

基于BI-APF-RRT算法的无人机路径规划

许锦杰, 杨晶晶, 魏童俊, 李广龙

江西理工大学电气工程与自动化学院, 江西 赣州

收稿日期: 2025年11月3日; 录用日期: 2025年12月23日; 发布日期: 2025年12月29日

摘要

针对双向RRT算法中存在较大搜索范围随机性问题, 论文提出一种基于目标偏置的BI-APF-RRT (Bidirectional Artificial Potential Field Method RRT)改进算法的无人机飞行路径规划策略。首先, 将目标偏置策略用于随机采样点的产生和双树扩张方向的指导, 并通过双向RRT算法建立两棵相互交替的随机搜索树完成搜索, 实现了算法收敛速率的提升。其次, 将改进的人工势场融入双向生长树, 进一步较大程度减少了搜索迭代次数。同时, 在平滑路径的过程中, 通过采用3次B样条插值算法实现了轨迹路径的优化。最后, 通过仿真实验表明: 与一些已有算法相比, 基于目标偏置的策略改进BI-APF-RRT算法能够有效减少迭代次数, 提升收敛速度, 同时改善了新节点生成的方向, 有效降低了路径成本, 较好地解决了双向RRT算法中存在较大搜索范围随机性问题。

关键词

双向RRT, 路径规划, 人工势场法, 目标偏置策略

UAV Path Planning Based on BI-APF-RRT Algorithm

Jinjie Xu, Jingjing Yang, Tongjun Wei, Guanglong Li

School of Electrical Engineering and Automation, Jiangxi University of Science and Technology, Ganzhou Jiangxi

Received: November 3, 2025; accepted: December 23, 2025; published: December 29, 2025

Abstract

To address the issue of significant search space randomness inherent in the Bidirectional Rapidly-exploring Random Tree (RRT) algorithm, this paper proposes an improved Unmanned Aerial Vehicle (UAV) flight path planning strategy based on a goal-biased Bidirectional Artificial Potential Field method RRT (BI-APF-RRT) algorithm. Firstly, a goal-biasing strategy is employed for the generation of random samples and to guide the expansion direction of the bidirectional trees. The search is completed

文章引用: 许锦杰, 杨晶晶, 魏童俊, 李广龙. 基于 BI-APF-RRT 算法的无人机路径规划[J]. 传感器技术与应用, 2026, 14(1): 26-38. DOI: 10.12677/jsta.2026.141004

by establishing two alternately growing random search trees via the bidirectional RRT mechanism, thereby enhancing the algorithm's convergence rate. Secondly, an improved Artificial Potential Field (APF) method is integrated into the bidirectional tree growth process, which further significantly reduces the number of search iterations. Concurrently, during the path smoothing phase, trajectory optimization is achieved through the application of a cubic B-spline interpolation algorithm. Finally, simulation experiments demonstrate that, compared to several existing algorithms, the proposed goal-biased BI-APF-RRT algorithm effectively reduces the number of iterations, enhances convergence speed, improves the directionality of new node generation, and effectively lowers the path cost. Consequently, it successfully mitigates the significant search space randomness problem associated with the conventional bidirectional RRT algorithm.

Keywords

Bidirectional RRT, Path Planning, Artificial Potential Field Method, Goal-Biased Strategy

Copyright © 2026 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

近些年来,随着无人驾驶技术的发展,无人机在农业[1]、军事[2]、物流运输[3]和工业[4]等领域得到广泛的应用。同时,凭借着高机动性、易于控制和价格便宜等优势,无人机深受国内外学者喜爱。随着无人机飞行任务越来越复杂、飞行环境越来越多变,路径规划在无人机控制系统中起着越来越重要的作用。目前,路径规划已成为无人机领域的重要研究热点之一。常见的路径规划算法有粒子群算法[5]、Dijkstra [6]、蚁群算法[7]等传统算法。基于网格的搜索算法典型代表是 A*算法,应用图论来解决离散状态空间中的运动规划问题[8]。

值得指出的是,基于样本的路径规划算法是目前较受欢迎的路径规划算法之一。由于该算法不受空间维度的限制,因此它可以在更高维度空间中找到更好的路径,并且具有概率完整性。最典型的算法之一是快速探索的随机树算法(RRT) [9]。与其他路径规划算法相比,该算法被广泛使用,因为它简单的结构和强大的搜索能力可以解决复杂的约束,但是该算法无法找到最佳路径,而全局采样会导致存在大量冗余点。

为了应对 RRT 的缺点,研究学者加入人工势场法,提出了 APF-RRT 算法,该算法解决了 RRT 的随机性问题[10]。RRT*算法在 2011 年通过引入渐进最优化机制,在保持概率完整性的同时,显著提升了路径质量[11]。该算法通过重选父节点和重新布线等方式减少搜索时间,但随着采样点的增加,会大大增加算法的计算量。尽管 RRT*可以找到更好的路径,但收敛速率仍然无法满足实时要求。有人对 RRT*算法进行研究,提出 RRT*-SMART-A*算法加速了收敛速度,并以两种智能采样的方式降低了 RRT*算法的路径成本[12]。为了 RRT*生成更好的采样空间,有人提出 Informed-RRT*算法[13]。

为了加快搜索速度,有人提出了基于双采样点的 BI-RRT 算法[14],该算法在起点和终点同时扩展两棵搜索树,减少了路径规划的时间,但规划的路径不是最优。通过改进双向树的连接方式,提出的 RRT-Connect 算法,能有效减少搜索时间[15]。

还有人提出改进的 RRT-Connect 算法[16],在采样过程中引入转角约束,以减少搜索节点,但上述算法随机树的生长是随机的,在不必要的区域进行采样会耗费大量的时间,规划的路径也不是最优的。

在双向树的基础上, 提出双向版的 RRT*算法, 称为 BI-RRT*算法, 搜索速度加快, 能有效优化路径, 但在复杂环境下计算节点数量较大[17]。在基于势场的双向规划算法中, BI-APF-RRT*代表了当前最先进的技术水平, 它通过引入人工势场(APF)来引导双向 RRT 树的扩展, 有效地提升了在复杂狭窄环境中的规划效率和解路径质量[18]。然而, 我们发现, BI-APF-RRT*的算法框架在追求渐进最优性的同时, 也引入了显著的在线计算负担。其核心在于, 它在每次迭代中都需要执行复杂的父节点重连和树形优化操作, 这在许多对实时性要求极高的应用场景(如机器人动态避障、自动驾驶紧急决策)中可能成为性能瓶颈。因此, 本文旨在探索一种在规划效率与路径质量之间取得更佳平衡的替代方案。为了适应更复杂的环境, 提出 DBVSB-P-RRT*算法, 设计自适应步长大小以增强环境适应性[19]。

