

基于改进 RRT^{*} 的移动机器人路径规划算法

裴以建, 杨超杰, 杨亮亮

(云南大学 信息学院, 昆明 650500)

摘 要: RRT^{*} 算法在路径规划过程中确保了其概率完备性和渐进最优性, 但仍存在收敛速度慢且产生大而密集的采样空间等问题。为此, 在 RRT^{*} 的基础上提出一种新的离线路径规划算法。在开始采样前快速瞄准目标区域, 生成包含起始点与目标点的有界连通性区域。采用人工势场法优化采样过程, 减少计算量, 在势场合力的作用下有界区域内执行目标偏差采样, 调整采样区域, 在较短时间内生成较优路径。实验结果表明, 当生成的路径成本相同时, 该算法所需的迭代次数和执行时间远小于 RRT^{*} 算法, 算法效率更高。

关键词: 机器人; 路径规划; 人工势场; 有界区域; 偏差采样

中文引用格式: 裴以建, 杨超杰, 杨亮亮. 基于改进 RRT^{*} 的移动机器人路径规划算法[J]. 计算机工程, 2019, 45(5): 285-290, 297.

英文引用格式: PEI Yijian, YANG Chaojie, YANG Liangliang. Path planning algorithm for mobile robot based on improved RRT^{*} [J]. Computer Engineering, 2019, 45(5): 285-290, 297.

Path Planning Algorithm for Mobile Robot Based on Improved RRT^{*}

PEI Yijian, YANG Chaojie, YANG Liangliang

(School of Information Science and Engineering, Yunnan University, Kunming 650500, China)

[Abstract] The RRT^{*} algorithm ensures its probabilistic completeness and asymptotic optimality in the path planning process, but there are still some problems such as slow convergence and large and dense sampling space. To solve these problems, this paper proposes a new offline path planning algorithm based on RRT^{*}. The proposed algorithm quickly aims at the target area before starting sampling, and generates a bounded connected region that includes the starting point and the target point. The artificial potential field method is used to optimize the sampling process by reducing computational load. The target deviation sampling is executed in the bounded area under the action of the resultant force in the potential field, and generate a better path in a short period of time by flexible adjustment of the sampling area. Experimental results show that at the same cost of path planning, the number of required iterations and execution time of the proposed algorithm is much less than those of the RRT^{*} algorithm, which means the proposed algorithm is more efficient.

[Key words] robot; path planning; artificial potential field; bounded region; deviation sampling

DOI: 10.19678/j.issn.1000-3428.0049692

0 概述

路径规划是指生成一条从初始位置到指定目标位置的无碰撞路径, 其有着广泛的应用, 如无人车、太空任务的行星探测车、电脑游戏、计算机辅助手术等。完整的路径规划算法是在有限的时间内收敛到一个确定存在的路径解算法。然而在多数实际路径规划过程中, 这些算法计算效率被证明是非常低的^[1], 如: 人工势场法是一种成熟的机器人局部路径规划算法, 适用于实时控制, 但存在容易陷入局部最

优和死锁现象^[2]; 蚁群算法、粒子群算法、遗传算法、细胞分解法等对于较复杂的环境需要大量的迭代去寻找并优化路径, 导致求解困难、收敛时间长^[3]。

此外, 在安全生活应用中, 例如搜救任务的目标是在短时间内找到一条相对较优路径, 因此需要一个合适的算法去规划出路径^[4]。基于抽样的方法被广泛应用于解决上述问题, 其主要包括 PRM^[5] 和 RRT^[6]。这些方法具备概率完备性, 即如果在起始点和目标点间有路径解存在, 只要规划或搜索的时间足够长, 就一定能确保找到一条路径解。PRM 对

基金项目: 云南大学服务云南行动计划项目(KS161012)。

作者简介: 裴以建(1957—), 男, 教授, 主研方向为智能机器人; 杨超杰、杨亮亮, 硕士研究生。

收稿日期: 2017-12-13 **修回日期:** 2018-03-20 **E-mail:** pei3p@ynu.edu.cn

障碍物几何形状的依赖使其仅适用于静态环境,对于杂乱的环境,RRT 比 PRM 更为可行,并且对非完整约束也有固有的支持^[7]。但 RRT 也存在一些制约因素,比如在整个空间中进行探索和抽样时,增加了不必要的节点,提高了对存储器性能的要求,导致计算时间相应增长,收敛速度缓慢。

正是由于这些制约因素的存在,RRT 及其变体在最近几年得到了广泛的研究。文献[8]提出 RRT-connect 算法,通过 2 颗树不断朝对方交替扩展,采用贪心启发式连接 2 棵定向树,提高了算法的收敛速度。文献[9]提出的 RRT* 算法,允许对树进行重新布线,使得从根到叶的路径长度减小,改进了 RRT 算法产生非最优解的问题。文献[10]提出的 B-RRT* 算法,是 RRT* 算法的简单双向实现,通过用贪心 RRT-connect 启发式的轻微变化来连接 2 棵树。文献[11]提出 IB-RRT* 算法,通过引入智能样本插入启发式使算法快速收敛到最优路径解。文献[12]提出的 RRS 算法是改进的可变形活动轮廓模型 snack 和 RRT 算法的组合,在路径安全性方面给出了很好的解决方案。

