Heliyon 10 (2024) e32451

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Review article

Recent advances in Rapidly-exploring random tree: A review

Tong Xu

School of Information Technology, Jiangsu Open University, Nanjing, 210000, China

ARTICLE INFO

Keywords:
Rapidly-exploring random tree
Branching strategy improvement
Sampling strategy improvement
Post-processing
Model driven
Robot

ABSTRACT

Path planning is an crucial research area in robotics. Compared to other path planning algorithms, the Rapidly-exploring Random Tree (RRT) algorithm possesses both search and random sampling properties, and thus has more potential to generate high-quality paths that can balance the global optimum and local optimum. This paper reviews the research on RRT-based improved algorithms from 2021 to 2023, including theoretical improvements and application implementations. At the theoretical level, branching strategy improvement, sampling strategy improvement, post-processing improvement, and model-driven RRT are highlighted, at the application level, application scenarios of RRT under welding robots, assembly robots, search and rescue robots, surgical robots, free-floating space robots, and inspection robots are detailed, and finally, many challenges faced by RRT at both the theoretical and application levels are summarized. This review suggests that although RRT-based improved algorithms has advantages in large-scale scenarios, real-time performance, and uncertain environments, and some strategies that are difficult to be quantitatively described can be designed based on model-driven RRT, RRT-based improved algorithms still suffer from the problems of difficult to design the hyper-parameters and weak generalization, and in the practical application level, the reliability and accuracy of the hardware such as controllers, actuators, sensors, communication, power supply and data acquisition efficiency all pose challenges to the long-term stability of RRT in large-scale unstructured scenarios. As a part of autonomous robots, the upper limit of RRT path planning performance also depends on the robot localization and scene modeling performance, and there are still architectural and strategic choices in multi-robot collaboration, in addition to the ethics and morality that has to be faced. To address the above issues, I believe that multi-type robot collaboration, human-robot collaboration, real-time path planning, self-tuning of hyper-parameters, task- or application-scene oriented algorithms and hardware design, and path planning in highly dynamic environments are future trends.

1. Introduction

Path planning is one of the extremely important problems in robotics, and one definition is that a robot works autonomously to plan a path from the initial point to the goal point in a certain map scenario under the premise of ensuring that no collision occurs with other obstacles in the map. Challenges in this field include the difficulty of modeling the environment, convergence of algorithm, and avoiding local optimal solutions. In addition, most traditional path planning algorithms are designed for scenarios with known maps and may not be suitable for use in unfamiliar environments.

The classical path planning algorithms contain a search strategy represented by A* and a sampling strategy represented by

E-mail address: xutong0901@126.com.

https://doi.org/10.1016/j.heliyon.2024.e32451
Received 5 February 2024; Received in revised form 15 May 2024; Accepted 4 June 2024
Available online 7 June 2024

probabilistic roadmap (PRM), the former relies on the current position and historical position information, and thus is prone to local optimal solutions; the latter makes use of the global information of the map, which is possible to get the optimal solution, but this method lacks the optimization strategy of the local paths, which leads to the redundancy of path generation. The Rapidly-exploring Random Tree (RRT) [1,2] algorithm combines the characteristics of search and sampling: Its search characteristics are manifested in the fact that the algorithm starts from the root node and keeps branching, growing like a tree until it searches for the target node; its sampling characteristics are manifested in the fact that the algorithm is affected by the random sampling points in the process of branching, so the RRT improvement algorithm has the potential to achieve a better performance than the other algorithms. Theoretically, the hyper-parameters or strategies involved in the RRT algorithm, such as the setting of the search step size, the selection of branching nodes and the design of branching strategies, can be designed to address different optimization objectives; application-wise, the improvement strategies of RRT also needs to be designed to address the objectives and constraints of specific application scenarios; however, both the theoretical optimization itself and the leap from theory to reality are facing challenges. At the theoretical level, we need to know whether the RRT-related improved algorithms and the model-driven RRT are capable of searching large-scale scenarios? How is the real-time performance of the improved RRT algorithms? Can the improved RRT algorithms cope with uncertain environments? Has the path quality been further improved? At the application level, we need to know in which fields the improved RRT algorithm has been applied? What are the results? What problems are still faced in real scenarios? Therefore, it is crucial to review the recent advances in the RRT-related literature at both the theoretical and application levels, summarize the challenges mentioned in the literature, and look ahead to future work.

Based on the above motivations, the layout of this paper is as follows: In section 2, the advantages and disadvantages of RRT and other traditional path planning algorithms are compared. In order to show the improvement strategies of the RRT algorithm more clearly in the subsequent sections, this section also describes the implementation process of the traditional RRT algorithm in detail. Section 3 describes in detail the research progress of the RRT improvement algorithm at the theoretical level, from the aspects of branching strategy improvement, sampling strategy improvement, post-processing, model-driven RRT, and summarizes the advantages of the theoretical improvement in large-scale scenario search, real-time performance, uncertain environments, and path quality improvement. Section 4 reviews the achievements of RRT-related algorithms on real robots, including arc welding robots, assembly robots, search and rescue robots, surgical robots, free-floating space robots, mining robots, inspection robots, and interactive robots. Section 5 summarizes the challenges faced by the algorithms at the level of real-world applications, as well as future research trends.

2. Rapidly-exploring random tree

2.1. Fundamentals of rapidly-exploring random tree

The Rapidly-exploring Random Tree (RRT) algorithm commonly referred to in the field is actually heuristically biasing RRT (HBRRT) [3], target-biased RRT (TBRRT) [4–7], goal-biased RRT (GBRRT) [8–11], goal-oriented RRT (GORRT) [12–14] or goal-directed RRT algorithm (GDRRT) [15–18], the idea of this algorithm is to take the initial position as the root node and then add leaf nodes by random sampling. When the leaf nodes of the random tree arrive at the target position, the path from the initial position to the goal position is planned. The implementation of the RRT algorithm is shown in Algorithm 1: Firstly, randomly scatter the node $\boldsymbol{q}_{\mathbf{rand}}$, then call the function **FindQNear** to find the nearest node $\boldsymbol{q}_{\mathbf{near}}$ to $\boldsymbol{q}_{\mathbf{rand}}$, then call the function **FindQNew** to get the new node $\boldsymbol{q}_{\mathbf{new}}$, next, judge whether the generated the new node $\boldsymbol{q}_{\mathbf{new}}$ has collision with the obstacles in the map environment, if not, then a space search extension is successfully completed and the new node $\boldsymbol{q}_{\mathbf{new}}$ is determined, on the contrary, $\boldsymbol{q}_{\mathbf{new}}$ needs to be discarded. Finally, repeat the above process until $\boldsymbol{q}_{\mathbf{new}}$ reaches the defined desired goal position $\boldsymbol{q}_{\mathbf{goal}}$.

Algorithm 1: Goal-Directed RRT (GDRRT)

Input: $K \in \mathbb{N}$ $q_{init} \in C_{free}$ $q_{goal} \in Q_{goal} \in C_{free}$
Output: G

```
V ⟵ {q_init} E ⟵ {∅} G ⟵ (V,E)
for k ⟵ 1 to K do
    if rand() < p1 then
        q_rand ⟵ q_goal
    else
        q_rand ⟵RANDOMSTATE
    q_near ⟵FindQNear(q_rand,V)
    (q_new,success)⟵FindQNew(q_near,q_rand)
    if success then
        V⟵V∪{q_new},  E⟵E∪{q_near, q_new}
        if q_new == q_goal or q_goal is on the edge then
            if q_new ~= q_goal then
                V⟵V\{q_new}, V⟵V∪{q_goal}
            path ⟵ TRUE
            G ⟵ (V,E)
Return G  if path == TRUE; else failure
```

In the traditional RRT algorithm, among all the existing nodes, the strategy to select the node $\boldsymbol{q}_{\mathbf{near}}$ that is currently the most suitable for branching is to choose the Euclidean distance as the cost function, from which the node with the minimum surrogate value is selected; and the strategy for generating the path from $\boldsymbol{q}_{\mathbf{near}}$ to $\boldsymbol{q}_{\mathbf{new}}$ is to connect these two nodes directly, thus generating a line segment. In an ideal path planning scenario, these two strategies of the traditional RRT algorithm seem to be the optimal solution, but

Table 1
Performance comparison of classical algorithms.

Algorithms	Principle	Advantages	Disadvantages
Dijkstra	Expand outward, layer by layer, centered on the starting point, until it reaches the target point.	Optimal paths can be found	Too many nodes traversed
A*	The cost function is designed based on the Dijkstra algorithm	Improved search efficiency compared to the Dijkstra algorithm	computationally expensive
D*	An improved version of A*	For dynamic path planning problems, it reduces the repeated computation of the same data by the A* algorithm and is an improved version of the A* algorithm.	Not applicable to changes occurring on the shortest path over long distances
Visibility graph	Connect the obstacle to the target point with a line segment and remove the connecting line through the obstacle; the remaining connecting line is the desired visual graph.	The shortest path can be solved and path generation is also efficient	Algorithm complexity is limited by the shape and number of obstacles, and the representation of environmental information is too idealized
APF	Take the target point as the gravitational force and the obstacle as the repulsive force, and plan the obstacle avoidance path through the resultant force	Path generation is efficient	Prone to local minima and unable to find paths between neighboring obstacles
GA	Mimicking the process of natural selection and reproduction	Strong global search capability for suboptimal solutions	weak local search capability, very long time to get the optimal solution
PSO	Mimicking a flock of birds to share information	Fast convergence	premature convergence, easy to fall into local optimal solutions
ACO	Mimicking an ant colony to share information	Fast convergence	Complex parameter settings leading to deviation from high quality solutions
PRM	Random sampling in obstacle maps	Plan paths in higher dimensional spaces; Paths can be conform to the constraints of the robot's kinematic model	Inefficient search in complex environments; Random sampling leads to redundancy of most nodes and increases computational cost
RRT	Build a tree that grows and spreads in all directions with the starting point as the root node, and nodes are also obtained by random sampling in the obstacle map	In contrast to the PRM algorithm, the random sampling process can be oriented by target bias probability	Inefficient search in complex environments;

most of the autonomous robots in real scenarios, such as unmanned aerial vehicle (UAV), self-driving cars, unmanned surface vessel, etc., belong to non-holonomic constraint systems, and thus, scholars do the following thinking: (1) Is there a better strategy for selecting $\boldsymbol{q}_{\mathbf{near}}$? (2) Is there a better strategy to connect $\boldsymbol{q}_{\mathbf{near}}$ and $\boldsymbol{q}_{\mathbf{new}}$? In addition, it may also be worthwhile to further investigate these issues involved in the GDRRT algorithm: (3) Can the feasible solutions generated by the GDRRT algorithm be further optimized? (shorter, smoother and traceable)? (4) How can the algorithm can be improved in cases where the map is not globally known? (5) Whether hotly debated large-scale models can also be incorporated into RRTs. This review focuses on these improvement strategies for RRT algorithms, presenting recent advances in the last three years (2021–2023).