本文主要提出了 BI-APF-RRT 算法的路径规划。它摒弃了 BI-APF-RRT*中计算密集的渐进最优性保证机制, 转而采用一种轻量级但高效的启发式优化策略。我们的核心思想是: 通过精心设计的 APF 引导和双向收敛机制, 在保持快速收敛到满意解的同时, 极大地提升算法的收敛速度。首先, 双向 RRT 算法同时创建两个搜索树, 并根据目标偏置策略交替生长, 可以获得更高质量的采样并减少收敛时间。接下来, 通过引入改进的人工势场方法, 能够有效减少迭代次数。然后, 应用 3 次 B 样条插值优化方法以获得更光滑的路径, 以便解决飞行中无人机频繁转弯的问题。最后, 与 RRT、APF-RRT、RRT*、APF-RRT* 和 BI-RRT 算法相比, 该算法生成的路径具有更高的性能。

2. 问题定义

数学符号用于描述无人机路径规划问题。路径规划算法的目的是规划无人机从起点到目标点的无障碍路径。本文研究了物流运输中无人机的路径规划算法。在本文中, F 被定义为整个环境, 即整个运输空间。 F_{ob} 是运输途中的障碍区域。 F_s 是环境中没有障碍的自由搜索区域。 (F, F_{start}, F_{goal}) 定义了一个路径规划问题, 其中 $F_{start} \in F_s$ 是随机生长树的根节点, 并且 F_{goal} 是生长树的目标节点。

定义 1: 存在连续的函数 $\sigma: [0, 1]$ 如果 $v \in [0, 1]$, $\sigma(v) \in F_s$, 则该路径是从根节点到目标节点的无碰撞路径。对于路径规划问题 (F, F_{start}, F_{goal}) , $\sigma(0) = F_{start}$, $\sigma(1) = F_{goal}$, 如果找不到解决方案, 则报告失败。

定义 2 (最佳轨迹): 给定轨迹规划问题 (F, F_{start}, F_{goal}) , 在发现了可行的路径 $\sigma(v)$ 之后, 经过优化后, $\sigma(v)$ 为最佳轨迹, 其中 $c(\sigma)$ 是路径的全长。如果找不到解决方案, 则报告失败。

定义 3 (最短时间路径): 在最短的时间内, 找到了最低成本的轨迹, 并获得了最佳解决方案。

3. BI-RRT 算法

双向扩展随机生长树 BI-RRT 算法从目标点和起始点同时生长两棵树, 直到两棵树相交, 最后回溯路径[20]。伪代码显示在算法 1 中。

算法 1. BI-RRT 算法

1. $T_{tree1} \leftarrow \{q_{start}\}, T_{tree2} \leftarrow \{q_{goal}\}$
2. for $i = 1 : N_{max}$ do
3. $q_{rand} \leftarrow \text{sample}(i)$;
4. $q_{near} \leftarrow \text{Near}(q_{rand}, T_{tree1})$;
5. $q_{new} \leftarrow \text{Steer}(q_{rand}, q_{near}, T_{tree1}, s_1)$;
6. if $\text{CollisionFree}(q_{near}, q_{rand}, F_{obs})$ then
7. $T_{tree1} \leftarrow T_{tree1} \cup \{q_{new}\}$;
8. else;
9. Continue;

```

10. End if;
11. if Connect( $T_{tree1}, T_{tree2}$ ) then
12. return  $T_{tree1}, T_{tree2}$ ;
13. break;
14. else;
15. Swap( $T_{tree1}, T_{tree2}$ );
16. Continue;
17. End if;
18. End for;
19.  $T_{tree} \leftarrow \text{Combine}(T_{tree1}, T_{tree2})$ ;

```

在伪代码中，第一棵树 T_1 开始生长，并且在每次迭代结束时确定与另一棵树之间的最小距离，如果距离小于固定值，则 Connect 函数将两棵树连接，否则，用 Swap 函数切换到另一棵树 T_2 再生长一次，重复以上过程。

4. 改进的目标偏置 BI-APF-RRT 算法

4.1. 高质量抽样策略

随机采样点产生得到了改善。目标策略根据以下等式(1)选择随机采样点。

$$q_{rand} = \begin{cases} random, & \text{if } rand() > m \\ q_{goal}, & \text{else} \end{cases} \quad (1)$$

其中， $rand()$ 是 0 到 1 之间的随机数， m 是一个目标偏置阈值，随机树中新节点的的生长方向由 m 确定。当 $rand() > m$ 时，生长树在采样空间中随机生长，反之，生长树朝目标点位置生长。在每次迭代中同时扩展了两棵随机树，如图 1 所示。

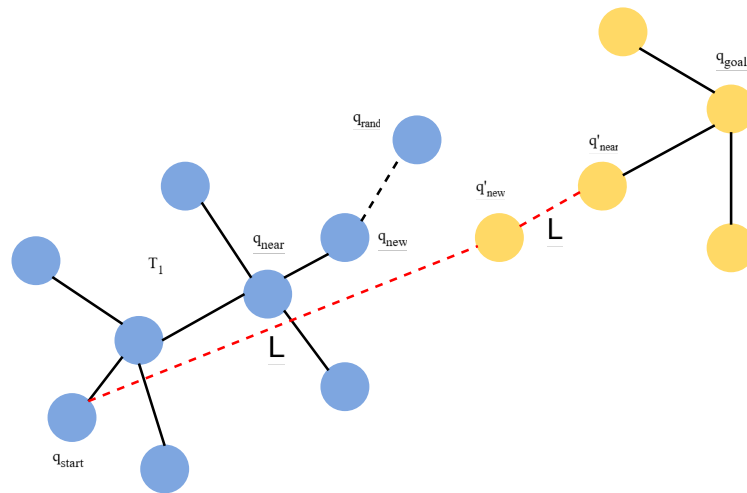


Figure 1. Bidirectional RRT node expansion process
图 1. 双向 RRT 节点扩展过程

4.2. 有效的搜索策略

人工势场法指导新的随机点的产生，通过目标的吸引力和障碍物的排斥力，使新的随机点往目标点

方向产生，从而衍生出新的树枝。

人工势场方法包括引力场 $U_{att}(p)$ ，如等式(2)所示。

$$U_{att}(p) = \frac{1}{2} k_a \rho_g^2(p) \quad (2)$$

式(2)中， P 是无人机的随机点， k_a 是重力场增益常数。排斥场 $U_{rep}(p)$ ，如等式(3)所示。

$$U_{rep}(p) = \begin{cases} \frac{1}{2} k_r \left(\frac{1}{\rho(p)} - \frac{1}{\rho_0} \right) & \rho(p) \leq \rho_0 \\ 0 & \rho(p) > \rho_0 \end{cases} \quad (3)$$