本文提出新型离线路径规划算法 RP-RRT*。以 RRT* 算法为基础,该算法开始采样之前瞄准目标区域,在起始点和目标点之间构建一个可调边界的连通性区域。该连通区域可以根据不同的障碍物环境调整大小,具有足够的灵活性。同时,本文引入人工势场法理论,通过建立虚拟势场,优化 RRT* 算法的采样过程,减少采样过程的计算量。

1 背景知识

1.1 RRT* 算法

给定状态空间 Z ,障碍物区域和无障碍物区域分别表示为 Z_{obs} 和 $Z_{\text{free}} = Z \setminus Z_{\text{obs}}$, $A \in Z_{\text{free}}$ 是起始位置, $B \in Z_{\text{free}}$ 是目标位置。RRT* 算法构建的树为 $T = (V, E)$,其中, V 代表节点集合, E 代表连接节点的边沿结合。本文仅考虑欧几里得空间,定义树中任意 2 个节点 (z_1, z_2) 之间的路径成本值 $\text{Cost}(\Psi(z_1, z_2))$ 是它们之间的欧几里得距离, $\text{Cost}(z, T)$ 是起始点朝向目的节点 z 的欧几里得距离总和。

RRT* 的伪代码如算法 1、算法 2 所示。

算法 1 RRT* 伪代码

```

 $T \leftarrow \text{RRT}^*(A, B)$ 
1.  $V \leftarrow (A), E \leftarrow \emptyset; T \leftarrow (V, E)$ 
2. for  $i \leftarrow 0$  to  $N$  do
3.  $z_{\text{rand}} \leftarrow \text{Sample}(i)$ 
4.  $z_{\text{nearest}} \leftarrow \text{Nearest}(z_{\text{rand}}, V)$ 
5.  $z_{\text{new}} \leftarrow \text{Steer}(z_{\text{nearest}}, z_{\text{rand}})$ 
6. ChooseBestParent and Rewiring( $z_{\text{nearest}}, z_{\text{new}}, T$ )

```

```

7.  $\Psi_{A,B}(T) \leftarrow \text{FindBestPath}(A, B, T)$ 
8. Return  $\Psi_{A,B}(A, B, T)$ 

```

算法 2 选择最佳父节点和重新布线伪代码

```

ChooseBestParent and Rewiring( $z_3, z_4, T$ )
1. if ObstacleFree( $\Psi(z_3, z_4)$ ) then
2.  $z_{\text{min}} \leftarrow z_3$ 
3.  $Z_{\text{near}} \leftarrow \text{NearVertices}(z_4, T)$ 
4. for  $z_{\text{near}} \in Z_{\text{near}}$  do
5.  $\text{NewCost} \leftarrow \text{Cost}(z_{\text{near}}, T) + \text{Cost}(\Psi(z_{\text{near}}, z_4))$ 
6. if  $\text{NewCost} < \text{Cost}(z_4, T)$  then
7.  $z_{\text{min}} \leftarrow z_{\text{near}}$ 
8.  $V \leftarrow V \cup \{z_4\}, E \leftarrow E \cup \{\Psi(z_{\text{min}}, z_4)\}$ 
9. for all  $z_{\text{near}} \in Z_{\text{near}} \setminus \{z_{\text{min}}\}$  do
10. if ObstacleFree( $\Psi(z_4, z_{\text{near}})$ ) and  $\text{Cost}(z_{\text{near}}, T) > \text{Cost}(z_4, T) + \text{Cost}(\Psi(z_4, z_{\text{near}}))$  then
11.  $z_{\text{parent}} \leftarrow \text{Parent}(z_{\text{near}})$ 
12.  $E \leftarrow E \setminus \{\Psi(z_{\text{parent}}, z_{\text{near}})\}$ 
13.  $E \leftarrow E \cup \{\Psi(z_4, z_{\text{near}})\}$ 

```

在算法 1 中,初始化的 RRT* 算法通过从给定的无障碍区域 Z_{free} 中随机采样一个节点 z_{rand} 开始其迭代过程($\text{Sample}(i)$),由 Nearest 函数得出节点集合 V 中距离 z_{rand} 最近的节点 z_{nearest} ,再采用 Steer 函数得到沿 z_{rand} 方向上距离 z_{nearest} 一定步长的节点 z_{new} 。与原始 RRT 算法不同的是,选择最佳父节点和重新布线(步骤 6)。最后,随着迭代的进行,逐渐从无障碍物区域 Z_{free} 中优化出一条低成本较优路径 $\Psi_{A,B}(T)$ 。

