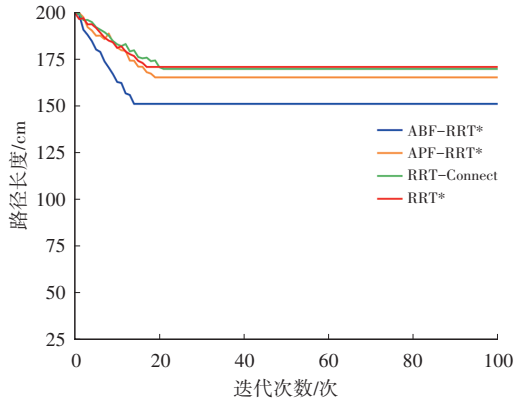
图12　各算法在不同环境的运行结果



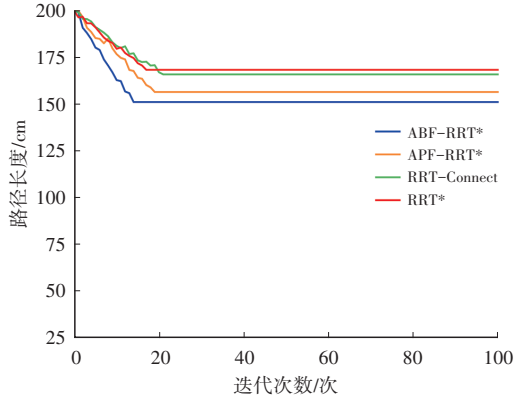
(a)环境1



(b)环境2



(c)环境3



(d)环境4

图13　本文方法在不同环境的收敛曲线

表1　各算法在不同环境下仿真数据

环境	算法	平均路径长度/cm	平均节点数量/个	平均耗时/s
1	ABF-RRT*	145.63	119	1.94
	APF-RRT*	159.39	1 224	36.24
	RRT-Connect	163.07	190	3.28
	RRT*	176.77	1 268	43.23
2	ABF-RRT*	143.66	126	1.58
	APF-RRT*	160.65	990	27.20
	RRT-Connect	159.08	140	2.31
	RRT*	175.78	1 083	45.36
3	ABF-RRT*	151.36	255	3.98
	APF-RRT*	165.55	901	40.43
	RRT-Connect	170.08	379	6.31
	RRT*	171.15	1 685	52.53
4	ABF-RRT*	151.26	139	2.98
	APF-RRT*	156.65	1 590	44.20
	RRT-Connect	166.08	219	4.31
	RRT*	168.53	1 833	50.60

由表1可知,在环境1中,相较于RRT*算法,ABF-RRT*算法的平均节点数减少90.6%,平均路径长度减少17.7%,平均耗费时长缩短95.5%。这是由于本文的自

适应步长双向扩展机制使2棵树能够相向生长，避免了传统RRT*单向随机扩展产生的大量无效节点。同时，改进人工势场的双重斥力使新节点在障碍物密集区能够主动偏离危险区域，减少了因碰撞导致的重复采样，从而显著降低了节点数量、缩短了搜索时间。

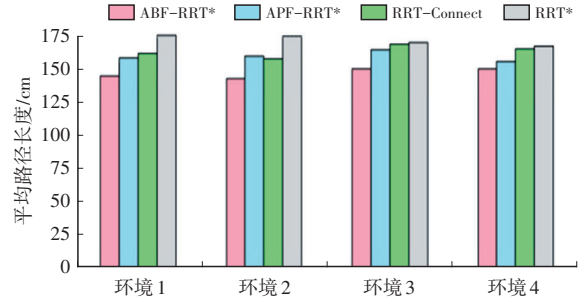
由于环境2的障碍物分布更分散，传统RRT*因缺乏目标导向性，容易在空旷区域产生大量无意义的探索节点。而ABF-RRT*通过自适应目标偏置策略，能够根据环境密度动态调整采样方向，使搜索树更早地指向目标区域，从而在保证路径质量的同时，大幅缩短规划时间。因此，相较于RRT*算法，优化后的算法的平均路径长度减少18.2%，平均节点数减少88.4%，平均耗费时长缩短96.5%。