2.2. Performance comparison

Table 1 summarizes the advantages and disadvantages of some classical path planning algorithms. These algorithms can be classified into search-based and sampling-based strategies. Representative search-based algorithms include Dijkstra, A*, D*, etc. Among them, some use traversal-based search, such as breadth first search (BFS) and Dijkstra, thus have high cost when the target point is far away from the initial position; some are cost-function oriented, such as the A*, which expects to find the relatively optimal path quickly, but search-based algorithms rely on information about current and historical positions, and thus are prone to have locally optimal solutions. Such methods have their inherent limitations. The representative sampling-based algorithm is probabilistic roadmap (PRM), which employs a completely randomized node search and determines the feasibility of the generated paths based on the presence of obstacles between nodes, utilizing the global information of the map and therefore possessing the potential to obtain optimal solutions. However, this random search lacking objectives leads to redundancy of generated nodes and lacks theoretical basis for optimizing local paths. To solve the problems above, strategies that fuse search-based and sampling-based strategies are widely used, the most representative of which is the familiar RRT.

In addition, some strategies from other fields have been introduced into the field of path planning. For example, artificial potential field (APF), genetic algorithm (GA), particle swarm optimization (PSO), ant colony optimization (ACO) in the field of optimization, deep deterministic policy gradient (DDPG) in the field of deep learning, all of these have been applied to path planning with good results.

Overall, RRT is a hybrid algorithm which contains both searching and sampling features, its searching feature is manifested in the fact that RRT starts branching from the root node and finds feasible paths connecting the start and end points in the obstacle map, and its sampling feature is manifested in the fact that the branching process is affected by probability, so the algorithm has the potential to achieve better performance metrics compared with searching features-based or sampling features-based algorithms, and furthermore, the development of other fields such as optimization and artificial intelligence also contributes to the development of RRT.

3. Overview of RRT-based algorithm improvements

The stochastic nature of the branch makes it possible to further optimize the algorithm. At the theoretical level, it is generally necessary to study the optimality of the paths, and at the practical level, it is generally necessary to weigh the scene constraints, model constraints, and other constraints, and design the most suitable paths according to the established requirements.

In response to the large amount of literature emerging on RRT-based improvement algorithms, this review discusses four aspects, including branching strategy improvement, sampling strategy improvement, post-processing, and model-driven RRT, and looks into several possible future research directions in this field.

3.1. Branching strategy improvement

3.1.1. Branch node selection

There are no more classical algorithms for branch node selection than RRT* [19], which adds a two-step optimization strategy including 'rewire' and 'random relink', where rewire means that after a new node is added to the tree, a parent node is re-selected for it, so that the cost value of the newly generated path is smaller, and random relink means that after rewiring, the nodes surrounding the new node are reconnected to the new node. Karaman et al. [20] prove the asymptotic optimality of the RRT* algorithm, implying that the algorithm can find the optimal path when the time tends to infinity.

Some researchers suggest improvements to the algorithm in response to the shortcomings of RRT* in specific application scenarios. Aiming at the problem that many RRT*-based variants are inefficient in sampling and slow in convergence in the environment consisting of long corridors, Ding et al. [21] propose the expanding path RRT* (EP-RRT*) based on heuristic sampling in the path expansion area, and experiments show that compared with RRT* and Informed-RRT*, the proposed EP-RRT* improves the node utilization, speeds up the convergence rate, and obtains better paths for the same number of iterations. Reconfigurable modular robotic systems under RRT* are inefficient due to the formation variety. For this reason, Lu et al. [22] propose an obstacle-aware hybrid density network to guide the generation of polygonal nodes, and the results show that the strategy of connecting convex polygon trees* (CPT*) in the form of RRT* improves scalability to large environments. To minimize the energy consumption of the robot, Yu et al. [23] propose a cylinder-based informed-RRT* (Cyl-iRRT*) algorithm, which seeks to find the optimal homotopy path by focusing the search space on the designed gradually shrinking cylinder. Alam et al. [24] present a pick-and-place RRT* under the novel flight cost (FC-RRT*), which generates nodes in a predetermined direction and then calculates the energy consumption using the circle-point method. Modular self-reconfiguring robots can change their configurations to efficiently adapt to various tasks. To address the characteristics of such robots, Odem et al. [25] defined RRT*-based topological configurations, a new approach that represents a set of modular self-reconfiguring robot equivalent configurations as topological configurations, thus significantly reducing the tree size. The Quick-RRT* algorithm proposed by Jeong et al. [26] reselects the parent nodes and optimizes the pruning range, and the superiority of the algorithm is verified by calculating the space complexity and time complexity.

RRT-connect [27,28] also known as bidirectional RRT (B-RRT), introduces a dual-tree expansion mechanism that simultaneously expands the random tree with the initial and goal points as root nodes, respectively, Rajendran et al. [29] present RRT-Connect that incorporates human awareness, and paths generated by the algorithm are as visible as possible to humans, making it easier for human-robot collaboration. Chen et al. [30] fuse RRT-Connect and Bezier curves and apply them to a cooperative assembly system with two robotic arms composed of six degrees of freedom, effectively optimizing the trajectories of the dual robotic arms. Li et al. [31] notice that Rapidly-exploring random vines (RRV) [32] performs well in single narrow channel environments but poorly in cluttered environments, while RRT presents the opposite result, so they propose a fusion algorithm based on RRT-Connect and RRV. Cao et al. [33] point out the advantages of RRT-Connect, artificial potential field (APF) and cubic B-spline for global path planning, local path planning and curve smoothing, respectively and fuse these three algorithms together to optimize the paths of unmanned aerial vehicle (UAV). In Ref. [34], a novel adaptive gravity field-based RRT-connect method is proposed and the efficiency of the improved algorithm is further demonstrated on an actual manipulator platform. Li et al. [35] optimize the RRT-Connect nearest node selection mechanism, which has better adaptability in the apple orchard environment and promote the automation of orchard operations. Other application scenarios for RRT-connect include path planning of the fruit tree pruning manipulator [36], insect-like mobile robots in a narrow environment [37], UAV trajectory tracking [38] and navigation [39], multi-UAV formation shape generation [40], automatic sampling of exhaust emissions [41], closed-loop control of microrobots subjected to flowing fluid disturbances in a microfluidic environment [42], home service robot arms [43]. Several researchers investigate improved strategies for RRT-connect, Spinos et al. [44] propose the link-augmented graph to increase the range of feasible solutions for truss robots and apply it to RRT-Connect to search all target points more efficiently. Kang et al. [45] propose a rewiring method based on triangular inequalities to bring RRT-Connect closer to the optimum. To deal with manipulator path planning in complex multi-obstacle environment, Petit et al. [46] introduces a new method called RRT-Rope, which builds on RRT-connect by using a deterministic shortcut technique for fast post-processing while adding intermediate nodes at branches of the tree.

The study of the fusion of RRT-connect and RRT*-based algorithms is also a branch of the study of improved RRT-based algorithms. The B-RRT* algorithm [47] directly fuses RRT-connect and RRT* and also inherits the advantages of both algorithms. To improve the obstacle avoidance efficiency of the redundant robotic arm, Dai et al. [8] propose a B-RRT* based on a novel potential field guidance (PB-RRT*), and the results show that the proposed algorithm plan shorter paths, larger gap between the manipulator and the obstacle and fewer invalid nodes when compared with the B-RRT*. Singh et al. [48] incorporate B-RRT and a modified Bezier curve, and try and test the technique in various real-world experiments. Shama et al. [49] use a probabilistic Gaussian mixture model to identify the regions most likely to generate nodes for faster convergence. Wang et al. [50] propose a growth point evaluation function based on the adaptive resolution octree map, which guides the generation of RRT paths to make the growth strategy purposeful, and then the algorithm also reselects the parent and candidate nodes and rewires them. The experimental results show that the improved RRT algorithm can eliminate the redundant bifurcations of the growing tree, reduce the number of sampling times, and greatly improve the growth efficiency compared with the traditional RRT, RRT* and B-RRT*. Other application scenarios for B-RRT or B-RRT* include ice navigation [51], arc welding robot [52], the autonomous flight of UAV [53], robot path planning in constrained environment [48,54], redundant manipulators [8,55], autonomous parking [56], lunar rover [57], litchi-picking robot [58].

There is also some literature that implements branch-node selection improvements based on RRT. In order to design trackable paths for articulated vehicles, I and my team propose a hybrid strategy combining RRT, farthest node search and head correction with fixed wheel position [16]. To solve the non-convex optimization problem, He et al. [59] devise an barrier pairs-based RRT (BP-RRT) algorithm, where each barrier pair consists of a quadratic barrier function and a full-state feedback controller, to achieve the synthesis of locally optimal controllers, which are validated in the simulation of a dual linkage manipulator robot. Hao et al. [60] present the complex environments RRT (CERRT), which restricts vertex selection and expansion, and samples the region near obstacles multiple times to avoid useless exploration. Guo et al. [61] introduce the concept of candidate point array, where when an obstacle is present on the straight line between the nearest point and the randomly generated current point, that point is not used for the current stage of path generation but for subsequent path optimization, and this strategy is applied to the Yobogo legged robot navigation. Pareto dominance is a very well known theorem in game theory, where a Pareto improvement is defined as an allocation of resources that makes at least one person better off without making anyone worse off. When using the RRT path planning algorithm, the Pareto improvement method is used to select the optimal node from dozens to hundreds candidate nodes [62], motion planning based on zero-sum games is also a similar idea [63].

3.1.2. Branch design

Branching designs are generally improved in terms of branch step size or branching mode. In terms of branch step size improvement, some researchers determine the optimal branch size through extensive experiments in a given scenario [64,65], and others give specific strategies, for example, In Ref. [6], a step growth rate is introduced into the expression of angle selection [66]. introduces the "step-size dichotomy" to solve the problem of excessively long step size in the APF algorithm due to the large range of obstacle rejection, and applies it to the motion planning of the citrus picking manipulator. In Ref. [67], the step size varies according to the density of obstacle distribution and Q-learning is used to reduce the randomness of RRT, a scenario that no longer requires accurate environment modelling and vehicle modelling. In Refs. [36,68], the adjustment coefficient is introduced and when encountering an obstacle collision, the step size becomes progressively larger, and vice versa, the branch size returns to its initial value. A similar strategy is presented in Ref. [34], where the branch size increases by a fixed length when the node is successfully extended, and vice versa, the branch size decreases by the fixed length. Another similar strategy is presented in Ref. [26], where a direct connection between ancestor nodes and the new node is considered during the rewiring procedure. In Ref. [69], the step size is adapted using the gradient descent method to make the step size smaller as the target region is approached. In Ref. [70], the dynamic step size is designed by means of a segmentation function, where the branch length varies in a logarithmic trend when it is less than a certain threshold, and is a fixed constant when it is greater than a certain threshold. In Ref. [37], a dynamic step size formula is designed based on the pattern of exponential changes. In Ref. [71], Peng et al. calculate the number of expansion failure times and use this parameter as a variable in the branch size adjustment coefficient. In Ref. [72], gravity adaptive step size strategy is applied. In terms of branch mode improvement, Nasir and Islam notice that the paths explored by RRT and RRT* are often stitched together by multiple folded segments, whereas the optimal path in the obstacle-free space is generally a straight line, and therefore design an algorithm called RRT*-Smart [73,74], which, after generating a feasible path, continuously searches for an unobstructed and direct connection to the parent node from the leaf node, turning multiple folded segments into a single folded segment. A similar strategy is that of Chung et al. [75] who argue that when encountering obstacle collision, branches extend in the same direction. In Ref. [76], a simplified rewiring step is performed, where only nodes in the set of goal trees are rewired compared to RRT*. Literature [77] proposes the cyclic pruning algorithm to shorten the paths where the first pruning is used to delete unwanted nodes and the second pruning is used to optimize the paths.