选择最佳父节点和重新布线(ChooseBestParent and Rewiring)的伪代码如算法 2 所示。ObstacleFree($\Psi(z_3, z_4)$)过程为碰撞检测,即任意给定 2 个节点 z_3, z_4 ,如果连接这 2 个节点的路径 $\Psi(z_3, z_4) \in Z_{\text{free}}$,则函数 ObstacleFree($\Psi(z_3, z_4)$)返回 true;否则函数返回 false。 $\text{NearVertices}(z_4, T)$ 过程求出 z_4 的邻近节点集合 Z_{near} ,如图 1(a)所示。这些节点是靠近节点 z_4 的,它们位于以 z_4 为球心,半径 $r \in R, r > 0$ 的封闭球形区域内部,表达式如式(1)所示。

$$B_{r,z} = \{z_4 \in Z_{\text{free}}, d(z, z_4) \leq r\} \quad (1)$$

$$r = \min\{\xi (\lg(\text{Num}(V))/\text{Num}(V))^{1/n}\} \quad (2)$$

其中, ξ 为常数, $\text{Num}(V)$ 是集合 V 中所有节点的数目, n 是配置空间的维度。

如图 1(b)所示,以集合 Z_{near} 中的节点作为中间节点,通过比较到达节点 z_4 时的路径成本大小得出 z_4 的最佳父节点 z_{min} ,将节点 z_4 插入到树中,把边沿 $\Psi(z_{\text{min}}, z_4)$ 添加到集合中。如图 1(c)所示,采用回溯的方法,以 z_4 作为中间节点,比较到达节点 $z_{\text{near}} \in Z_{\text{near}}$ 时的路径成本大小对树进行重新布线。

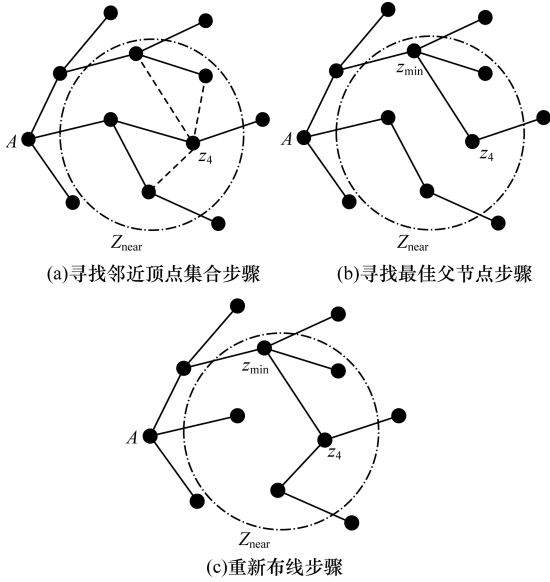


图1 算法2 执行过程

1.2 人工势场法

人工势场法是由 Khatib 首先提出的一种虚拟力法^[13],其基本思想是将机器人的运动环境虚拟成抽象的人造势场,在目标位置构造引力势场,在障碍物周围构造斥力势场,引力势场的引力吸引机器人朝向目标点移动,斥力势场的斥力阻止机器人朝向障碍物方向移动,最后在这2个势场力合力的作用下,机器人朝向目标点方向移动^[14]。

假设机器人位于 $\mathbf{X} = (x, y)^T$ 的平面区域中,目标点的坐标为 $\mathbf{X}_b = (x_b, y_b)^T$,定义引力势场函数如式(3)所示,斥力势场函数如式(4)所示。

$$U_{att} = \frac{1}{2} \lambda (\mathbf{X} - \mathbf{X}_b)^2 \quad (3)$$

$$U_{rep} = \begin{cases} \frac{1}{2} \eta \left(\frac{1}{\rho} - \frac{1}{\rho_0} \right), & |\rho| \leq |\rho_0| \\ 0, & |\rho| > |\rho_0| \end{cases} \quad (4)$$

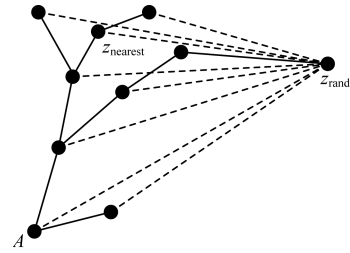
其中, λ 为引力势场常数, η 为斥力势场常数, $\mathbf{X}$ 为机器人当前坐标, $\mathbf{X}_b$ 为目标点坐标, ρ_0 为障碍物影响的最大距离, ρ 为障碍物和机器人的最短距离。这些常数的取值取决于配置空间的大小。由于引力函数为引力势场的负梯度^[15],因此定义引力函数如式(5)所示。同理斥力函数如式(6)所示,机器人受到的合力为: $\mathbf{F} = \mathbf{F}_{att} + \mathbf{F}_{rep}$ 。

$$\mathbf{F}_{att} = -grad(U_{att}) = \lambda (\mathbf{X}_b - \mathbf{X}) \quad (5)$$

$$\mathbf{F}_{rep} = -grad(U_{rep}) = \begin{cases} \eta \left(\frac{1}{\rho} - \frac{1}{\rho_0} \right) \frac{1}{\rho^2} \frac{\partial \rho}{\partial \mathbf{X}}, & |\rho| \leq |\rho_0| \\ 0, & |\rho| > |\rho_0| \end{cases} \quad (6)$$