在环境3中，传统RRT*和RRT-Connect受随机采样影响，扩展节点易在通道入口处反复碰撞障碍物，导致节点数量激增、收敛缓慢。ABF-RRT*依托改进人工势场的双重斥力与自适应步长机制，在狭窄通道中能够精确控制扩展方向与步长，避免无效碰撞，有效穿越通道。相较于RRT*算法，ABF-RRT*的平均路径长度减少11.6%，平均节点数减少84.9%，平均耗费时长缩短92.4%。

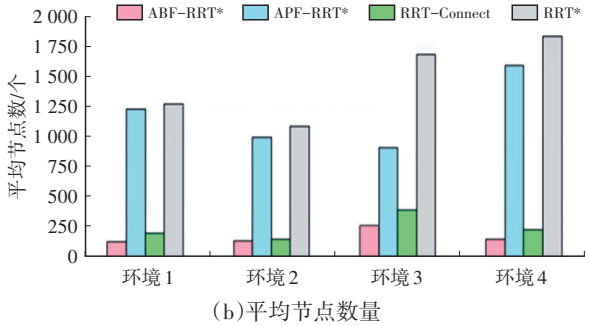
环境4为简单通道型环境，障碍物稀少、通行空间充裕，各算法均能较快规划出可行路径。传统RRT*因随机性过强，仍会产生大量偏离目标方向的冗余节点。而ABF-RRT*在空旷区域凭借自适应目标偏置策略使采样点以较高概率指向目标区域，同时自适应步长在低密度环境下保持较大值，进一步加速搜索树的生长。因此，相较于RRT*算法，ABF-RRT*的平均路径长度减少10.3%，平均节点数减少92.4%，平均耗费时长缩短94.1%，本文方法在简单环境下同样保持明显的效率优势。

本文在4种不同环境中重复进行100次试验，记录下平均路径长度、平均节点数量、平均运行时间，仿真数据对比统计结果如图14所示。在4种不同环境下，相较于传统RRT*算法，本文算法的路径长度依次减少17.7%、18.3%、11.6%和10.25%，说明自适应引力系数在复杂环境中更能发挥引导作用；平均节点数量依次减少90.6%、88.4%、84.9%、92.4%，表明双向扩展机制和贪婪剪枝策略有效避免了无效节点的累积；运行时间依次缩短95.5%、96.5%、92.4%、94.1%，主要得益于协同优化策略加速了2棵树的收敛过程。

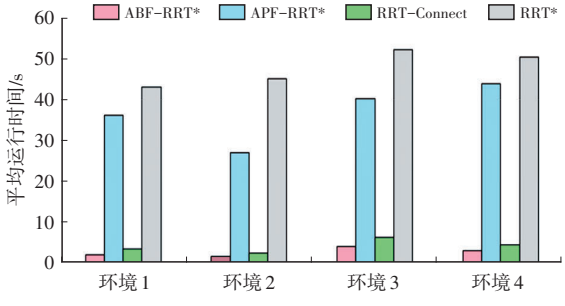
本文在Ubuntu 20.04操作系统环境中，借助Gazebo仿真平台，模拟环境3构建10 m×10 m虚拟地图，结果如图15所示。ABF-RRT*的路径长度为15.1 m，运行时间为6.7 s，转折2次；RRT*的路径长度为15.6 m，运行时间为10.2 s，转折3次。在引入贪婪算法后，ABF-RRT*与传统RRT*虽然均可规划出一条从起点到终点的无碰撞路径，但传统RRT*算法规划出的路径距离障碍物过近且转折点不够平滑，而ABF-RRT*算法规划的路径更加平滑、安全。