Some literature design branching mode depends on scenarios; for example, Zacchini et al. [78] predict the effect of taking more actions in unknown environments by considering branch information gain and assessing the impact of the rewiring process on tree growth. To enable the spacecraft to plan time-optimal attitude trajectories under complex pointing constraints, Xu et al. [79] propose an intelligent branching model based on geometric-level spherical and quadrilateral interpolation, which is capable of planning suboptimal solutions within a few milliseconds. Long et al. [80] propose RRT*-Smart based adaptive regional dynamics (RRT*Smart-AD) to deal with dynamic characteristics in dynamic environments. Some literature design branching modes depend on kinematic constraints, for example, Song et al. [81] develop the kinodynamic RRT*-smart algorithm for collision avoidance strategy for unmanned surface vessels. To achieve the lychee harvesting with a rigid robotic arm, Ye et al. [58] propose an adaptive weighted particle swarm optimization method. Wang et al. [82] suggest an efficient branch pruning strategy that introduces a new state while taking kinematic constraints into account and apply the algorithm to a differential drive robot. James et al. [83] develop two types of fillets to meet curvature-constrained vehicles need, a circular fillet for limiting the maximum curvature of the path and a spline fillet for achieving continuous path curvature. To address the angular limitations of RRT* for hyper-redundant manipulator applications, Ji et al. [84] design an ellipsoid-shape RRT* (E-RRT*), which adopts ellipses instead of line segments to connect adjacent nodes, and verifies the superiority of the algorithm in a narrow environment. Xu et al. [16] propose the farthest node search strategy and head correction with fixed wheel position to redesign the branch based on the articulated vehicle model. To account for the different pointing constraints faced by multiple instruments during spacecraft attitude manoeuvres, discrete quadratic path nodes are obtained using RRT* [85]. To solve the path planning problem under ocean currents [86], establish a data-driven energy model and a time-varying current model, and introduce these constraints into RRT.

3.2. Sampling strategy improvement

B-RRT is an improved RRT algorithm based on branch node selection. However, this algorithm still suffers from slow convergence, some scholars note that more subtrees may accelerate search efficiency, Chai et al. [87], Tu et al. [88] and Zhang et al. [89] propose generating multiple subtrees within a narrow channel and merging them into a main tree in subsequent sampling. Luo et al. [90] construct a third random tree based on B-RRT, which constructs intermediate points to find and converge global paths in an efficient way. In Ref. [68], a pre-harvesting guide point is designed between the start point and the end point and four random trees are generated between these three points. Some scholars make improvements based on bias probabilities [91]. proposes to construct the RRT with the probability of changes of 1 and 0 (1-0 Bg-RRT) in order to jump out the local minima in time, while [77] proposes the GM (1,1) model to optimize the background values and verifies that the proposed model has a faster convergence speed and higher stability in a narrow-channel environment as compared to the 1-0 Bg-RRT.

Some scholars notice that the slow convergence of the RRT* algorithm is due to uniform sampling of the free space, so Informed-RRT* [92] employs an elliptical sampling approach instead of globally uniform sampling. Uzun et al. [93] propose a n-sliced Informed-RRT* method, which divides the reference path into a certain number of path segments and optimizes them one by one, solving the convergence problem of Informed-RRT* in multi-curve paths. In Ref. [94], spatio-temporal Informed RRT* is proposed to provide the coordinates and velocity of each planning point in a road scene. In Ref. [95], a linear quadratic minimum time local planner-based Informed-RRT* method is proposed for a laboratory-scale 3D gantry crane. In Ref. [73], Informed-RRT* incorporates pre-harvest points and quad-tree for fast citrus harvest. In Ref. [96], Informed-RRT* incorporates dynamic window approach further reduces the global path length. To address the safety and productivity of construction sites due to temporal and spatial conflicts, a hybrid algorithm incorporating Informed-RRT*, geometry and discrete event simulation is implemented and evaluated, and experiments show that the proposed algorithm satisfies all types of obstacle avoidance constraints, provides priority for higher priority construction activities, and also obtains the shortest route and time for each transport vehicle [97]. In order to plan energy-efficient paths for reconfigurable robots in complex environments, Kyaw et al. [98] introduce an energy objective function into the batch informed trees*, where the energy objective function considers the energy cost of each reconfigurable action of the robot. In Ref. [99], the design of the short-horizon planner for collision avoidance is addressed, and an improved Informed-RRT* algorithm is proposed, which encodes the International Regulations for Preventing Collisions at Sea (COLREGs) by defining a rule-compliant region in the configuration space and searching for feasible paths within that region.

Similarly, the circle-based sampling strategy is proposed. For example, the center-circle sampling strategy based B-RRT shows the advantage of generating a smaller number of nodes [100], the forward and reverse sampling circles strategy based B-RRT demonstrates the advantages of local optimization of quadrotors trajectories in complex dynamic environments [101]. In Ref. [24], an energy-efficient industrial robot motion planning method is proposed and the energy consumption is calculated using the circle point method.

Another hotly discussed sampling strategy is Quick-RRT* [26], which uses triangular inequality for parent node selection and rewire improvement. One class of studies uses Quick-RRT* as a baseline to highlight the superiority of the proposed algorithm, for example [102], generates semi-triangular regions based on triangular inequality to improve the path quality and uses a tabu table to enhance the path generation efficiency during the rewiring process, and the results show that the proposed algorithm outperforms Quick-RRT* in terms of both path quality and path generation efficiency [103]. constructs the gravity field function based on seafloor topographic data and uses it for underwater path planning, and the results show that the proposed algorithm is more efficient under the demand of path planning considering gravity, with an efficiency of 5.98 times that of Quick-RRT*. The underwater path planning problem has also been considered in the literature [104], with a focus on rescue missions for underwater robots. The algorithm takes the ancestor node that is farthest from the sampling point and has no collision as the parent node, increases the intermediate nodes in the path depending on the step size, and uses the triangular inequality several times throughout the process to obtain the optimized path, and the results validate that the path planned by the proposed algorithm is faster compared to the Quick-RRT* algorithm. Unlike the above algorithms, Literature [105] is more concerned with robustness, and tries to adjust the sampling strategy to optimize the planned path, specifically, adjusting the size of the sector sampling region and the sampling probability through the scene, and the results verify that the algorithm is more robust than Quick-RRT*. In Ref. [106], potential function-based optimal path planning considering congestion (CCPF-RRT*) is proposed and the results show a better performance in initial solution, quick convergence speed and the movement cost compared to Quick-RRT*. Another class of studies performs further optimization based on Quick-RRT*, for example, Wang et al. [107] combine the backtracking idea of Quick-RRT*, the greedy search strategy of RRT-Connect, the triangle inequality path optimization, and verify the effectiveness of the proposed algorithm, they also propose a novel algorithm based on circular arc fillet (CAF-RRT*) [108] for the path planning problem in two-dimension workspaces, which obtains the initial path by combining Quick-RRT* with B-RRT. Qureshi et al. introduce APF to achieve a trade-off between exploration and exploitation, named P-RRT* [109], and potential functions based Quick-RRT* (PQ-RRT*) is proposed [110] to overcome the limits of the slow convergence rate of RRT*. Experiments show that the proposed algorithm guarantees a fast convergence to an optimal solution and generates a better initial solution. In Ref. [111], Quick-RRT*-based map is constructed and unsafe nodes for mobile robots are deleted from the constructed map. In Ref. [112], the virtual light-based Quick-RRT* is proposed, where in this algorithm, a sector-shaped light intensity sensing region centered on the target point is constructed [113]. optimizes the sampling area in real time based on the node density. On the application area of the Quick-RRT* algorithm, aiming at the application object of the cable-driven super-redundant manipulator with 17 degrees of freedom and the narrow and complex application environment [114], design paths based on the relationship between the maximum deflection angle and the operating speed of the manipulator, and achieve good results, Jeong et al. [115] develop an optimal obstacle avoidance path planner for stabilizing the robot's heading, and this strategy. Similar improvement strategies to Quick-RRT* are Fast-RRT* and F-RRT*. Fast-RRT* [116] combines a target bias strategy with constrained sampling to reduce the blindness of sampling, and prioritizes the ancestors of the nearest node to the root node in the branching phase. To address the issue of autonomous navigation for micro aerial vehicles, Martinez et al. [117] optimize the paths by reconnecting the nodes based on Fast-RRT*. They also develop a signed distance field to facilitate collision checking. In Ref. [4], a new fusion algorithm for target bias, gravitational potential field and Fast-RRT* is also proposed. F-RRT* [118] optimizes path creation by generating parent nodes for random points, outperforming traditional RRT's parent selection strategy. Experiments demonstrate that the algorithm surpasses RRT*-Smart [73], and Quick-RRT* [26] in terms of initial solution and fast convergence rate. Spatial offset sampling RRT* (SOF-RRT*) [7] is an enhancement of the F-RRT* algorithm, which introduces a spatially probabilistic weighted sampling strategy that increases the likelihood of sampling in regions with larger feasible regions. Cong et al. [119] propose FF-RRT*, a hybrid method that combines the target bias sampling strategy with the random sampling strategy. The branching strategy in this algorithm creates new parent nodes. Experiments demonstrate a significant reduction in convergence time when compared to the Fast-RRT* [116] and F-RRT* [118] in both simple and complex maze environments. The improved path planning algorithm considering congestion in Ref. [106] incorporate the advantages of F-RRT* and utilize the movement cost function to design an ideal path in a crowded environment. In Ref. [120], radar data is used to generate maps and the Graham algorithm is employed to delineate the dynamic flight exclusion zone (FEZ) for planning feasible paths under hazardous weather. The influence range, size and distribution characteristics of the FEZ are then investigated. Other mainstream strategies based on region sampling improvements include the generalized voronoi diagram-based sampling regions [121], non-threshold adaptive sampling region [122], Gaussian distribution and local biasing sampling regions [123], dynamic region sampling [105], the adaptive forward and backward sampling regions [124].