2 RRT* 算法的改进

RRT* 算法是在 RRT 算法的基础上开发的,其中添加了一个找到最佳父节点的过程,并对树进行重新布线,在给定的环境中生成较优路径。然而, RRT* 算法在迭代过程需要对整个环境进行完全采样,产生了大量没有必要的节点,而且计算量随样本数量的增加而增加,最后优化整个采样区域中的路径比较困难。如图2所示,当生成样本点 $z_{nearest}$ 时, RRT* 会执行一个计算处理来找到距离 z_{rand} 最近的节点,这需要计算出与所有当前节点之间的距离并比较大小,且计算时间取决于树中的节点数量。随着迭代的进行,树中节点的数目逐渐增加,生成样本点 $z_{nearest}$ 的计算时间将增加。

图2 寻找 z_{rand} 的最近节点示意图

为了减少计算量和额外的无用节点,增强计算效率,本文算法将人工势场法引入 RRT* 中。如算法3所示,将给定的状态空间 Z 视为一个势场域,随机生成一个节点 z_{rand} ,新算法舍去原算法的 Nearest 和 Steer 过程,算法在目标点引力和障碍物斥力的共同作用下朝合力方向沿一定的步长 $|\tau|$ 生成新的节点 z_{new} 。为了使树生长得更快,判断节点 z_{new} 是否与障碍物发生碰撞,如果没有碰撞,则朝 z_{new} 的方向继续前进步长 $|\tau|$ 生成节点 z'_{new} ;如果发生碰撞,则朝新的合力方向继续前进步长 $|\tau|$ 生成节点 z'_{new} 。由于在地图中障碍物的分布比较复杂,因此计算合力方向取距离最近的障碍物和障碍物上距离节点 z_{rand} 或 z_{new} 最近的点。

算法3 RP-RRT* 伪代码

```

T ← RP-RRT* (A, B)
1. V ← {A}, E ← ∅, T ← (V, E)
2. C_region ← ConnectRegion(A, B, T)
3. for i ← 0 to N do
4. z_rand ← RegionSample(i)
5. z_new ← PotentialSteer(z_rand)
6. z'_new ← PotentialSteer(z_new)
7. if Obstaclefree(Ψ(z_new, z'_new)) then
8. V ← V ∪ {z'_new}
9. E ← E ∪ {Ψ(z_new, z'_new)}
10. ChooseBestParent and Rewiring(z_rand, z_new, T)
11. Ψ_{A,B}(T) ← FindBestPath(A, B, T)
12. if Ψ_{A,B}(A, B) then

```

13. Return $\Psi_{A,B}(A,B,T)$

14. else

15. $C_{region} = \text{ConnectRegion}(A,B,T)$

对节点 z_{rand} 和 z_{new} 进行扩展时,在配置空间中某些位置的节点受到的引力可能等于斥力,即受到的合力为 0,这时将无法扩展出新的节点,这些节点在人工势场法中称为局部极小值点^[16]。在靠近障碍物附近,节点 z_{rand} 和 z_{new} 受到的斥力远大于引力,扩展出的新节点将远离障碍物,无法保留障碍物附近有益的节点。

为了解决极小值点问题和保留障碍物附近有益的节点,当节点 z_{rand} 和 z_{new} 所受斥力等于引力或在靠近障碍物附近斥力远大于引力(设置斥力大于引力的阈值)时,将斥力一分为二,如式(7)所示。

$$F_{rep} = \begin{cases} F_1 + F_2, & |\rho| \leq |\rho_0| \\ 0, & |\rho| > |\rho_0| \end{cases} \quad (7)$$

其中一个指向节点 z_{rand} 或 z_{new} 为 F_1 ,如式(8)所示,另一个指向节点 z_{rand} 或 z_{new} 的反方向为 F_2 ,如式(9)所示。

$$F_1 = \frac{\rho}{\rho + \tau} \eta \left(\frac{1}{\rho} - \frac{1}{\rho_0} \right) \frac{1}{\rho^2} \cdot \frac{\partial \rho}{\partial X} \quad (8)$$

$$F_2 = -\frac{\tau}{\rho + \tau} \eta \left(\frac{1}{\rho} - \frac{1}{\rho_0} \right) \frac{1}{\rho^2} \cdot \frac{\partial \rho}{\partial X} \quad (9)$$

沿合力的方向扩展出新的节点 z_{new} 和 z'_{new} ,当距障碍物的最短距离 $|\rho|$ 大于步长 $|\tau|$ 时,斥力上的合力指向障碍物的反方向,节点扩展方向如图 3(a)所示,新的节点远离障碍物。当距障碍物的最短距离 $|\rho|$ 小于步长 $|\tau|$ 时,斥力上的合力指向障碍物,节点扩展方向如图 3(b)所示,新的节点靠近障碍物,这样既克服了极小值问题,又能保留障碍物附近的节点。