(a)平均路径长度

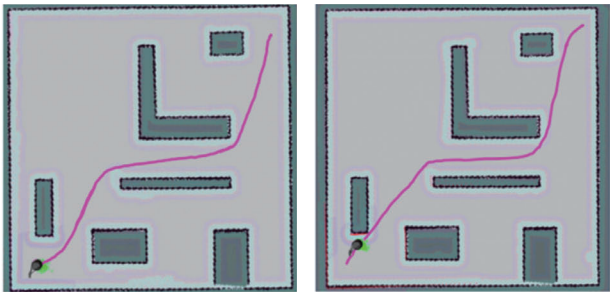


(b)平均节点数量



(c)平均运行时间

图14　仿真数据对比统计结果



(a)ABF-RRT*　　(b)RRT*

图15　Gazebo仿真结果

5　试验结果分析

为验证本文方法在实际环境中的有效性，按照环境3搭建6 m×6 m实际试验场地，如图16所示。使用四轮

差速移动机器人HW-Autocar进行试验，HW-Autocar的上位机运行Ubuntu 20.04.6 LTS focal操作系统和ROS noetic机器人操作系统，建图时启动基于SLAM算法的功能节点。

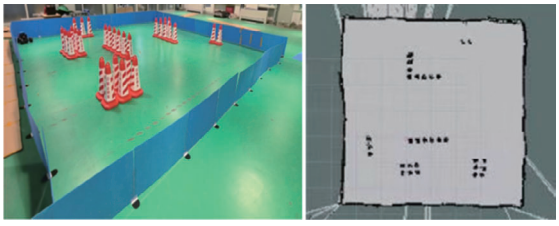
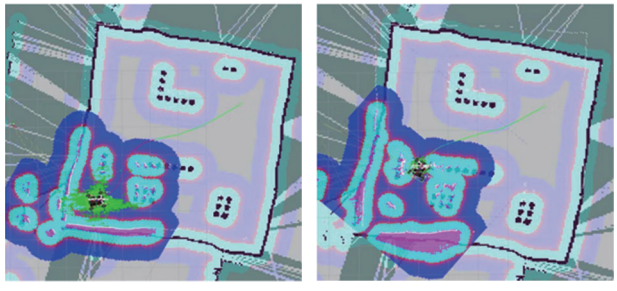


图16　实际试验环境

地图构建完成后，使用HW-Autocar进行导航试验。将构建的地图数据导入可视化插件Rviz，结果如图17所示。蓝色区域为已探索的可行区域，红色区域为障碍物及不可通行区域，绿色线条为规划路径，绿色箭头为AMCL定位信息。ABF-RRT*算法在实际环境中成功规划出一条从起点到目标点的连续无碰撞路径。绿色规划路径平滑连贯，与障碍物区域保持合理安全距离，未出现穿越红色障碍区域或紧贴障碍物边界的情况；AMCL定位产生的绿色箭头分布集中，表明定位准确，为路径跟踪提供了可靠位姿信息；路径转折较少，符合智能车运动学约束，验证了本文算法在实际场景中的有效性和可行性。



（a）起始阶段　　（b）运行途中

图17　HW-Autocar运动过程

在导航过程中，将本文算法与RRT-Connect、RRT*算法进行对比，结果如表2所示，相较于其他算法，本文方法的实际路径规划效果更优。

表2　实物试验数据

算法类型	路径长度/m	运行时间/s	转折次数/次
ABF-RRT*	8.87	7.1	1
RRT-Connect	9.09	9.8	1
RRT*	9.35	10.5	3