式(3)中， k_r 是排斥场增益常数。 ρ_0 代表每个障碍物的影响半径。总场就是斥力场和引力场的叠加，如等式(4)所示。

$$U_{total} = \sum U_{rep} + \sum U_{att} \quad (4)$$

引力就是引力场对距离的导数，斥力就是斥力场对距离的导数。因此，引力函数定义如等式(5)所示。

$$F_{att}(p) = k_a \rho_g(p) \quad (5)$$

式(5)中： $\rho_g(p)$ 表示该随机点与目标点的欧几里得距离。同样，斥力函数在等式(6)显示。

$$F_{rep} = \begin{cases} k_r \left(\frac{1}{\rho(p)} - \frac{1}{\rho_0} \right) & \rho(p) \leq \rho_0 \\ 0 & \rho(p) > \rho_0 \end{cases} \quad (6)$$

式(6)中： $\rho(p)$ 代表随机点和障碍物之间的距离。总的力就是引力和斥力的叠加，如等式(7)所示。

$$F_{total} = \sum F_{rep} + \sum F_{att} \quad (7)$$

双向 RRT 算法结合了改进的人工势场方法，其中以 T1 为例， F_{att} 是 q_{goal} 上的引力， F_{rep} 是 F_{obs} 的排斥力。根据平行四边形规则，发现所得的力方向是 q'_{rand} 的扩展方向。然后 q_{near} 朝着 q'_{rand} 的方向扩展一定步长得到 q_{new} 。双树 RRT 扩展方向如图 2 所示。

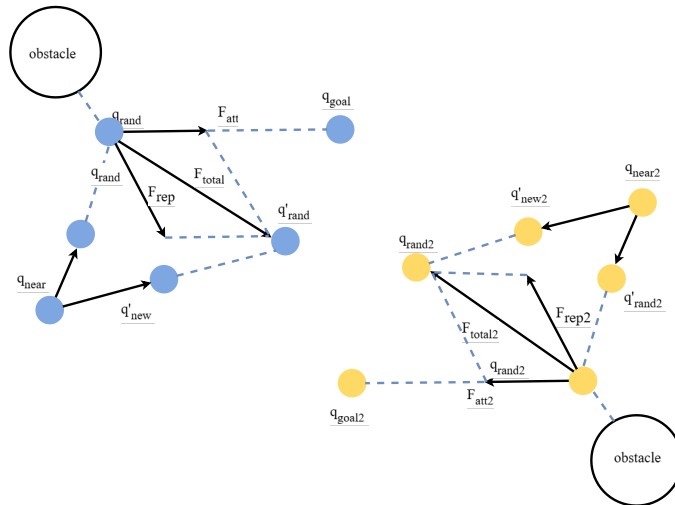


Figure 2. Force analysis diagram of the closest point of BI-RTT
图 2. BI-RTT 最近点的力分析图

但是,当无人机即将接近目标时,目标点附近存在障碍,而无人机的排斥力大于吸引力,因此无人机很难飞向目标,并且出现了局部最小值的问题。因此,本文提出了改进的人工势场函数(8),如下所示:

$$F_{total} = \begin{cases} \sum F_{rep} + \sum F_{att} & \varepsilon > \rho(q_{rand}, q_{obs}) \\ \sum F_{rep} + \sum F_{att} + F_{random} & \varepsilon < \rho(q_{rand}, q_{obs}) \end{cases} \quad (8)$$

F_{random} 是一个随机方向上的扰动,目的是跳出局部最小值。 ε 是设定的阈值, $\rho(q_{rand}, q_{obs})$ 是随机点与障碍物的欧几里得距离。

4.3. 改进连接方式

改进连接方式。当两棵树最新节点之间没有与障碍物发生碰撞,则直接连接。如算法 2 所示。

算法 2. Connect 算法

Input : $T_{tree}, target, F_{obs}$

Output : $T_{tree}, connected$

1. $connected \leftarrow false$;
2. $(q_{near}, q_{near_id}) \leftarrow Near(T_{tree}, target)$;
3. if CollisionFree($q_{near}, target, F_{obs}$)
4. $T_{tree} \leftarrow Fuz(target, q_{near_id})$;
5. $connected \leftarrow true$;
6. End if;

相关解释说明, q_{near_id} 为最近节点的索引值, $target$ 用来判断是否相连的标签。Fuz 函数用来更新树的节点和索引值。

4.4. 路径优化算法

三次样条插值是一种分段式插值方法,通过一系列基于三次多项式的插值点区间,形成一条光滑曲线。用三次样条插值方法拟合出的无人机移动路径曲线更加平滑,这样可以保证无人机在急停或急转时有较好的动力学特性。

设在区间 $[a, b]$ 上取 $n+1$ 个节点, $a = x_0 < x_1 < x_2 < \dots < x_n = b$, 给定这些节点的函数值 $f(x_i) = f_i$, $i = 0, 1, 2, \dots, n$, 若 $s(x)$ 满足以下条件:

- 1) $s(x) \in C^2[a, b]$;
- 2) $s(x_i) = f_i, i = 0, 1, 2, \dots, n$;
- 3) 在每个小区间 $[x_i, x_{i+1}]$ 上, $s(x)$ 是三次多项式。则称 $s(x)$ 为三次样条插值函数。

本文中该算法产生的路径节点通过三次样条插值进行了优化,以获得适合无人机飞行的光滑飞行路径。无人机路径规划应满足两个条件:

- 1) 不能与障碍物发生碰撞;
- 2) 路径长度应尽可能短。

本文就以满足上述条件的无碰撞的路径长度最短作为适应度函数的评价标准。

本文构造的适应度函数为:

$$z = L(1 + \varpi + \eta) \quad (9)$$

其中, ϖ 是一个足够大的数字,可以排除穿过障碍区域的路径,在这里它的值为 100。

$$L(p) = \sum_{i=1}^{n-1} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \quad (10)$$

其中, $L(p)$ 是路径长度, (x_i, y_i) 是 i 路径点, n 是一个终点。 η 是一个标志变量, 其初始值设置为 0。

4.5. 参数选取

为了使 BI-APF-RRT 算法在实际应用中达到最佳性能, 对本算法中引入的关键参数进行深入理解与合理设置至关重要。本节将详细讨论主要参数的作用、选取方法及其对算法性能的影响。

本文提出的 BI-APF-RRT 算法主要涉及以下关键参数。 M : 目的控制随机采样时直接选择目标点作为采样点的概率。 M 值选取越接近 1, 树生长方向越朝向目标点。 k_a (引力增益系数)与 k_r (斥力增益系数): 目的控制人工势场中引力场与斥力场的相对强度。在简单或者开阔环境中可使用较小的 k_r/k_a 比值(如 1.5~2.0), 让引力主导, 加快收敛。反之在复杂或者狭窄环境, 需使用较大的 k_r/k_a 比值(如 3.0~5.0), 增强斥力以确保安全, 避免碰撞。 ε 设定阈值的目的是防止算法在无解或极端复杂环境中无限循环, 是保证算法实时性的硬性约束。 ε 通常设置为一个较小的值(如 0.05 到 0.2)。这是因为过高的 ε 会使算法退化为贪婪搜索, 在存在“陷阱区域”(如 U 型障碍)时极易失败。

5. 仿真结果

本文使用 MATLAB 作为仿真软件。比较了改进的 BI-APF-RRT、APF-RRT、APF-RRT*的仿真结果以及现有的 RRT、RRT*和 BI-RRT*算法的仿真结果。事实证明, 改进的 BI-APF-RRT 具有更快的收敛性和较短路径的优势。模拟环境是二维空间, 其尺寸为 $100 \text{ km} \times 100 \text{ km}$ 。模拟环境如图 3 所示。

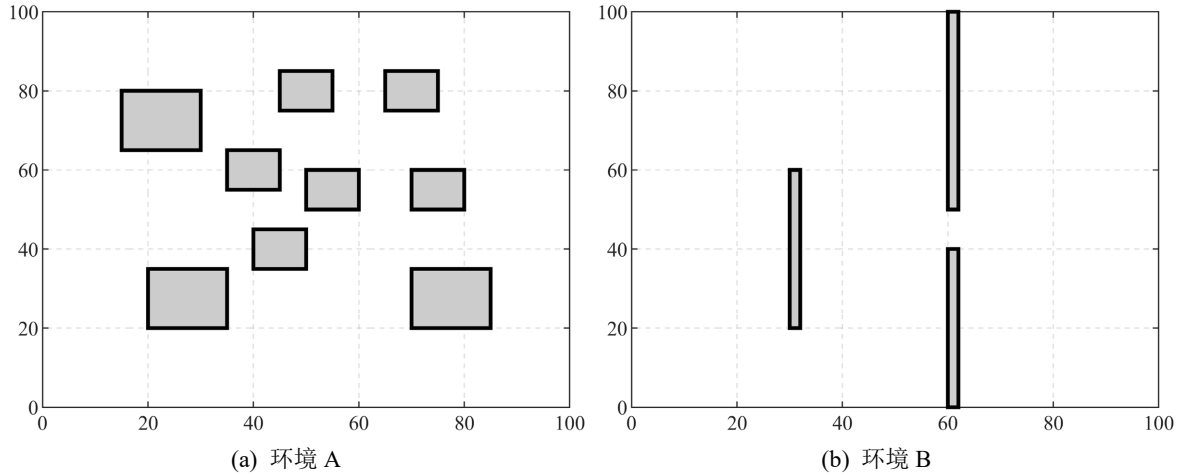


Figure 3. Simulation environment
图 3. 仿真环境

在模拟环境 A 中, 无人机开始于(5, 5), 并以(95, 95)结束。在模拟环境 B 中, 无人机开始于(5, 55), 并以(95, 55)结束。无人机无法穿越深灰色的障碍。 q_{near} 和 q_{new} 之间的连接由红线表示。在双向搜索树中, 绿线表示另一个生长树 q_{near} 和 q_{new} 之间的连接。

5.1. 环境 A

环境 A 的仿真结果如图 4(a)所示, 红线表示路径规划的无人机的飞行轨迹。在图 4(a)中, 由于 RRT 的抽样空间较大, 该轨迹具有更多的拐点和更长的路径。图 4(b)显示了 RRT*算法的仿真结果。与原始的

RRT 算法相比, 尽管搜索时间更长, 但路径得到优化, 去除多余冗余点。尽管 RRT*算法改善了路径, 但由于缺乏障碍排除, 该算法在障碍物周围具有多余的点。因此, 加入人工势场法后, 图 4(c) APF-RRT 算法和图 4(d) APF-RRT*算法, 与 RRT 算法和 RRT*算法相比, 明显提高了算法的搜索效率; 在图 4(e)中, BI-RRT 双向随机生长树的生长速度更快, 但路径不够平滑。图 4(f)显示了本文中给出的算法, 黑色轨迹为原始路径, 红色轨迹为经过 B 样条曲线优化得来, 能明显看出 BI-APF-RRT 算法路径更平滑且较短。