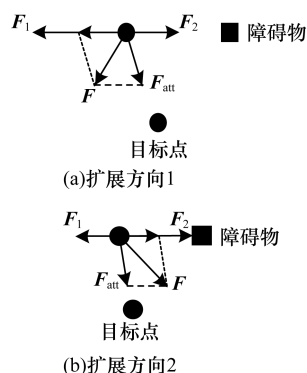
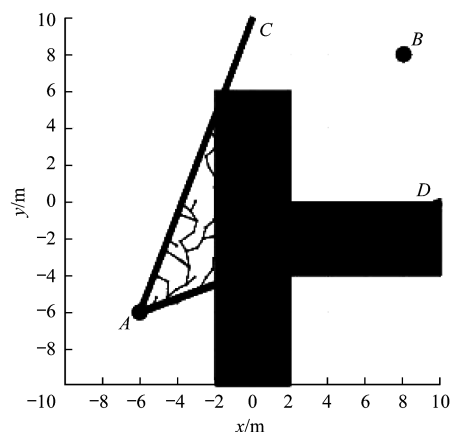


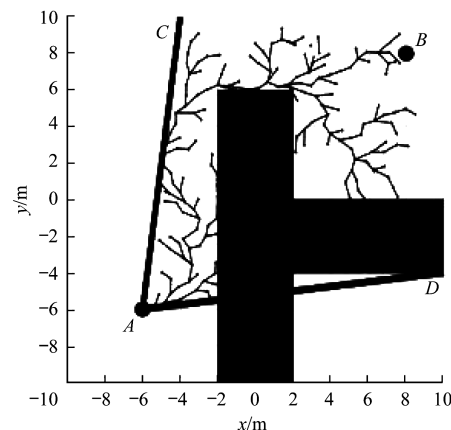
图 3 节点 z_{rand} 或 z_{new} 的扩展方向

如算法 3 的第 2 步所示,RP-RRT* 算法在开始采样前在离线的配置空间 Z 中确定一个包含起始点与目标点在内的连通区域 $C_{region} \in C_{region}$ 的定义比较简单,如图 4 所示,起始点与目标点之间的连线 AB

为采样区域中 $\angle CAD$ 的角平分线,当起始的采样区域不能构建全局路径时,连线 AC 与 AD 分别沿逆时针方向与顺时针方向旋转一定的角度 θ 来扩大采样区域,直到生成全局路径或者 2 条直线重合。



(a) 初始采样区域



(b) 扩展后的采样区域

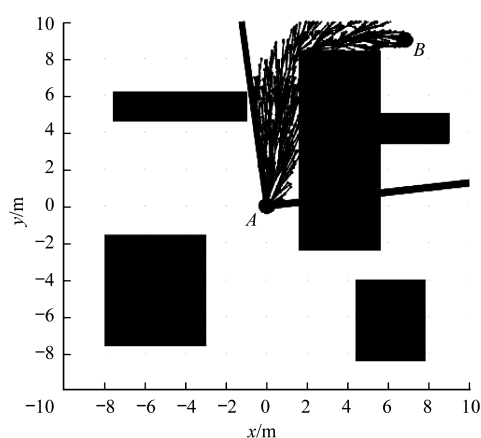
图 4 采样区域的扩展

3 实验与结果分析

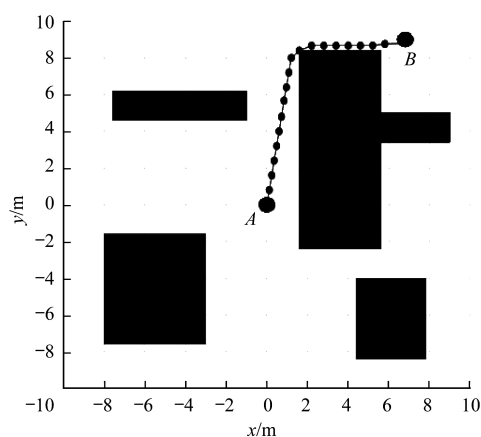
为了验证改进算法的有效性,本节在相同实验条件下对 RP-RRT* 算法和 RRT* 算法^[9]进行仿真,给出 RP-RRT* 算法最终生成的随机树和连接节点的全局路径,对 2 种算法的执行时间、迭代次数和路径成本进行对比。

在仿真环境中, x 轴坐标范围 $[-10, 10]$, y 轴坐标范围 $[-10, 10]$,黑色方块为障碍物, A 、 B 分别为起始位置和目标位置。设 RP-RRT* 的初始采样区域夹角为 90° ,每次扩展角度 $\theta = 25^\circ$,斥力大于引力的阈值为 5,即采样点所受到的斥力与引力的比值大于等于 5,障碍物影响的最大距离 $|\rho_0| = 4$ 。实验过程为对比文献^[9]中 RRT* 算法生成的路径成本,由 RP-RRT* 算法在相同的实验环境中灵活调整采样区域大小规划出一条相同成本的全局路径。