6　结束语

本文提出一种结合改进人工势场与自适应步长双向扩展的ABF-RRT*算法，通过引入自适应引力系数、双重斥力成分，解决了目标点附近障碍物过多导致斥力大于引力而无法到达目标点的局限性。同时，采用自适应步长、协同优化策略、动态调整目标偏置概率，显著提升了路径规划的效率、质量与目标导向性。ABF-RRT*在复杂和简单环境中均可快速找到更短、更优的路径，减少了节点数量、缩短了计算时间，在实际试验中进一步验证了其有效性和可靠性。未来工作将考虑在动态障碍物环境下，结合实时避障策略与速度规划，进一步提升算法在非结构化场景中的适应能力。

参　考　文　献

[1] MENG Q H, QIAN C J, SUN Z Y, et al. Autonomous Parking Method Based on Improved A* Algorithm and Model Predictive Control[J]. Nonlinear Dynamics, 2025, 113: 6839-6862.

[2] CAI Z Z, SELEZNEVA M S, YANG M. Dijkstra Algorithm Based Minimum Acceleration/Snap Quadrotor UAV Trajectory Planning[C]// The 14th Asia Conference on Mechanical and Aerospace. Hong Kong, China: IOP, 2023.

[3] 曹晓意, 罗煦琼, 李景, 等. 改进人工势场法下的多无人机编队路径规划方法[J]. 计算机应用, 2025, 45(增刊1): 183-187.
CAO X Y, LUO X Q, LI J, et al. Multi-UAV Formation Path Planning Method Based on Improved Artificial Potential Field[J]. Journal of Computer Applications, 2025, 45(S1): 183-187.

[4] 陈国军, 林羊龙, 陈巍, 等. 多策略融合的扑翼飞行器三维航路规划RRT算法[J]. 制造业自动化, 2024, 46(9): 104-113.
CHEN G L, LIN Y L, CHEN W, et al. RRT Algorithm for 3D Route Planning of Flapping-Wing Aircraft Based on Multi-Strategy Fusion[J]. Manufacturing Automation, 2024, 46(9): 104-113.

[5] JANJOS F, REICHART R, NIERMEYER P. Smooth Path-Generation around Obstacles Using Quartic Splines and RRTs[J]. IFAC-PapersOnLine, 2017, 50(1): 9108-9113.

[6] 周秦源, 张磊, 邓越平, 等. 复杂环境下改进RRT算法的机械臂路径规划[J]. 机械传动, 2024, 48(9): 18-26+34.
ZHOU Q Y, ZHANG L, DENG Y P, et al. Path Planning of Manipulator with Improved RRT Algorithm in Complex Environments[J]. Journal of Mechanical Transmission, 2024, 48(9): 18-26+34.

[7] 鲁宇明, 周羽迳, 郭鑫, 等. 改进Informed RRT*算法移动机器人路径规划[J]. 计算机工程与应用, 2025, 61(8): 283-293.
LU Y M, ZHOU Y K, GUO X, et al. Mobile Robot Path Planning with Improved Informed RRT* Algorithm[J]. Computer Engineering and Applications, 2025, 61(8): 283-

293.

[8] 王昊，卢玉斌. 基于深度强化学习的RRT*路径规划[J]. 自动化与仪器仪表，2025(3)：15-19.
WANG H，LU Y B. RRT* Path Planning Based on Deep Reinforcement Learning[J]. Automation & Instrumentation，2025(3)：15-19.

[9] 盛兆康，宋家乐，水翔，等. 概率采样和目标偏置的变步长RRT*机器人路径规划[J]. 计算机工程与应用，2025，61(20)：114-122.
SHENG Z K，SONG J L，SHUI X，et al. Variable-Step RRT* Robot Path Planning with Probability Sampling and Goal Bias[J]. Computer Engineering and Applications，2025，61(20)：114-122.

[10] 罗济雨，孙丙宇. 基于改进型RRT*算法的移动机器人路径规划[J]. 计算机工程与设计，2024，45(8)：2357-2363.
LUO J Y，SUN B Y. Mobile Robot Path Planning Based on Improved RRT* Algorithm[J]. Computer Engineering and Design，2024，45(8)：2357-2363.