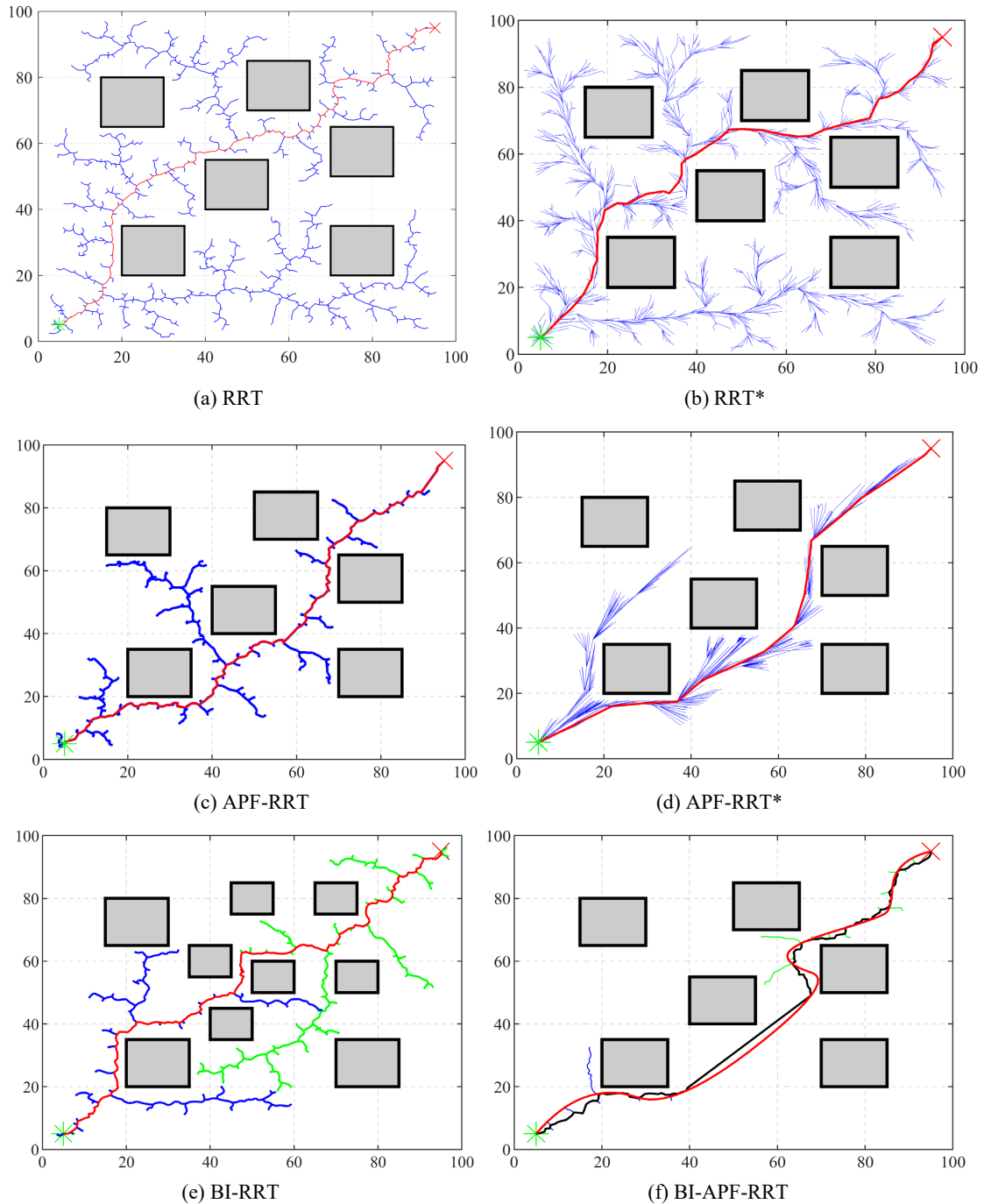


Figure 4. Simulation results in environment A
图 4. 环境 A 仿真结果

每种算法在复杂的环境进行了 30 次测试, 以验证算法的稳定性, 环境 A 中的仿真数据见图 5。通过分析图 5 中的数据, 能明显看到本文算法相比其他算法在路径长度、迭代次数、运行时间方面有明显的优势。