如图5所示的仿真环境,RRT*算法需要对整个环境进行完全采样,经过20 000次采样迭代后生成了一条连接位置 $A(0,0)$ 和位置 $B(6.5,9)$ 之间的较优路径,执行时间125.32 s,路径成本14.51^[9]。对于RP-RRT*算法,图5中的采样区域仅为初始区域,表1给出了RP-RRT*算法在初始采样区域内执行50次测试的平均值。从表1可以看出,RP-RRT*算法的平均迭代次数是852,执行时间为10.62 s。当生成的路径成本相同时,RP-RRT*算法所需的迭代次数和执行时间要远小于RRT*算法,算法效率更高。同时,从图5(a)可以看出,改进算法较好地保留了连通性有界采样区域中障碍物附近有益的节点,且在势场力的作用下采样快速朝目标方向处偏移,使得采样在给定的状态空间中分散较小。



(a)RP-RRT*采样



(b)RP-RRT*路径

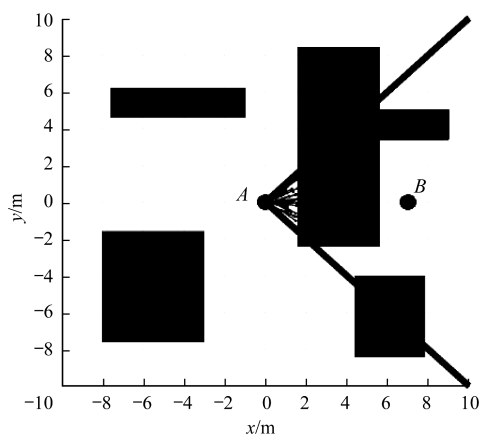
图5 在初始区域下RP-RRT*路径规划

表1 RP-RRT*算法在初始采样区域下的实验数据

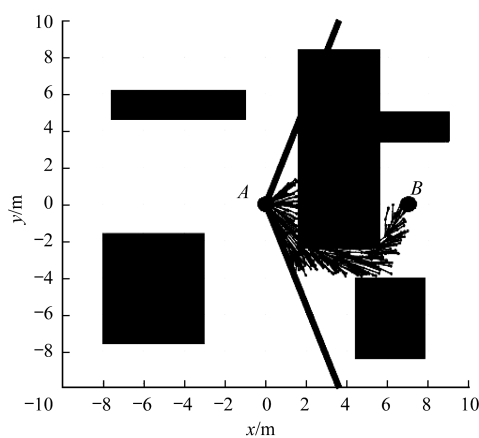
迭代次数	路径成本	执行时间/s
852	14.51	10.62

图6是RP-RRT*算法在初始采样区域中得不到全局路径解的情况。如图6(a)所示,改变目标位置后RP-RRT*算法在新的有界区域中不能生成全局路径,为了解决这个问题,在图6(b)中,通过一次

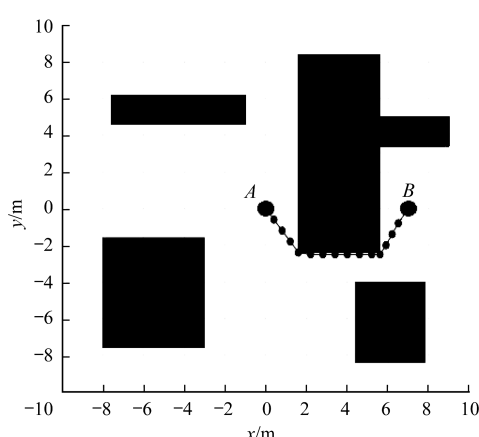
采样区域的扩展使得随机树继续朝目标位置生长,最终获得全局路径。



(a)采样区域夹角为90°



(b)采样区域夹角为140°



(c)最终路径

图6 在动态采样区域下RP-RRT*路径规划

表2给出了RP-RRT*算法在动态采样区域下的50次实验数据的平均值。从表2可以看出,RP-RRT*算法在1.72 s时刻扩展了采样区域,迭代结束时,平均迭代次数是435,执行时间为5.48 s。在同一环境下,当路径成本相同时,RRT*算法仍然对整个环境进行完全采样,迭代次数为20 000,执行时间为112.37 s^[9]。对比图6和表2可以看出,采样区域

扩展后,RP-RRT* 算法仍能够在较短的时间内以较少的迭代次数找到相对最优的可行解。

表 2 RP-RRT* 算法在动态采样区域下的实验数据

迭代次数	路径成本	扩展时刻/s	执行时间/s
435	8.07	1.72	5.48

为进一步验证人工势场法的作用,本文给出 RP-RRT* 算法对采样区域扩展 3 次的规划结果,如图 7 所示。表 3 列出了 RP-RRT* 算法对图 5~图 7 进行路径规划时的迭代次数和连通性区域与总区域面积的比值。