In addition to considering more trees and designing differently shaped sampling regions, there are a number of other strategies for improvement. Wang et al. [52] devise a function to compute the variation parameter whose exponential term introduces the number of collision detection failures. In Ref. [76], the conditional sampling method is proposed, where nodes that cannot be part of the solution path due to velocity constraints are prohibited from sampling. In Ref. [125], dynamic RRT is proposed which, by heuristically using this path length as the major axis diameter of the informed subset, balances convergence time and path length in an environment with randomly distributed obstacles. In Ref. [126], sequential convex feasible set is introduced to RRT*, and bad local optima are avoided to get stuck in. In Ref. [72], double sampling points comparison and selection strategies are used to reduce the randomness of sampling points [127]. introduces a virtual field sampling algorithm and a current constraint function for multi-unmanned surface vehicle path planning under spatially varying currents.

In addition, requirements at the application level also contribute to the improvement of RRT sampling strategies, to address the problem of relatively low efficiency due to the increase in degrees of freedom in the robot system equipped with gantry structures, Wang et al. [128] introduces the sampling pool mechanism, and selects the node nearest to the connection line between the starting node and the target node in the sampling pool, which effectively shortens the length of the search path. Experiments show that compared with IB-RRT* [129], the path cost and time cost are increased by 22.2 % and 32.5 %, respectively, and the success rate is relatively stable. To address the issue of low search efficiency in large-scale road network environments, Adjacent-relation based RRT repetitive sequence optimization (A-RRT-RSO) is proposed, in which the A-RRT adopts a greedy strategy to sample neighboring mesh nodes, increase the leaf nodes and generate the RRT + extended tree. The proposed A-RRT-RSO reduces the number of search nodes, avoids blind search, and minimizes the cost of path computation [130]. To address obstacle avoidance path planning for manipulators, Zhanga et al. [131] develop a sampling motion planning algorithm. The algorithm samples configurations in advance that satisfy specific constraints associated with a prescribed motion planning task and stores them in an offline configuration dataset. This approach is based on the premise that the manifolds are continuous over a specific range. Zeng et al. [132] propose a tournament-selected point sampling strategy based on RRT to guide the underwater vehicle to the area of interest for sampling. The strategy maximizes information collection while working within a limited budget. Other applications include circular sampling strategy for autonomous vehicles [133], path planning for a UAV in communication-constrained operating environments [134], flight cost-based RRT for energy-efficient industrial robot motion planning [24], vector field stream based RRT* to plan unmanned surface vehicle paths under spatially variable ocean currents [135,136].

3.3. Post-processing

The reason why RRT algorithms need post-processing can be stated at the theoretical level and the application level. From the theoretical level, the feasible solutions can be further optimized in terms of evaluation metrics such as path length, complexity, and consumption time, etc. Some of the previous reviews also falls into post-processing, like Some kinds of RRT-Connect [35,36,45,46], informed-RRT* [23,92,96], Quick-RRT* [26,110,112,113], RRT*-Smart [73,79,80], Fast-RRT* [116,119], F-RRT* [7,118], post triangular rewiring strategy [52,137], interpolation post-processing [138,139]. From the application level, the feasible solutions are likely to not satisfy the kinematic model constraints of the robot or the scenario constraints in some application scenarios, therefore, the theoretically generated feasible solutions cannot be applied to real scenarios and further optimization is still required. In terms of considering kinematic model constraints, Zhou et al. [140] design a redundant dual-chain manipulator with two kinematic chains and a fixed base. They generate joint trajectories corresponding to the paths using flow shape analysis. Experiments confirm the method's practicality. Berg et al. [141] design curvature-aware with closed-loop RRT (CA-CL-RRT) to enhance the path planner's performance on curved highways, where in the closed-loop prediction phase, the virtual car follows the generated reference path and speed profile. In the curvature phase, the upper bound of the curvature constraint is introduced. Experiments demonstrate that the CA-CL-RRT algorithm proposed can significantly improve the path quality, particularly on curved roads with a radius of less than 1000 m. In the current automatic parking system, the parking trajectory planning algorithms based on geometric connection or optimization problem descriptions have problems such as strict requirements on the starting position, low planning efficiency, and discontinuous reference trajectory curvature. To solve these problems, Wang et al. [142] propose a hierarchical planning algorithm combining nonlinear optimization and an improved RRT* algorithm with Reeds-Shepp curves. Simulation results show that the proposed algorithm can design an effective parking trajectory under multiple parking scenarios. The Stanley algorithm is also used for path tracking and Reeds-Shepp curve to adjust the final parking attitude of the truck [143]. Mao et al. [144] propose the retains the discrete search of the original rules of RRT while adding the continuity of the motion of unmanned surface vehicle (USV), where each movement including position, yaw angle, velocity, etc. is takes into account the complete dynamic constraints, which is called the state prediction RRT (SP-RRT) algorithm. In Ref. [145], the improved two-step timed elastic band is introduced to smooth the path and optimize path lengths of automatic guided vehicles. To address the significant challenges faced by RRT-related algorithms in ship path planning, such as slow convergence and excessive turning points, Gu et al. [146] first cluster the priori data to construct a bootstrap region that guides the RRT branching. Then, they optimize the paths using the Douglas-Peucker compression technique. The results show that the proposed algorithm achieves a good balance between efficiency and accuracy. To solve the online cooperative path planning problem for multiple quadrotors in unknown dynamic environments, Jia et al. [147] establish the kinematic constraints for quadrotors and propose a spatio-temporal coordination strategy applicable to RRT. In addition, commonly used curve smoothing methods are also introduced into the RRT algorithm, including the Bezier curve-based RRT [83,107,148], B spline curve-based RRT [37,116,149].

In terms of considering scenario constraints, Kim et al. [150] introduce TargetTree-RRT* for complex environments like narrow parking spots, where in this algorithm, clothoid paths are introduced for post-processing to address curvature discontinuity, a cost function is used to create an objective tree that considers obstacles. For the problem of path planning in dynamic environments, Guo et al. [151] propose a hierarchical structure that updates the surrounding information in real-time at the perception layer. This structure aims to obtain heuristic paths at the path planning layer and improve path quality at the path optimization layer by combining a sampling method with an artificial potential field function. To address the long-distance and multi-level planning tasks of self-driving vehicles, Zhao et al. [152] propose a lifelong learning framework with GAN and RRT, taking the tractor-trailer as an application case and testing the proposed method in several scenarios with different characteristics. To address the path planning problem in visual servoing, Reyes et al. [153] incorporate visual servo control into state information to create local trajectories for RRTs. They demonstrate that this approach is probabilistically complete. To address the problem of RRT* failure in certain constrained environments, Ramasamy et al. [154] design an adaptive RRT* (ARRT*) algorithm in a created digital twin simulation environment, which designs a collision detection function to achieve dynamic sampling. The results show that ARRT* performs better than RRT* in constrained environments. In response to stochastic disturbances in the scene, Pedram et al. [155] propose a novel path length metric containing a weighted sum of the robot motion cost and the robot perception cost in an uncertain configuration space, and combine this strategy with the existing RRT* algorithm. In Ref. [156], the improved RRT algorithm takes into account the direction of movement of the obstacles, iteratively generates paths within a specified time and selects the shortest flight path. To overcome the situation when scanning complex objects with many obstacles, Yan et al. [65] propose the direction-guided RRT, an algorithm that builds on RRT by first simplifying the invalid paths through linear processing and then, smoothing the paths.

Some scholars notice that RRT-based improvement are generally applicable to static scenarios, but there is a need for dynamic obstacle avoidance in real scenarios, so combining other dynamic obstacle avoidance strategies to achieve RRT dynamic obstacle avoidance path planning is also a hot research topic. Dynamic window enables dynamic obstacle avoidance tasks, but there is a risk of falling into local optimal solutions, therefore [157], proposes RRT*-fuzzy dynamic window approach (RRT*-FDWA) for collision-free path planning, where a reward and penalty function is designed so that the robot can quickly enter global path planning once it has successfully avoided an obstacle. D*, or Dynamic A*, is also a general class of techniques for dealing with dynamic obstacles, based on the principle of updating the overheads between states in the map in real time when a moving obstacle is detected during robot motion, and in Ref. [80], dynamic A* evaluation function is designed and introduced to RRT*-smart. The finite acceptance of bad solutions is the core of the Metropolis acceptance criterion. It is frequently used in simulated annealing algorithms to calculate the acceptance probability of a solution. In Ref. [158], based on the Metropolis acceptance criterion, an asymptotic vertex acceptance criterion and a nonlinear dynamic vertex acceptance criterion are developed. Q-Learning is an algorithm in reinforcement learning for making decisions and learning based on behavioural norms and rewards, in Ref. [63], a decoupled real-time motion planning framework is proposed that combines robust intermittent Q-learning with a sampling-based motion planner in which the sampling module begins each iteration by updating the neighbourhood radius. Some advanced control theories, such as linear quadratic regulator (LQR) and nonlinear model predictive control (NMPC), can withstand dynamic disturbances. The literature [159] proposes a two-stage risk aversion architecture designed for the safe control of stochastic nonlinear robotic systems. This architecture combines a novel RRT* variant for nonlinear steering, distributed robust collision checking, and a low-level reference tracking controller. Numerical experiments on unicycle dynamics demonstrate that NMPC outperforms LQR and its LQR variants in terms of performance metrics. In Ref. [160], the nominal mean value of the stochastic control distribution in the model predictive path integral is provided by RRT, leading to satisfactory control performance in both static and dynamic environments without any parameter fine-tuning. Other applications include manipulator dynamic obstacle avoidance [161], hybrid assembly path planning for complex products [162], 10-DOF rover traversing over 3D uneven terrains [163], UAV path planning [77,151,164], electric inspection robot navigation [165], cobot in dynamic environment [166], underground vehicles [167], automated guided vehicle [145], mining truck [143], redundant robots [168].

3.4. Model-driven RRT

3.4.1. RRT with support vector machine classifier

In 2019, I and our team plan paths fuse RRT, Support vector machine (SVM) classifier and longest accessible path with course correction (LAP-CC) to address the articulated vehicle path planning problem under globally known maps [169]. We first use the RRT algorithm to generate a feasible path from the starting point to the end point, and then label the obstacles on both sides of the path as positive and negative classes, and learn to obtain the zero-potential decision curve through the SVM algorithm, and then for the problem that the curvature of the zero-potential decision curve does not conform to the kinematic model of an articulated vehicle, we propose the LAP-CC algorithm for post-processing of the path.

3.4.2. RRT with random trees classifier

Dominik et al. [170] investigate the classification results of Random Trees (RT) classifier and SVM classifier in a real outdoor scenes containing grass, leaves, pavement, trees, asphalt, wall, bushes and concrete as a way to construct the obstacle map, and the authors conclude that RT classifiers are more suitable for field scenes compared to SVMs, and then the authors use the RRT-Connect algorithm for six-legged walking robot path planning in scenarios modelled after RT-based classifiers.