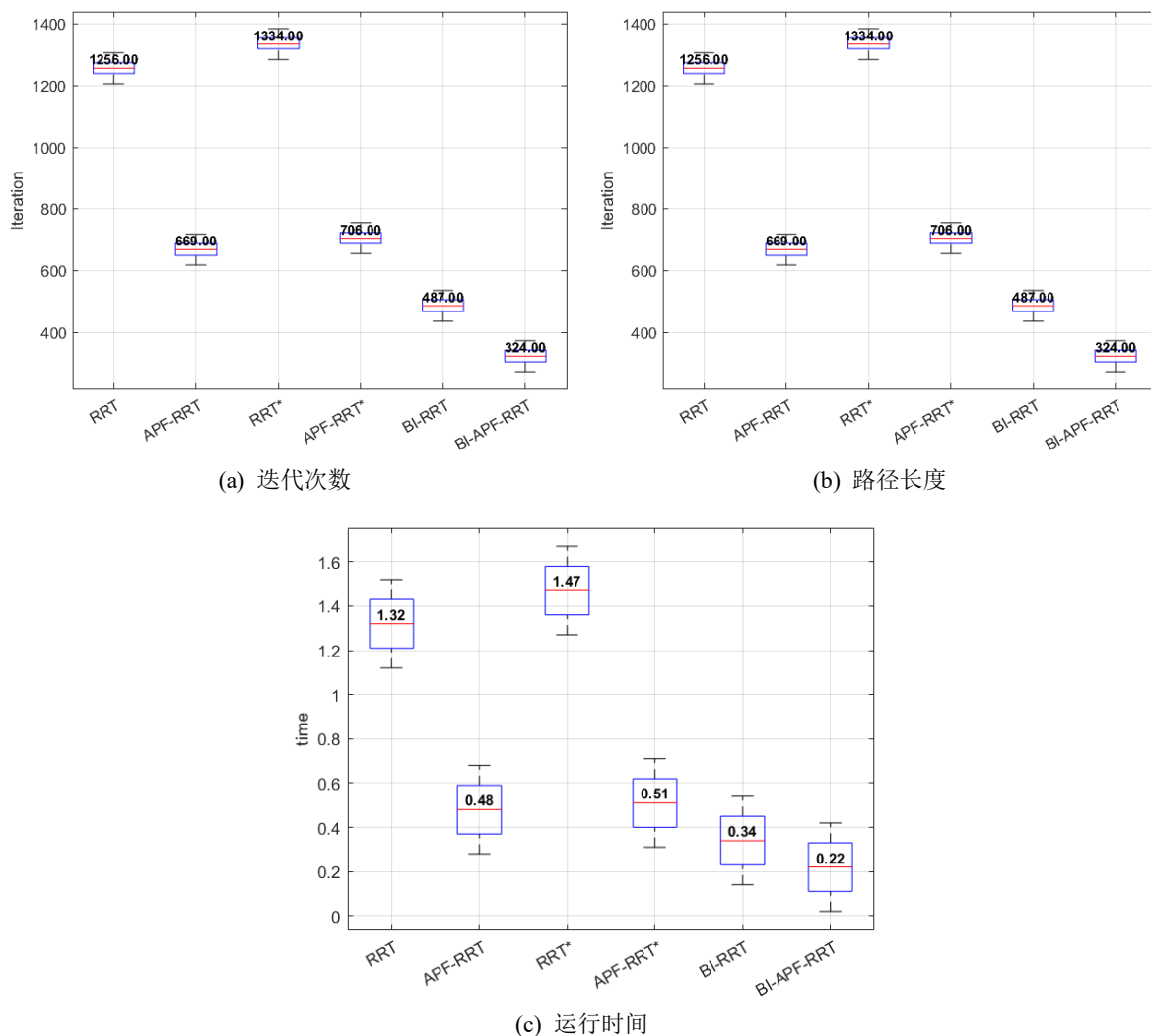


Figure 5. Data comparison in complex environments

图 5. 复杂环境下数据对比

通过分析表 1, 能看到 BI-APF-RRT 算法相比 APF-RRT 算法产生轨迹的平均长度减少 20.5%, 平均运行时间减少 54%, 平均迭代次数降低了 51%。BI-APF-RRT 算法与 BI-RRT 算法相比产生轨迹的平均长度减少 23.9%, 平均运行时间减少 35%, 平均迭代次数降低 33%, 这表明该算法在复杂的环境中具有更高的搜索效率, 并且该算法路径更平滑。

5.2. 环境 B

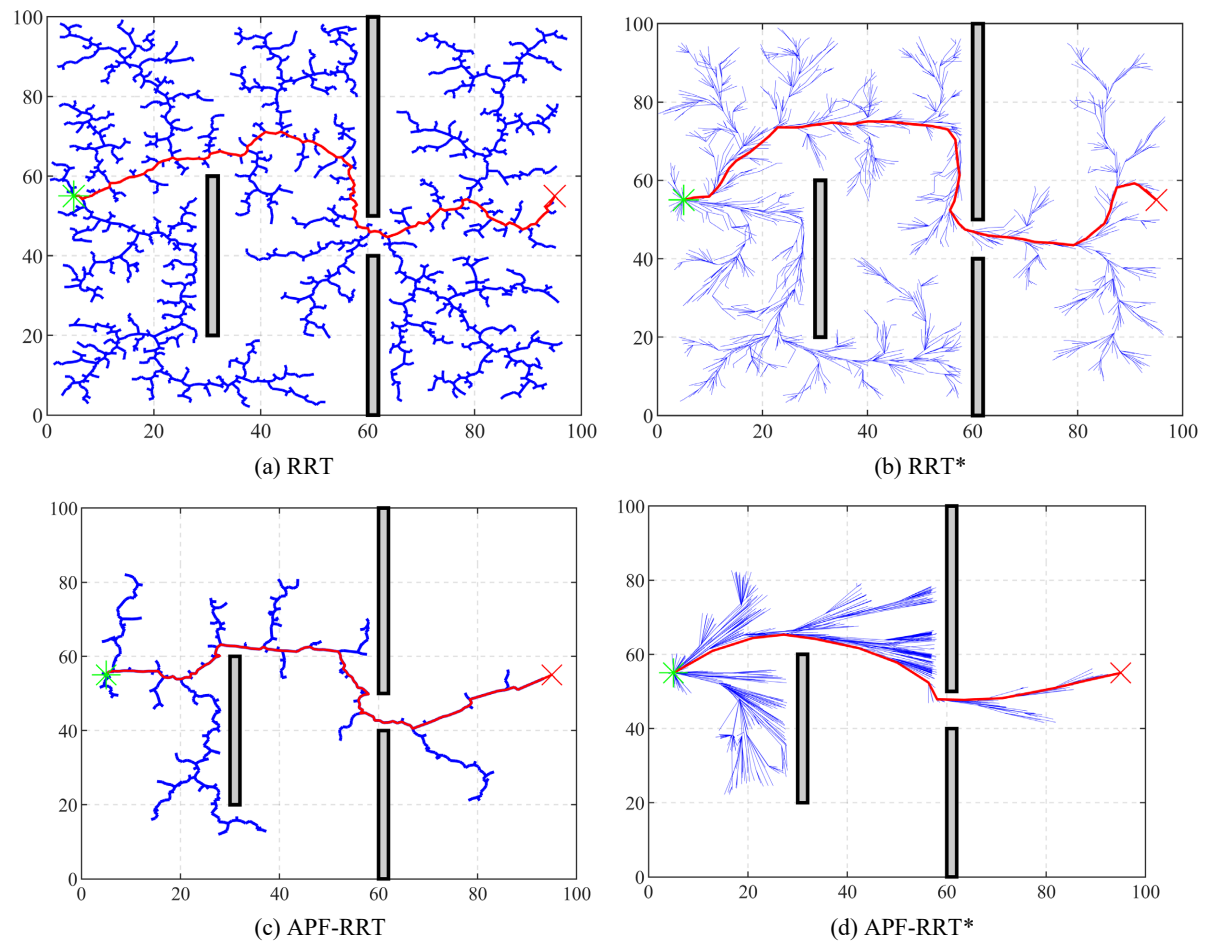
在环境 B 中, 图 6 中显示了 6 种算法计划的轨迹。本文中新算法的飞行路径如图 6(f)所示。黑色轨迹为原始路径, 红色轨迹为经过 B 样条曲线优化得来。与其他算法相比, 生成的冗余点较少, 飞行路径更符合实际需求。

每种算法在狭窄的环境进行了 30 次测试，以验证算法的稳定性，环境 B 中的仿真数据见图 7。通过分析图 7 中的数据，能明显看到本文算法相比其他算法在路径长度、迭代次数、运行时间方面有明显的优势。

通过分析表 2，本文算法产生轨迹的平均长度比 APF-RRT 算法减少 21.4%，平均运行时间减少 52%，平均迭代时间降低 52%，本文算法产生的轨迹平均长度比 BI-RRT 算法少 28%，平均运行时间减少 36%，平均迭代时间降低 15%，这表明该算法在狭窄的环境中具有更高的搜索效率，并且该算法路径更平滑。