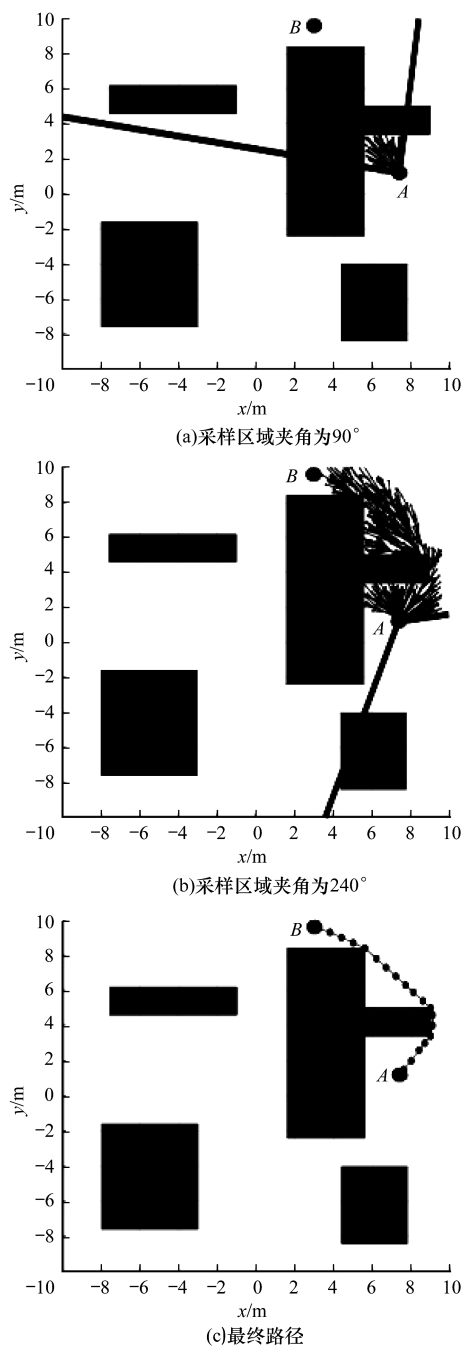


图 7 采样区域扩展 3 次的 RP-RRT* 路径规划

表 3 RP-RRT* 算法路径规划结果

仿真图形	迭代次数	连通性区域面积/总区域面积
图 5	852	0.25
图 6	435	0.40
图 7	771	0.80

由表 3 可以看出,当采样区域增加时,迭代次数不一定增加,RT-RRT* 算法的采样过程仍是使用较少的节点在势场合力的作用下快速收敛到目标位置。同时,将图 5(a)、图 6(a)和图 7(a)中的初始采样区域与文献[9]中 RRT* 算法进行大量迭代后的完全采样相比,RP-RRT* 算法的初始采样区域总能包含全局路径所在的重要区间,并且在势场力的作用下,当有界采样区域调整时,采样也是集中在有效区域中。

4 结束语

本文提出 RP-RRT* 离线路径规划算法。通过引入连通性有界区域和人工势场模型,在配置空间中进行目标偏差有界采样,从而优化了采样过程,减少了迭代次数。实验结果表明,本文算法比 RRT* 算法有更快的收敛速率。在实验过程中,本文仅随机设置了初始采样区域夹角和每次采样区域扩展的角度,但在不同环境下,不同的初始夹角和扩展角度会影响采样的迭代次数和收敛速度,下一步将对上述问题进行研究。

参考文献

[1] VARRICCHIO V, CHAUDHARI P, FRAZZOLI E. Sampling-based algorithms for optimal motion planning using process algebra specifications[C]//Proceedings of IEEE International Conference on Robotics and Automation. Washington D. C. , USA ;IEEE Press,2014 : 5326-5332.

[2] LIU Jianhua, YANG Jianguo, LIU Huaping, et al. An improved ant colony algorithm for robot path planning[J]. Soft Computing,2016,1(11):1-11.

[3] 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]//Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems. Washington D. C. , USA ;IEEE Press,2014:2997-3004.

[4] NOREEN I, KHAN A, RYU H, et al. Optimal path planning in cluttered environment using RRT* -AB[J]. Intelligent Service Robotics,2017(8):1-12.

[5] KAVRAKI L,LATOMBE J C. Randomized preprocessing of configuration for fast path planning[C]//Proceedings of IEEE International Conference on Robotics and Automation. Washington D. C. , USA ;IEEE Press,1994:2138-2145.

[6] LAVALLE S M, KUFFNER J J. Rapidly-exploring random trees: progress and prospects [M]//DONALD B. Algorithmic and computational robotics new directions. Cleveland,USA ;CRC Press,2000:293-308.