3.4.3. RRT with k-nearest neighbor queries

Pan et al. [171] present a new method for fast probabilistic collision checking to accelerate the performance of RRT-based motion planning, where k-nearest neighbor (k-NN) is used to find the nearest prior query sample to the new query configuration. The results show that this fusion method improves the RRT-based path planner in terms of accelerating local path and improving search order on the roadmap, and the findings are also validated on rigid and articulated robots. Similar work is done in Ref. [172]. The authors also state the shortcomings of the fusion algorithm and discuss future directions: first, the algorithm is parameter-sensitive, which means that the hyper-parameters of k-NN need to be adaptively tuned for datasets of different sizes; second, the collision search strategy should not be balanced, and well-explored regions should be subjected to as few collision detection queries as possible; furthermore, it is also meaningful and necessary to improve the method to adapt it to collision exploration in dynamic environments.

3.4.4. RRT with logistic regression

To address the problem of obstacle avoidance path planning for irregularly shaped obstacles, Peng et al. [173] propose a control barrier function-based RRT* to generate a collision-free path for a bipedal robot with multiple polynomial-shaped obstacles, where the polynomial-shaped obstacles are estimated by logistic regression. The authors summarize the shortcomings of this method, including an obstacle occupancy that is too small to generate a suitable control barrier function, and the inability to express the surrounding obstacles with a single control barrier function.

3.4.5. RRT with random forest

Cano et al. [174] evaluate and compare four different hyper-parameter tuning methods (random sampling, AUC-bandit, random forest, and bayesian optimization) for RRT-connect, and random forest (RF) is more effective in hyper-parameter optimization as compared to random sampling and bayesian optimization, and its path generation under RRT-connect is 1.2 times more efficient than the default hyper-parameters.

Song et al. [19] use five machine learning algorithms to learn static obstacle data from the scene in the expectation of improving obstacle avoidance efficiency. Experiments show that RF provided the best predictions compared to k-NN, multinomial naive bayesian, gaussian naive bayesian and SVM.

3.4.6. RRT with neural networks

Baldoni et al. [175] utilize neural networks to guide the generation of path planning with the expectation of improving the generation of datasets and increasing the efficiency of path planning. The authors conclude that the path planning task for narrow passages and maze-like maps is very similar to the image segmentation task, so they choose U-net, which is widely used in the field of image segmentation, to train the path planning task, and the results show that the neural network-guided RRT significantly outperforms the traditional RRT in terms of path planning.

In many cases, robots do not want their paths to be predictable, and to satisfy this need, Nichols et al. [176] exploit the feature of adversarial neural networks to improve network generalization and propose adversarial RRT*, which is the addition of a deception cost term to RRT* through the use of recurrent neural networks (RNN). The results show that the paths generated by RRT* containing RNN reduce the observer accuracy (47 % for RRT* and 19 % for adversarial RRT*), increase the path length by 29 % and the entropy by 22 % compared to the optimal paths, suggesting that the paths generated by adversarial RRT* are more difficult to predict.

Considering that RRT* fails to generate optimal paths due to the inability to know the features of the environment in advance, Ma et al. [177] propose a supervised neural network based method to learn scene features. In terms of network design, considering that conditional generative adversarial networks (CGANs) have the property of improving the performance of both adversarial and collaborative work, they have more potential for algorithmic performance metrics improvement than traditional supervised neural networks. The authors fuse CGAN into RRT*, and the results show that CGAN-RRT* outperforms Neural RRT* [178] on the same training set.

In highly dynamic environments where efficient embedded implementation of algorithms is critical due to limited resources of the on-board microcontroller, Chaulwar et al. [179] propose a hybrid augmented CL-RRT including strategies such as introducing iterative ConvNet feature generation, input iterative convolution, compression of fully-connected layers in sparse columns, and storing only the states with the lowest severity of injuries for rapid generation of safety trajectories in critical traffic scenarios and deploy it on a TMS570LS20216 microcontroller.

3.4.7. RRT with reinforcement learning

To address the non-homotopic path problem that leads to significant differences between the demonstration path and the generated path (the generated path may be shorter, but do not conform to human social relations), Ding et al. [180] suggest combining a non-isotropic path penalty strategy with RRT inverse reinforcement learning, where the non-isotropic features are extracted as a penalization term in the objective evaluation metrics, and the subjects are allowed to perform a Turing test as a real moving obstacle on the generated paths in the subjective evaluation metrics.

For multi-robot scheduling problems, such as the multi-ship aircraft unmanned path optimization problem for ship deck scheduling efficiency, Shang et al. [181] point out that reinforcement learning is used to generate paths for multi-ship aircraft, and the results show that the reinforcement learning based path planning algorithms are superior in terms of response time, scheduling completion rate and average path length compared to traditional path planning algorithms.

For the path planning problem in hostile environments, the classical path planning method fails when the risk characteristics cannot be accurately labeled manually, to address this problem, Guo et al. [182] reduce the risk of generating paths based on the data-driven approach that constraints the growth direction of RRT*. The authors also mention that multi-UAV risk aversion and the need for further balance between unbiased and biased sampling also deserve attention.

In safety-related domains such as driverless driving, it is extremely difficult to design a cost function and assign appropriate weights taking into account self-interests, ethics, laws, and limitations faced by the field of view, and therefore the use of human driving experience data and reinforcement learning algorithms for generating local goals and semantic speeds is also an important research topic. Yu et al. [183] develop a framework for self-driving cars by combining deep learning with RRT. Specifically, the authors extend deep neural networks (DNNs) to environments with multiple traffic participants and accelerate the training process using double-deep Q-networks (DDQNs) and prioritized experience replay (PER). Reinforcement learning is also needed to guide the generation of path planning for similar problems containing good candidate states identification and the accurate steering angle calculation [184], bevel-tipped needle paths design for surgical robots [185], assembly tasks with human-robot collaboration [186].

3.5. Advantages of the RRT-based improvement strategies

3.5.1. Branching strategy improvement for large-scale scenario search

The reason why traditional RRT is difficult to be applied in large-scale scenarios is that RRT tends to get stuck under certain obstacles, such as maze-like obstacles and narrow passages, resulting in the failure to obtain a feasible solution, as a contrast, branching strategy improvement plays an important role in the application of RRT in large-scale scenarios. For example, RRT-Rope is designed for UAV exploring large-scale environments such as underground mining stopes. The algorithm is validated in real time on an HP Z440 workstation equipped with 12 Intel Xeon processors for a long tunnel over 20 m, long and a sloped planar space over 30 m, and an indoor environment containing columns and doors over 40 m, with path generation taking 0.25 s, 0.55 s, and 0.45 s, respectively [46]. Studies that are relevant to branching strategy improvement and also claim to be large-scale scenarios include 131.4 s to explore an 8 m*8 m map [187], 56.75 s to explore 143.13 square meters of 182-square-meter indoor maze map with 81.46 % precision and 92.89 % accuracy [188], 1758.8 s to explore 30 m*30 m map [189], 555.6 s to explore 1000 m*1000 m with 3D terrain map [64], 10 s to explore 467*785 pixel maps of complex ocean environment with multiple vortices [136], 12 machine cycles for power inspection [165], 53.1 s to explore 20*20 grid maps [190], 6.654 s, 8.8845 s, 6.654 s, 16.1148 s, 7.7544 s, 6.0529 s to explore the dataset of Chem97ZtZ, gemat12, bcsstk33, kron_g500-logn162, CoAuthorsCiteseer [191]. The reason for the large gap in time cost is that some results only consider the time required for path generation, while others consider the time required for the robot to explore the map, and the above results demonstrate that branching strategy improvements can handle the challenges of large-scale environments.

3.5.2. Sampling strategy improvement for real-time performance

The improvement to branching strategy, sampling strategy, and post-processing increase the complexity of the algorithm, but due to the increased computational power of the hardware, these improved strategies still result in a fast solution while substantially increasing the quality of the paths. The real-time performance of the improved RRT in a static scenario is shown in 3.4.1, and even for dynamic scenarios, the improved algorithm still shows its superior real-time performance. For example, STL-RT-RRT* [192] and RT-RRT* [193] can react to moving obstacles in real scenarios (1.1 m/s for moving obstacles and 0.55 m/s for the robot). STL-RT-RRT* even achieves 100 % obstacle avoidance success rate in 1000 trials, with a failure chance of less than 0.1 %. In Ref. [151], the time-based sampling process performs well in radar and missile tracking avoidance (92 % success rate). In Ref. [192], a car-like robot on a miniature circular runway at a speed of 0.17 m/s successfully avoid the moving obstacles across the runway. In simple dynamic scenarios, the average dynamic obstacle avoidance success rate of the robotic arm is 100 % and the average path generation time is less than 0.1s [161]. Minkyu et al. [156] conduct UAV path planning experiments in which UAVs can generate an average of 1262 potential paths in a single obstacle environment and 770 potential paths in a multi-obstacle environment within 0.1s. In Ref. [101], when the distance between the quadrotor and the mobile platform is 15 m, the average computation times for path search, corridor generation, and trajectory generation are 3.7 ms, 4.9 ms, and 20.1 ms, respectively, even if the environment is filled with various types of dynamic obstacles. In Ref. [194], the robot with a speed of 4 m/s explore a dynamic scene of 97 m*142 m in less than 70s, and the path generation time is almost negligible compared to the path tracking time.

3.5.3. Post-processing in uncertain environments

One type of uncertain environment is the dynamics of the scene, Alexis et al. [192] introduce a signal temporal logic (STL) for real-time dynamic obstacle avoidance, which has the potential for dynamic obstacle avoidance optimization as it expresses through quantitative semantics that a robot should always keep a safe distance from a human or move slowly in narrow passages. The authors compare the performance of STL-RT-RRT* and RT-RRT* [193] and show the better performance of STL-RT-RRT* in terms of success rate in dynamic obstacle avoidance (no collisions for STL-RT-RRT* and 513 collisions for RT-RRT* in 1000 trials), the number of stops affected by dynamic obstacle avoidance (4 times for STL-RT-RRT* and 286 times for RT-RRT* in 1000 trials), and safe distance from dynamic obstacles (e.g., 1.2 m radius from human). Aiming at the targeted dynamic threat (radar or missile tracking) and random dynamic threat (tracking moment and tracking speed) that UAV may face, Guo et al. [151] propose a time-based sampling process for continuous change process of dynamic obstacles, fuse the APF structure for potential collision process, and introduce the cost function consisting of the true distance cost and the estimated distance cost to construct the heuristic path-finding process, and the comparative experiments verify the advantages of this algorithm in terms of the navigation time, path length, and the success rate of generated paths. Yu et al. [195] construct a miniature circular runway as an autopilot environment for a car-like robot and set moving obstacles across the runway as dynamic obstacles, and the algorithm can plan the trajectory points over time in advance according to the trajectories of the moving obstacles and smooth the paths under the premise of satisfying the curvature constraints of the car-like robot, tracking the trajectories at a speed of 0.17 m/s and avoiding the obstacles successfully. For the robotic arm path planning problem in a dynamic scene, Yuan et al. [161] compare D-RRT and DBG-RRT, and both algorithms have good real-time performance and high success rate of obstacle avoidance under three different maps. Under map 1, the average time and the average success rate for generating paths are 0.064s and 97 % for D-RRT and 0.017s and 100 % for DBG-RRT. Under map 2, the average time and the average success rate for generating paths are 0.083s and 98 % for D-RRT and 0.022s and 100 % for DBG-RRT. Under map 3, the average time and the average success rate for generating paths are 0.204s and 91 % for D-RRT and 0.021s and 100 % for DBG-RRT. Shubhi et al. [163] investigate the rapidity of path generation and the stability of path tracking for rover in 3D terrain, and several experiments show that the rewiring process for dynamic obstacle avoidance only takes less than 3 ms. Another interesting example is the interaction with humans. Considering the two types of human behaviors (conservative walkers and aggressive walkers) when encountering obstacles, the authors [196] design two types of moving obstacles to simulate conservative and aggressive behaviors; when the moving obstacle is perceived as an conservative walker, the robot replans its trajectory to go around the obstacle from in front of the obstacle, and when the moving obstacle is perceived as an aggressive walker, the robot replans its trajectory that bypasses the obstacle from behind the obstacle.