Table 1. Experimental results of finding feasible solutions in complex environments
表 1. 复杂环境下找到可行解的实验结果

算法	平均路径长度/m	平均运行时间/s	平均迭代次数
RRT	166	1.32	1256
APF-RRT	156	0.48	669
RRT*	142	1.47	1334
APF-RRT*	138	0.51	706
BI-RRT	163	0.34	487
BI-APF-RRT	124	0.22	324



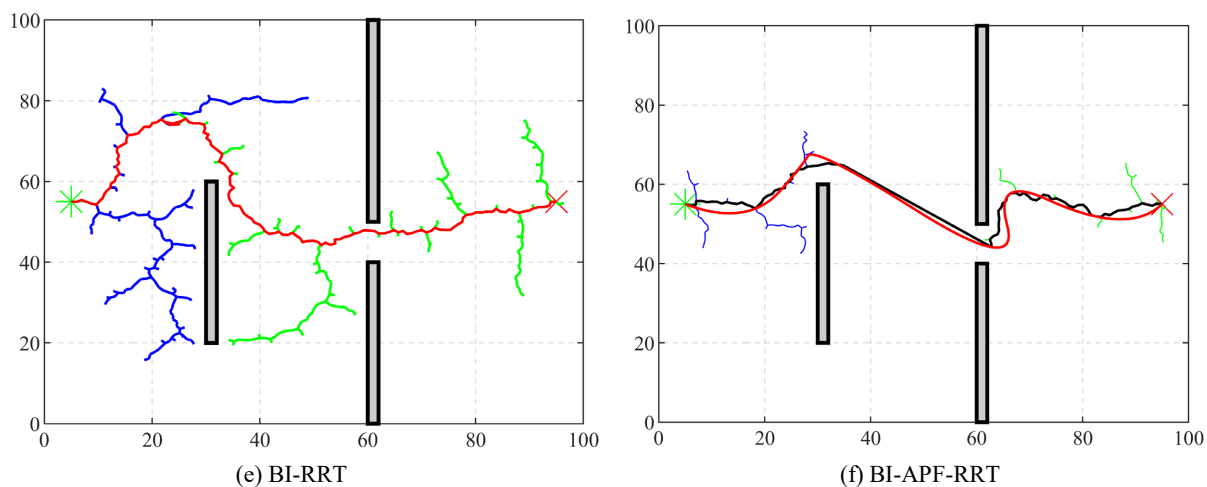


Figure 6. Simulation results in environment B
图 6. 环境 B 仿真结果

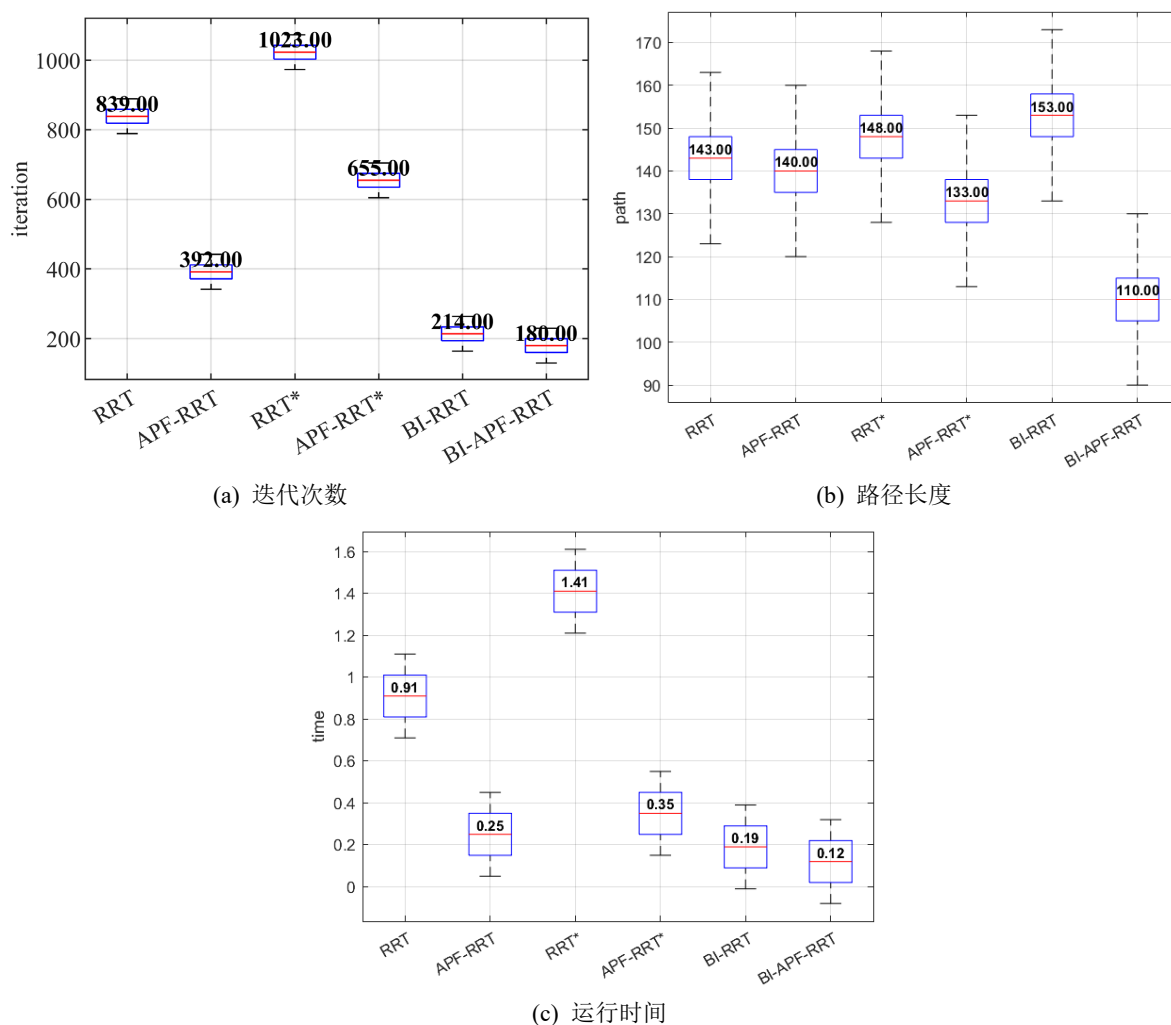


Figure 7. Data comparison in confined environments
图 7. 狭窄环境下数据对比

Table 2. Experimental results of finding feasible solutions in confined environments
表 2. 狭窄环境下找到可行解的实验结果