(下转第 297 页)

- [4] VLAHOIANNI E I, KARLAFTIS M G, GOLIAS J C. Optimized and meta-optimized neural networks for short-term traffic flow prediction: a genetic approach [J]. *Transportation Research Part C: Emerging Technologies*, 2005, 13(3): 211-234.
- [5] SYBERFELDT A, PERSSON L. Using heuristic search for initiating the genetic population in simulation-based optimization of vehicle routing problems [C]// *Proceedings of IEEE ISC'09*. Washington D. C., USA: IEEE Press, 2009: 524-536.
- [6] 巩现勇, 李靖涵. 一种道路路口和拐点的识别与重要性度量方法[J]. *测绘与空间地理信息*, 2018, 41(2): 5-9.
- [7] 王美玲, 潘允辉. 基于GIS与约束条件下的最优路径规划研究[J]. *北京理工大学学报*, 2016, 36(8): 851-856.
- [8] ZOURLIDOU S, SESTER M. Intersection detection based on qualitative spatial reasoning on stopping point clusters [C]// *Proceedings of International Conference on Remote Sensing and Spatial Information Sciences*. Washington D. C., USA: IEEE Press, 2016: 12-19.
- [9] 谭祥爽, 王静. 基于浮动车数据的路口探测方法[J]. *地理与地理信息科学*, 2015, 31(5): 34-38.
- [10] WANG Jing, WANG Chaoliang. Automatic intersection and traffic rule detection by mining motor-vehicle GPS trajectories [C]// *Proceedings of International Conference on Computers, Environment and Urban Systems*. Washington D. C., USA: IEEE Press, 2017: 19-29.
- [11] WANG Jing, RUI Xiaoping, SONG Xiaofeng. A novel approach for generating routable road maps from vehicle GPS traces [J]. *International Journal of Geographical Information Science*, 2015, 29(1): 69-91.
- [12] XIE Xingzhe, PHILIPS W, VEELAERT P. Road network inference from GPS traces using DTW algorithm [C]// *Proceedings of the 17th IEEE International Conference on Intelligent Transportation Systems*. Washington D. C., USA: IEEE Press, 2014: 906-911.
- [13] FATHI A, KRUMM J. Detecting road intersections from GPS traces [C]// *Proceedings of International Conference on Geographic Information Science*. Berlin, Germany: Springer, 2010: 56-69.
- [14] XIE Xingzhi, LIAO Wenzhi, AGHAJAN H, et al. A novel approach for detecting intersections from GPS traces [C]// *Proceedings of IEEE International Geoscience and Remote Sensing Symposium*. Washington D. C., USA: IEEE Press, 2016: 1816-1819.
- [15] KUMAR M, RAMA A, REDDY M. A fast DBSCAN clustering algorithm by accelerating neighbor searching using groups method [J]. *Pattern Recognition*, 2016, 58: 39-48.

编辑 索书志

(上接第290页)

- [7] KARAMAN S, FRAZZOLI E. Sampling-based optimal motion planning for non-holonomic dynamical systems [C]// *Proceedings of IEEE International Conference on Robotics and Automation*. Washington D. C., USA: IEEE Press, 2013: 5041-5047.
- [8] KUFFNER J J, LAVALLE S M. RRT-connect: an efficient approach to single-query path planning [C]// *Proceedings of IEEE International Conference on Robotics and Automation*. Washington D. C., USA: IEEE Press, 2000: 995-1001.
- [9] KARAMAN S, FRAZZOLI E. Incremental sampling-based algorithms for optimal motion planning [J]. *International Journal of Robotics Research*, 2011, 30(7): 5326-5332.
- [10] JORDAN M, PEREZ A. Optimal bidirectional rapidly-exploring random trees; MIT-CSAIL-TR-2013-021 [R]. Cambridge, UK: Computer Science and Artificial Intelligence Laboratory, 2013.
- [11] QURESHI A H, AYAZ Y. Intelligent bidirectional rapidly-exploring random trees for optimal-motion planning in complex cluttered environments [J]. *Robotics and Autonomous Systems*, 2015, 68: 1-11.
- [12] BAIZID K, CHELLALI R, LUZA R, et al. RRS: rapidly-exploring random snakes a new method for mobile robot path planning [M]// MENEGATTI E, MICHAEL N, BERNIS K, et al. *Intelligent autonomous systems 13*. Berlin, Germany: Springer, 2016: 291-305.
- [13] KHATIB O. Real-time obstacle avoidance for manipulators and mobile robots [J]. *International Journal of Robotics Research*, 1986, 5(5): 500-505.
- [14] 邓学强. 基于改进人工势场法的移动机器人路径规划[J]. *山东理工大学学报(自然科学版)*, 2014(1): 38-41.
- [15] 宋建辉, 代涛, 刘砚菊. 基于改进人工势场法的移动机器人路径规划[J]. *计算机工程与科学*, 2017, 39(7): 1328-1332.
- [16] 于振中, 闫继宏, 赵杰, 等. 改进人工势场法的移动机器人路径规划[J]. *哈尔滨工业大学学报*, 2011, 43(1): 50-55.

编辑 刘盛龄