Another type of uncertain environment is unknown environment, and this part of the task mainly involves exploration tasks, as detailed in section 4.2.5. The above literature demonstrates that the improved RRT has been widely used in two types of uncertain environments with good results.

3.5.4. Model-driven RRT for path quality improvement

Model-driven RRT is broadly classified into two categories, one in which there is already prior data (supervised learning), and the other in which lessons are learned over multiple explorations (reinforcement learning), both of which can improve the quality of path generation, as summarized below: (1) Considering irregular obstacles in real scenarios, some supervised learning strategies model obstacles in the environment to improve the accuracy of paths and enhance the applicability of the algorithm in complex scenarios [19,169,170,173]. (2) Supervised learning algorithms improve the search order of paths, thus improving the quality of local paths [172]. (3) Under supervised learning algorithms, the collision detection queries involved in RRT should not be balanced, i.e., regions with sparse obstacles can have fewer collision detection queries, which can further improve the path generation efficiency [171]. (4) The selection of hyper-parameters is difficult for any algorithm, whereas supervised algorithms can use historical data to select more appropriate hyper-parameters to optimize path quality [174]. (5) For the similar specific problems in the field of artificial intelligence and the field of path planning, the methods involved in the field of artificial intelligence may also be directly used in the field of path planning to achieve good results, for example in Ref. [175], the authors fuse U-net into RRT due to the similarity between path planning for maze-like maps and high-precision image segmentation. (6) An interesting work is to generate unpredictable paths to ensure privacy, a time when supervised adversarial neural networks show their strengths [176]. (7) Supervised networks that enable more accurate environment perception have the ability to enable RRT* with asymptotic optimality to achieve optimality [177,178]. (8) Some optimal paths that are consistent with human cognition are difficult to design suitable cost functions, in which case it is necessary to use reinforcement learning and social rules for path optimization, such as calculating non-homotopic features (area between the demonstration paths and the generated paths), as well as a questionnaire survey of the subjects (Turing test) to evaluate the degree of socialization of the generated paths [180]. (9) Reinforcement learning primarily addresses problems caused by the inability to accurately model or evaluate, including the inability to estimate risk characteristics [182], good candidate states identification and the accurate steering angle calculation [184] and the various factors caused by human-robot interaction [183,185,186].

4. Recent advances in the application of RRT to robotics

According to the different application scenarios of robots, the International Federation of Robotics (IFR) classifies robots into industrial robots and service robots, and this part summarizes the latest progress of RRT on various types of robots.

4.1. Industrial robots

Industrial robots are robots that can be automatically controlled, reprogrammable, multifunctional, and multi-degree-of-freedom robots, which include handling operation/loading and unloading robots, welding robots, spraying robots, machining robots, and assembly robots, etc., and this section reviews recent advances in RRT for these robots.

4.1.1. Welding robots

Wang et al. [52] design the adaptive extended bidirectional RRT* algorithm applied to the path planning problem in complex environments with concave and convex surfaces, narrow passages and multiple obstacles. The authors not only prove the probabilistic completeness and asymptotic optimality of the proposed algorithm, but also conclude that the time complexity and space complexity are the same as those of the RRT and RRT* algorithms. Although the authors do a lot of simulation experiments and verify that the performance of the proposed algorithm is much better than many improved RRT* algorithms, the field test consumes a lot of time because of the need for collision detection of the six arms of the robotic arm and the welding gun. The results show that with a search step size of 10 mm, when the path length are 831.5 mm, 353.7 mm, 554.0 mm and 543.7 mm, the running time is 1109s, 307s, 210.8s and 447.5s. Similar tasks, strategies and conclusions are presented in Ref. [128], where the improved RRT* is used for gantry welding robot system.

Considering three conflicting objectives of arc welding robots: minimum transfer path length, energy consumption and joint smoothness, Zhou et al. [197] calculate the path length using Euclidean distance and the energy consumption and joint smoothness expressions using the kinematics of the robotic arm, and assign 40 %, 30 %, and 30 % weights to the three objectives, respectively. In order to accomplish the multi-objective search task, the authors also propose a decomposition-based multi-objective evolutionary algorithm with a hybrid environment selection algorithm, and set the value ranges of the optimization parameters according to the application scenarios. The authors also mention that future work needs to focus on real-time path planning tasks for welding robots. Similar tasks, strategies and conclusions are presented in Ref. [198], where the goal is to minimize path length and energy consumption.

4.1.2. Assembly robots

Shu et al. [199] present an improved RRT* for the assembly of lightweight structures for COVID-19 healthcare facilities, which excels in collision avoidance, trajectory smoothness, trajectory length and execution time, and the authors note that the total assembly time for the assembly robots designed in this paper is approximately 28 min, whereas professionals recommend a manual assembly time of no less than 45 min, making it competitive. However, the authors also note that the assembly process is not yet fully automated due to the lack of flat-packed housing components, and suggest that the use of aerial operation robots to assist in tightening the bolted joints can further reduce installation time and fully automate the assembly.

Chen et al. [30] propose a dual-robot collaborative assembly path planning algorithm based on RRT-Connect and design a collaborative wire removal task for the dual robots. The results show that the trajectories generated in field tests are consistent with the simulation, and the task of removing the wires is successfully executed, and the results indicate that the designed trajectory planning method and the dual-robot collaborative system are effective.

For the problem of assembling parts with different geometries, Ahmad et al. [200] introduce constraints on the initial and final pose grasping of the parts, and successfully accomplish the task of assembling the parts in a disturbance-free environment; for the dynamic environment, the authors propose that in the future, they will intend to install proximity sensors on the robotic arm to detect the surrounding obstacles in real time and to generate dynamic obstacle avoidance paths.

4.2. Service robots

Service robots are designed to perform service tasks that benefit humans, including commercial and domestic robots in the fields of accommodation, catering, finance, cleaning, logistics, education, culture and entertainment, as well as robots that include assisting or replacing a human in performing tasks, such as search and rescue robots, surgical robots, underwater robots, free-floating space robots, etc., and this section reviews recent advances in RRT for these robots.

4.2.1. Search and rescue robots

For exploration missions, commonly used methods include frontier-based methods and sampling-based methods. Frontier-based methods are similar to search-based strategies in the path planning domain, but two different definitions are used in the field of path planning, but two different definitions are used in the field of path planning and in the field of simultaneous localization and mapping (SLAM), since exploration missions also involve mapping the unknown regions. Sampling-based methods in the field of SLAM are also similar to sampling-based strategies in the field of path planning, and RRT-based algorithms are also a representative algorithm for this type of methods for exploration task. The advantages and disadvantages of the two types of algorithms are shown in Table 2.

The two representative algorithms have their own strengths, therefore, the fusion of the two algorithms is a research direction in the field of task exploration. To address the problem of switching exploration targets back and forth, which exists in both methods, Bi et al. [201] design a novel utility function to evaluate candidate targets and a target reselection mechanism to assign exploration targets. Simulation and field tests show that compared with traditional RRT-based multi-robot exploration, the proposed framework has less time cost and path cost with 99 % exploration coverage rate. The authors also draw some conclusions and discuss them: (1) When the number of robots is increased from 2 to 4, the exploration efficiency increases almost linearly with the number of robots, while when the number of robots is increased from 4 to 7, the increase in the exploration efficiency is not significant, so the optimal value of the number of robots in the same scene can be determined through multiple experiments. The reason that increasing the number of robots does not increase the exploration efficiency is that the starting position of each robot, the complexity of the scene, the size of the scene, and the detection distance of the laser scanner all affect the exploration efficiency. (2) When the detection distance of the LiDAR is increased from 4 m to 16 m, the exploration efficiency is increased, while when the detection distance of the LiDAR is increased from 16 m to 50 m, the exploration efficiency is slightly decreased, so the detection distance should also be set appropriately according to the size of the environment. In addition, too large LiDAR range also increases the computational cost. (3) Compared with the frontier-based method, the proposed centroids of unknown connected regions is superior in terms of computational cost, detection robustness and decision-making performance in unknown environments. (4) This paper adopts a centralized architecture, and when the number of robots is large, the amount of data in the central node increases dramatically. (5) Data transmission can be optimized in the future to reduce the requirement of communication bandwidth.

For the underwater search and rescue missions, Wang et al. [104] propose the smooth-RRT algorithm and use the 3D point cloud of the underwater scene captured by the sonar as the environment data to simulate and verify the algorithm's contribution to the initial solution quality and convergence speed. Similar work is in Ref. [202], which introduces the KD-tree based RRT-connect algorithm and verifies that the algorithm can be used for fire guidance on a rescue robot equipped with sensors including LiDAR, IMU, and camera. The above literature only tests the performance of one robot in a single scenario containing only static obstacles and only verifies the feasibility of the algorithms, thus failing to evaluate the level of intelligence of rescue robots in real scenarios.

For the ground search and rescue missions, Noé et al. [203] investigate a ground robot navigation and exploration system in complex indoor 3D environments such as mines, solving or optimizing the problems of scenario creation, path planning, path tracking,

Table 2
Performance comparison of classical algorithms.

Methods	Advantages	Disadvantages
Frontier-based methods	Robustness	Inefficient in large maps; Trajectories overlap; Moving back and forth between two targets
Sampling-based methods (e.g. RRT)	Efficiency	Weak robustness; Trajectories overlap; Moving back and forth between two targets; Trapped in narrow regions; Missing small regions;

and region importance assignment, and summarize the following conclusions: (1) After filtering local point cloud with a fixed size and without walls and ceiling, the improved RRT algorithm can be prevented from sampling and evaluating in invalid regions, and therefore, the efficiency of path generation is significantly improved. (2) To balance the quality and efficiency of the generated paths, as well as to avoid inaccurate scene descriptions, this paper investigates the appropriate point cloud downsampling resolution, which is taken to be 0.05 m. (3) RRT rather than grid or voxel grid is used to find the frontiers with the strategy of clustering the leaf nodes, but the clustering radius and the clustering density still need to be selected manually. (4) The coefficients of the cost function of the proposed improved RRT algorithm need to be manually adjusted to find the optimal path, taking into account different terrains such as inclination, roughness, etc. (5) During path tracking, the maximum linear and angular velocities of the vehicle need to be manually adjusted for optimal performance. With the manually adjusted parameters, a 4-m wide tunnel is explored in 20 min and a 17-m by 17-m rugged terrain with different ramps and obstacles is explored in 25 min, both successfully covering more than 90 % of the area. (6) In terms of the rescue mission, the frontier threshold determines the level of exploration; excessively low frontier thresholds result in many areas being potentially unexplored, and too high frontier thresholds result in many areas being thoroughly explored, which can lead to some relatively unimportant regions being over-explored, resulting in a significant increase in rescue time, so an adaptive frontier threshold selection strategy is also particularly important. (7) To avoid repetitive exploration of the same area, the evaluation indicator of visit area assessment is also introduced, but no solution is given.