算法	平均路径长度/m	平均运行时间/s	平均迭代次数
RRT	143	0.91	1982
APF-RRT	140	0.25	382
RRT*	148	1.41	2215
APF-RRT*	133	0.35	655
BI-RRT	153	0.19	214
BI-APF-RRT	110	0.12	180

6. 结束语

在双向 RRT 的基础上, 本文引入了目标偏置策略以改善随机抽样点的产生。然后, 将改进的人工势场引入 BI-RRT 算法, 实现了选择质量更好的节点, 并将其添加到生长树中。改进的算法减少了不必要的冗余点, 降低了迭代次数, 进而解决了 RRT 算法中路径太长的问題, 算法的搜索效率得到有效提升。仿真结果表明: 算法能够有效优化路径长度并提高搜索速率, 较好地解决了双向 RRT 算法中存在较大搜索范围随机性问题。后期, 将进一步研究 3D 环境中的多架无人机合作运输问题。

参考文献

- [1] Ni, Z., Li, Q. and Zhang, M. (2025) Efficient Motion Planning for Chili Flower Pollination Mechanism Based on BI-RRT. *Computers and Electronics in Agriculture*, **232**, Article ID: 110063. <https://doi.org/10.1016/j.compag.2025.110063>
- [2] 崔锡杰, 王晓军, 李晓航. 改进 RRT 算法的机器人全局路径规划[J]. 计算机工程与应用, 2025, 61(4): 331-338.
- [3] 张启钱, 许卫卫, 张洪海. 复杂低空物流无人机路径规划[J]. 北京航空航天大学学报, 2020, 46(7): 1275-1286.
- [4] 栾庆磊, 郭继智, 屈紫浩, 史艳琼, 陈中. 改进 RRT 算法的机械臂路径规划[J]. 组合机床与自动化加工技术, 2024(5): 60-64.
- [5] 赵棣宇, 郑宾, 殷云华, 等. 改进粒子群算法的 UAV 突防路径规划[J]. 电光与控制, 2023, 30(4): 12-16.
- [6] 巩慧, 倪翠, 王朋, 等. 基于 Dijkstra 算法的平滑路径规划方法[J]. 北京航空航天大学学报, 2024, 50(2): 535-541.
- [7] 李皓婧. 采用新型蚁群算法的 UAV 动态航迹规划[J]. 电子器件, 2017, 40(1): 130-135.
- [8] Hart, P., Nilsson, N. and Raphael, B. (1968) A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics*, **4**, 100-107. <https://doi.org/10.1109/tssc.1968.300136>
- [9] Rahman, A., Kundu, A. and Banerjee, S. (2025) IQ-RRT*: A Path Planning Algorithm Based on Informed-RRT* and Quick-RRT*. *International Journal of Computational Science and Engineering*, **28**, 303-313. <https://doi.org/10.1504/ijcse.2025.146087>
- [10] 陈侠, 刘奎武, 毛海亮. 基于 APF-RRT 算法的无人机航迹规划[J]. 电光与控制, 2022, 29(5): 17-22.
- [11] Wang, H., Zhou, X., Li, J., Yang, Z. and Cao, L. (2024) Improved RRT* Algorithm for Disinfecting Robot Path Planning. *Sensors*, **24**, Article 1520. <https://doi.org/10.3390/s24051520>
- [12] Suwoyo, H., Adriansyah, A., Andika, J., Ubaidillah, A. and Zakaria, M.F. (2023) An Integrated RRT*SMART-A* Algorithm for Solving the Global Path Planning Problem in a Static Environment. *IJUM Engineering Journal*, **24**, 269-284. <https://doi.org/10.31436/ijumej.v24i1.2529>
- [13] 靳午煊, 马向华, 赵金良. 改进 Informed-RRT* 的移动机器人路径规划算法研究[J]. 计算机工程与应用, 2023, 59(19): 75-81.
- [14] Wang, L., Zhang, Y. and Guo, C. (2025) Path Planning for a Prostate Intervention Robot Based on an Improved Bi-RRT Algorithm. *IEEE/ASME Transactions on Mechatronics*, **30**, 668-678. <https://doi.org/10.1109/tmech.2024.3396872>
- [15] 刘建宇, 范平清. 基于改进的 RRT*-Connect 算法机械臂路径规划[J]. 计算机工程与应用, 2021, 57(6): 274-278.
- [16] Cao, M., Mao, H., Tang, X., Sun, Y. and Chen, T. (2025) A Novel RRT*-Connect Algorithm for Path Planning on Robotic Arm Collision Avoidance. *Scientific Reports*, **15**, Article No. 2836. <https://doi.org/10.1038/s41598-025-87113-5>

-
- [17] Zhang, M., Chen, Y., Luo, S., Li, Q., Zhao, H. and Zu, L. (2024) Path Planning of the Robotic Manipulator Based on an Improved Bi-RRT. *IEEE Sensors Journal*, **24**, 31245-31261. <https://doi.org/10.1109/jsen.2024.3441622>
 - [18] Fan, J., Chen, X. and Liang, X. (2023) UAV Trajectory Planning Based on Bi-Directional APF-RRT* Algorithm with Goal-Biased. *Expert Systems with Applications*, **213**, Article ID: 119137. <https://doi.org/10.1016/j.eswa.2022.119137>
 - [19] Guo, S., Gong, J., Shen, H., Yuan, L., Wei, W. and Long, Y. (2025) DBVSB-P-RRT*: A Path Planning Algorithm for Mobile Robot with High Environmental Adaptability and Ultra-High Speed Planning. *Expert Systems with Applications*, **266**, Article ID: 126123. <https://doi.org/10.1016/j.eswa.2024.126123>
 - [20] 刘奥博, 袁杰. 目标偏置双向 RRT*算法的机器人路径规划[J]. 计算机工程与应用, 2022, 58(6): 234-240.