[11] 左国玉，关海山，郑榜贵. 基于多次采样启发式策略的改进RRT路径规划算法[J]. 计算机测量与控制，2024，32(7)：280-287.
ZUO G Y，GUAN H S，ZHENG B G. Improved RRT Path Planning Algorithm Based on Multi-Sampling Heuristic Strategy[J]. Computer Measurement & Control，2024，32(7)：280-287.

[12] FAN J M，CHEN X，LIANG X. UAV Trajectory Planning Based on Bi-Directional APF-RRT* Algorithm with Goal-Biased[J]. Expert Systems with Applications，2023，213.

[13] GANESAN S，RAMALINGAM B，MOHAN E R. A Hybrid Sampling-Based RRT* Path Planning Algorithm for Autonomous Mobile Robot Navigation[J]. Expert Systems with Applications，2024，258.

[14] 朱敏，陈思源，陈杰. 基于改进APF引导的双向RRT机械臂路径规划算法研究[J]. 现代制造工程，2025(2)：1-9.
ZHU M，CHEN S Y，CHEN J. Research on Bidirectional RRT Manipulator Path Planning Algorithm Based on Improved APF Guidance[J]. Modern Manufacturing Engineering，2025(2)：1-9.

[15] 葛超，张鑫源，王红，等. 一种改进RRT-Connect算法的路径规划研究[J]. 电光与控制，2025，32(3)：21-26.
GE C，ZHANG X Y，WANG H，et al. Path Planning Research Based on Improved RRT-Connect Algorithm[J]. Electronics Optics & Control，2025，32(3)：21-26.

（责任编辑　瑞　秋）

修改稿收到日期为2026年4月25日。

《汽车工艺与材料》投稿须知

《汽车工艺与材料》于1986年创刊，是由中国第一汽车集团有限公司主办的国内外公开发行的汽车材料与制造技术类月刊，目前已入选《中文核心期刊要目总览》（第二版）、RCCSE中国准核心学术期刊（B+）、中国核心期刊（遴选）数据库、中国学术期刊综合评价数据库、欧洲学术出版中心数据库（EuroPub）、哥白尼精选期刊数据库（ICI Journals Master List）、EBSCO International数据库、J-Gate数据库。

《汽车工艺与材料》以“为中国报道汽车制造，为汽车引领工艺材料”为办刊宗旨，致力于报道以汽车轻量化技术和智能制造技术为核心的先进制造技术与材料应用技术，重点关注电动汽车蓄电池、电机、电控关键材料技术，燃料电池材料技术，高强度钢、铝镁合金、非金属材料及其成形技术，连接技术，智能装备与绿色制造等，以期通过高质量学术内容的出版和传播助推行业创新技术的交流与发展。

《汽车工艺与材料》杂志关注领先的整车及零部件企业和材料、装备等供应商，及时报道汽车行业最新的产品设计、制造、材料、加工技术、生产装备、检测技术等方面的成功案例。

主要栏目：

AT&M视界、生产现场、材料应用、生产装备、检测技术、数字化园地、行业动态等。

投稿要求：

（1）来稿须具有独创性并与实践相结合，文章字数最好控制在5 000～8 000字。

（2）来稿不能在国内、外公开杂志上发表过，请勿一稿多投。

（3）来稿的试验方法、试验数据、试验结论必须准确、可靠。

（4）来稿须包括以下项目：题名、作者姓名、作者单位、摘要（200字左右）、参考文献等。来稿采用word文档的格式。

（5）来稿文章格式应符合一般科技论文格式，或参考近期本刊所刊登文章格式。

（6）文章必须附有公开发表的、体现本领域最新研究成果的参考文献，且在文中应标注文献引用处。

（7）本刊使用网站投稿，投稿网址：http://qcgyycl.cbpt.cnki.net，咨询电话：0431-82026054。

竭诚欢迎汽车行业及相关各界的专家学者积极向本刊投稿。