Gui et al. [204] propose a decentralized multi-UAV cooperative exploration method that considers both the position and current task of each UAV to ensure that the task can be reassigned in real time during each UAV's exploration. Each UAV is equipped with a depth camera to achieve localized scene sensing in a dynamically partitioned area, and the improved RRT is used to explore the unknown environment. The paper draws the following conclusions: (1) The average time for the three UAVs to explore an area of 10 m*8 m*3 m with obstacles is 209. 4s, and there is no collisions or prolonged inability to complete the task, therefore, although the authors are not concerned with hardware limitations, network bandwidth, and flight trajectory control, the results are stable and effective; (2) Although multiple UAVs can accomplish the missions in a limited time, there are some undetected areas due to the inherent limitations of the depth camera, and the authors mention the need to use more powerful sensors (e.g., 3D LiDAR) to alleviate this problem, but this approach still requires further consideration of the applicability of the experimental system and its applicability to different environments; (3) In the early stage of the exploration, UAVs with sampling-based strategies can find a sufficient number of mission points in a short time, but as the unexplored area gradually shrinks, the time to compute the target becomes longer and even converges to a lower level. (4) Uncertainty in sampling leads to irrational partitioning, preventing different trials with the same parameter settings from providing a stable exploration process.

Communication constraints is an important problem faced by robots. To address the problem of signal interference of disaster relief UAVs caused by low-altitude clouds and smoke, Diao et al. [70] regard clouds as dynamic obstacles and set the direction and speed of their movement. At the algorithmic level, the authors use the adaptive step and angular incremental sampling to limit the sampling range, which reduces the curvature of the generated paths and accelerates the convergence speed.

None of the above literature considers the impact of communication constraints on the rescue robots, or simply identifies communication constraints as obstacles [70], but since robots can only communicate with each other within line-of-sight, the safety hazards due to communication constraints in decentralized collaborative exploration tasks should be considered. To address this issue, Victoria et al. [205] propose decentralize path planning for multi-robot systems with line-of-sight constrained communication, and make the following assumptions: (1) Known static scenes; (2) Robots can brake instantaneously; (3) Robots can communicate with other robots through a multi-hop communication network; (4) The communication between robots is lossless and without delay; (5) The initial positions of all robots are sufficiently safe; and (6) Robots start from a set of waypoints that satisfy all safety constraints. Some of the assumptions mentioned above are extremely idealized, therefore, although the results validate the effectiveness of the RRT-based method, it is far from meeting the requirements of practical applications.

4.2.2. *Surgical robots*

Zhang et al. [206] design flexible needles that can flexibly avoid blood vessels and organs and reach diseased tissues. Considering that the non-holonomic characteristics of the needle tip can lead to dynamic and tissue deformation of the flexible needle during insertion, the accessibility and safety of the needle state need to be considered in the path planning stage. In this paper, the following conclusions are drawn: (1) The proposed algorithm can adapt some of the parameters according to the strategies of different tissues during the insertion of the needle tip, which improves the safety of surgical procedures. (2) The proposed algorithm takes into account the potential field of the surrounding obstacles in the path cost and simulates the effect of the needle tip insertion process on the local motion of the tissues, so as to address the effect of needle tip insertion on local tissue movement. (3) The improved RRT* algorithm generates smooth and safe trajectories in a layered fabric environment, verifying the first two conclusions. It can be seen that for surgical robot path planning, the modeling of tip motion constraints and human tissue properties determines the quality of the generated paths. Similar design and modeling constraints are in Informed-RRT* based cranial puncture robot [207] and RRT-Connect based surgical suture robot [208].

Chen et al. [209] propose the shape state cross-entropy based RRT* planner for surgical experiments, and for the continuum robot, this algorithm has the following advantages: (1) High-dimensional continuum robot configuration space is considered. Arc coordinate domain is designed to perform both obstacle-free and approximate follow-the-leader motions. (2) Kinematic constraints are considered. The generated paths introduce kinematic and shape constraints, including constraints on the maximum extension direction angle of the robot tip, to ensure traceability of the trajectories. (3) Stable numerical solutions are ensured. Stable numerical solutions are ensured by rewiring paths via damped least squares to minimize robot configuration changes and trajectory costs, and by solving the singularity problem in the pseudo-inverse kinematics solutions. The authors segment the anatomical environment from medical images of real patients and generate point cloud maps of this environment in MATLAB. Simulation results show that the proposed algorithm performs well in follow-the-leader error and success rate, as long as collisions with intracranial blood vessels (arteries and veins) can be avoided.

Song et al. [210] also design a master-slave control framework applied to minimally invasive surgery, in which the motion of the master manipulator with respect to the monitor corresponds to the motion of the surgical instrument with respect to the endoscope, and realize the removal of a porcine gallbladder.

Fan et al. [211] design a micro-robot for working in vascular environments. Although the simulation results demonstrate the safety and feasibility of the experiments, as well as the ability to automatically avoid static and dynamic obstacles in the simulated vascular environments, the human vascular and tissue environments are extremely complex and almost difficult to model, and thus the implementation of micro-robots working in vascular environments is still a great challenge. Similar work can be found in Ref. [212].

Surgical robots have very different path planning strategies due to their specific application scenarios.

4.2.3. Free-floating space robots

Free-floating space robots (FFSR) consisting of spacecraft and robotic arms are commonly used for on-orbit servicing, and motion planning is also the most essential task. Tomasz et al. [213,214] investigate the trajectory tracking performance of a space station robotic arm and find that the actual values differ from the expected values, although the deviation does not affect the utility of path planning, the authors still analyze the phenomenon: (1) The planar air-float microgravity simulator is subjected to various perturbations. The unevenness of the granite table may cause the spacecraft to slide to one side of the table, the robotic arm is controlled in the joint space, and the control system is unable to compensate for the positional and directional errors of the gripper. (2) There are significant uncertainties in the known parameters of the spacecraft and the robotic arm. Attitude deviations are most likely due to non-perfect knowledge of spacecraft inertia. In response to the latter question above, the authors also perform Monte Carlo simulations, which shows that the results of the Monte Carlo simulations are still useable even with a large range of uncertainty ($\pm$10 %), which further suggests that the errors caused by the latter are negligible. The authors also discuss the problems associated with the application of RRT to FFSR, as summarized below: (1) Spline-based planner and B-RRT* are compared, and the paths generated by RRT have more complex shapes, larger errors, discontinuities and unstable results, but in complex scenarios filled with obstacles, spline planner fails to generate feasible paths, so both algorithms have their advantages and disadvantages. (2) Only static obstacles and target objects are considered. However, earth-based measurements of potential targets for active debris removal missions show that some of these targets have non-zero angular velocities. Therefore, the end effector linear and angular velocities should be allowed to be adjusted during trajectory tracking. (3) The generated optimal paths do not take into account self-collisions between the robotic arm links. For the specific kinematic structures, such as the WMS lemur robotic arm, there is no risk of collision between the links of the robotic arm, but this risk should still be avoided for some robotic arms, as it could lead to a further increase in path costs. (4) The experiments are carried out in open-loop control mode, so it is impossible to correct the paths, especially when the navigation state is uncertain. For example, unexpected changes in attitude during a grappling maneuver may disrupt the communication link or interrupt power generation. (5) The computational cost is not taken into account, which makes it difficult to apply in practice. To address (3) mentioned above, Yu et al. [215] separate inertial spaces when performing the spline-RRT* algorithm and apply it to a dual-arm FFSR.

Some studies do not analyze in depth the drawbacks of these algorithms in applications, but nevertheless validate the applicability of various improved versions of RRT in FFSR, such as RRT for 12+2n dimensional satellite-manipulator system [216], B-RRT for a three-link FFSR [217], RRT without inverse kinematics for FFSR [218], B-RRT for 7-DOF FFSR [219], Improved adaptive RRT for close-range FFSR with continuous thrust [220], RRT-Connect for dual-arm FFSR [221] et al. These studies corroborate that RRT-based planner has potential for further development in the field of FFSR.

4.2.4. Mining robots

Compared with ordinary robots, mining robots, as a kind of underground robots, work in a non-structural terrain with limited space closure and GPS rejection. In addition, explosion-proof, dust-proof, moisture-proof, water-proof, corrosion-proof and other problems are challenging!

In [46], RRT-Rope benefits from RRT-Connect and is designed with the expectation of finding suboptimal solutions in a short period of time. Simulation results show that the method works well in large-scale uncluttered 3D environments such as underground mining stopes. The algorithm is also insensitive to the number of iterations and the environment due to the introduction of the shortened rope algorithm, resulting in further improvements in stability and efficiency, although the authors also point out that future work should introduce dynamic constraints to handle more complex environments. In addition, the authors specify that RRT-Rope falls into the local minimum for pillar-like structural obstacles, but since underground mining stopes scene do not involve such obstacles, the proposed RRT-Rope is also more applicable to underground mining scenarios.

Wang et al. [167] attempt to solve the path planning problem that satisfies the scenario constraints and the articulated vehicle model constraints in coal mine roadways, but the following problems still exist: (1) Despite utilizing parallel computing, the path generation time reaches 86.12s, which still fails to satisfy the application requirements; and (2) Joint debugging with the real vehicle has not yet been realized, and thus the gap between the performance of the simulation and that of the field experiments has not yet been understood.

Aiming at the problem that geological structure, ore characteristics, rock characteristics, and mine planning parameters all affect ore transportation, Shao et al. [222] first use Dubins paths and hierarchical density-based spatial clustering of noisy algorithm to build a constraint model containing the above conditions mentioned, and then combine the 3D RRT with Dubins paths for obstacle avoidance. The proposed framework has the following advantages: (1) Nodes of varying importance are identified based on ore characteristics and rock characteristics to provide guidance to the path planner. (2) Safer and less costly paths are selected for underground haulage based on geological structure. (3) User-intervention step is added to improve path connections in some cases after Dubins paths smoothing.

RRT for drill arm of anchor drilling robots [223], for mining truck [143].

4.2.5. Inspection robots

(1) Ground inspection robots

Five-axis coordinate measuring machine is widely used for machining part data acquisition, but how to rationally plan inspection paths of multiple features from different locations is an important issue in the automatic inspection of machining parts. Unlike other path planning tasks, the special application scenarios of machining and manufacturing require that certain critical features need to be measured multiple times to ensure the accuracy of the product, and furthermore, the critical features to be inspected are different in each process. To address problems mentioned above, one strategy is to reuse the initial paths, but this approach generates a large number of redundant paths; another strategy is to re-plan the paths, which reduces the redundant paths but increases the path search time. Zhao et al. [224] formulate a path reuse strategy that takes into account both the number of paths and the direction of the probing measurements, which is achieved by firstly, based on the feasible cone of probing directions and the accessibility of MPs classification of measurement features to minimize the probe rotation time, secondly, RRT with multi-root node is proposed to plan the local paths for path reuse, and then, the intra- and inter-group planning paths are generated based on the enhanced genetic algorithm, and finally, the effectiveness of the method is verified by the simulation of the cylinder body inspection case.

Aiming at the problem that long-term operation of gas-insulated switchgear leads to the accumulation of foreign matter in the cavity, which can lead to safety accidents, Zhong et al. [225] propose a beetle-antennae search-guided RRT* for the gas-insulated switchgear inspection and maintenance robot. Since the map for path planning is a cylindrical surface, the authors introduce the idea of simulating the foraging behavior of beetles, but still only at the level of algorithm validation.

Huang et al. [226] investigate a possible multi-objective path planning problem in inspection and solve the path planning problem with 25 destinations on an open street map of Chicago containing 866,089 nodes and 1,038,414 edges in only 0.44 s. However, realistic inspection scenarios may encounter some great challenges: firstly, the realistic inspection scenarios are a three-dimensional, which means that the obstacles cannot be treated as points as in a 2D plane, and second, the maps are not known a priori and the quality of the map generation is limited by the accuracy of the sensors, which further exacerbates the difficulty of path planning. Nevertheless, the conclusions of this paper provide good evidence that improved RRT is capable of achieving excellent performance at this stage.

With the rapid growth of the demand for clean energy, the number of solar power plants is gradually increasing, and the automation of solar power plant fault inspection is also booming, for the path planning problem of such narrow and long passages of solar power plants, Wang et al. [227] improve the RRT* algorithm, and the authors also point out two limitations, one is that the parameters designed are not applicable to other environments, and the other is that it fails to solve the problem of real-time dynamic obstacle avoidance.

In [228], the authors design a snake robot and investigate the effect of dynamic obstacles velocity on its path planning. The results show that the success rate of dynamic path planning gradually decreases as the velocity of dynamic obstacles increases, although the path planning time remains almost the same. This reveals that highly dynamic environments substantially increase the difficulty of the path planning task.

(2) Aerial inspection robots

To address the difficulty of observing the faults of high-altitude equipment in substations, Yang et al. [229] construct an architecture involving sensor technology, edge computing, and UAV substation inspection, and design a path planner using UAV physical constraints, sensor operation constraints, and inspection task constraints as the constraints, and inspection distance, time, and energy consumption as evaluation metrics. Zhao et al. [165] also do research on multi-objective cooperative path planning for electric inspection robot. Since only simulation experiments are done, the application effect of the algorithm in real scenarios is difficult to evaluate, and the authors also mention that future research should focus on substation inspection in complex environments such as plateaus, where the problem may still have challenges. Fang [230] et al. do similar work, but the authors further explore the effectiveness of different numbers of aircraft performing inspection tasks at different intervals. Similar work and findings also includes UAV trajectory planning for track inspection of rail-mounted gantry cranes in container terminal yards in port environments [231,232].

To address the subway tunnel environment with poor light and no GPS signal, Zhou et al. [233] equip MB1212 sonar and HOKUYO UTM-30LX LiDAR on a quadcopter and validate RRT with regression filtering mechanism and trunk constraint. Similarly, the authors only use a quadcopter to obtain the map information of the tunnels and verify the performance of the proposed algorithm on the simulation software, so many issues remain unresolved, such as the stability, accuracy, real-time performance of the scene modeling, the percentage of the actual inspected area, and the communication constraints.

The olive fly is a pest that affects the quality of olives in the Mediterranean region and one strategy is the use of insect traps. In order to realize the inspection of insect traps in the olive tree area, Gabriel et al. [234] verify the performance of Dijkstra, genetic algorithm and RRT + DQN in a simulation environment built with Gazobo, where the Dijkstra algorithm, although it performs the best in terms of time, generates paths with a low accuracy (1 m error), and the genetic algorithm, while the genetic algorithm generates smooth paths but has the highest time cost. In contrast, RRT + DQN is more suitable for this scenario. Unlike traditional simulation experiments, the authors create a simulation environment that attempts to match reality, including adding the perturbed environments, non-random robots behavior, and stochastic functions to dynamically adjust the velocity of objects. However, DQN's performance in large-scale (>400 cubic meters) challenging environments is far from optimal, in which case the authors suggest that large tasks can be countered by dividing them into multiple subtasks. Future directions on this topic mentioned by authors include testing in real scenarios and fusion of different types of sensors to obtain multi-source data.

In order to improve the durability and accuracy of pipeline array inspection in chemical plants, Alejandro et al. [235] propose an aerial manipulation robot with a rolling base. In this system, a human operator determines the inspection point and manipulates the robot to realize the pipeline inspection task under human-robot collaboration, and the RRT algorithm used to assist the human operator in generating paths for the inspection task is also validated in an outdoor scenario as well. In addition, considering the specificity of the pipeline inspection task, the authors also design a slender robot implemented with the RRT path planning algorithm to realize the inspection of narrow spaces such as pipelines under human-robot collaboration. For another robot specifically designed robot for pipeline inspection of gas-insulated switchgear, see Ref. [236].

For hazardous inspection tasks in open-air warehouse environments, Zhang et al. [237] perform similar work using the high maneuverability and aerial view characteristics of UAVs, and they also note that countermeasures to dynamic changing environments to improve UAV reaction time to emergencies are a focus for future work.

(3) Underground inspection robots

In subsea inspection tasks, traditional strategies utilize sensors to passively collect data; these paths can be optimized by taking into account the characteristics of the sensors, but cannot predict the environmental conditions, therefore, they do not guarantee access to data across the entire domain of interest and may lead to repeated attempts. To address this problem, Leonardo et al. [78] build the forward-looking sonar seabed inspections framework, present a RRT-based sensor-driven receding horizon approach, and introduce the entropy-based evaluation metric for solution searching. Comparative results show the proposed algorithm achieves the same coverage area rate at a shorter path cost.

The effectiveness of the seabed inspection tasks is related to the quality of the data acquired by sonar sensors, which depends on the characteristics of the environment, the target, and the sensor, which are also difficult to predict. To address this issue, Zacchini et al. [238] design a sensor-driven path planning method. The proposed framework realizes the inspection path planning task in a 67 m*67 m*4 m seabed area in a few minutes and can cover more than 90 % of the area to be inspected, but with the following problems: (1) For some special environments, special attention needs to be paid to certain areas of interest, but the strategy of passively acquiring data through sensors is inefficient in inspections. (2) The modeling quality of the experimental scenarios is closely related to the quality of the data, but for undetected regions, the characteristics of the obstacles or targets are difficult to predict, and therefore the modeling quality is unstable and unpredictable. (3) High-precision sensors for environmental modeling contain large amounts of data and are accurate but time-consuming, while low-precision sensors are able to model the scene quickly but with low accuracy.

In order to obtain cleaner and better quality seafood from distant waters, large net cages are used for seafood farming, and damaged cages lead to huge economic losses. Aiming at this issue, Wu et al. [239] provide robust path tracking algorithms, in which RRT is introduced to correct the trajectory when the underwater vehicle deviates from the desired path.

4.3. Other applications of human-robot interaction

4.3.1. Crowd avoidance

Inspired by the social force model (SFM), Henderson et al. [240] introduce the social intention model into RRT to design paths capable of interacting with humans, and to clearly evaluate social interaction-based path performance, the authors introduce the social effort index (SEI) as a novel benchmark. The authors conduct experiments in three scenarios: In the oncoming traffic scenario, the robot is required to avoid collisions with oncoming crowds, and the results show that the median SEI of RRT-SMP is on average 86.3 % lower than that of typical RRT-embedded MP (RRT-MP). In the bi-directional intersections, the robot is required to avoid crowds in two-way traffic and at crosswalks, and the results show that the median SEI of RRT-MP increase significantly with increasing population density, whereas the median SEI of RRT-SMP does not increase significantly, implying that RRT-SMP is more socially competent in chaotic environments. The SEI performance of RRT-SMP and RRT-MP on the moving with the crowd is also similar to the bi-directional intersections case, showing the result that RRT-SMP is more socially competent in chaotic environments, but the conclusion is not verified in a real scenario.

4.3.2. Repetitive tasks requiring human cognitive assistance

Robots excel at simple repetitive tasks, whereas humans possess unique cognitive skills for handling a variety of different tasks, and are therefore more likely to handle repetitive tasks in unstructured environments with human collaboration. In Ref. [241], Kelly et al. design a human-assisted RRT path planner and validate on a Franco-Emika Panda robotic arm, which overcomes the constraint of actuator saturation and limited joint ranges and avoids the property of RRT that it tends to fall into the local optimal solution, and the authors summarize the conclusions as follows: (1) The planner takes 30–130 ms to plan paths with varying levels of obstacle complexity, so the human-robot collaborative system is able to perform path-planning tasks in complex scenarios when the rate of change of dynamic obstacles in the scene is less than 1 Hz. (2) The trajectory-based explicit reference governor, as a closed feedback control scheme, has a maximum average computation time of 1 ms and can therefore be neglected with respect to the path planner. (3)

Planners with human assistance are able to avoid falling into local minima in some cases compared to planners lacking human assistance. (4) The safety of the constrained planned path can only be demonstrated when the robot is moving slowly, since a fast-moving robot cannot stop moving immediately. Human-assisted RRT to surgical robots [210] and autonomous driving [242], Customized plug-ins for collaborative autonomous path planning of multi-manipulator collaboration [243], is presented in other sections.

4.3.3. Human driving experience assisted vehicle interaction

Due to the large number of decision-making behaviors during interaction with other vehicles, such as determining the safe distance, whether to overtake or not, and how the speed changes dynamically, human driving experience assisted vehicle interaction tasks are also important. Chi et al. [242] consider the process of interaction of unmanned vehicle with other moving vehicles. The authors specify the desired speed with a look-ahead distance (a safe distance to avoid emergency braking by other vehicles in front) of the unmanned vehicle, while the vehicle in front (viewed as a moving obstacle) moves at a slow speed. The results show that the unmanned vehicle goes through several phases; in the first phase, the distance between the position of the unmanned vehicle and the moving obstacle decreases gradually due to the fact that the desired speed of the unmanned vehicle is greater than the moving obstacle; in the second phase, the path tracker evaluates the speed and the safe distance in an integrated manner and reduces the speed of the unmanned vehicle; in the third phase, when the environment-awareness sensors detect that there is no vehicle in the neighboring lane, the unmanned vehicle performs an overtaking operation; in the fourth stage, it changes back to its original lane and adjusts the speed to the desired speed; and in the fifth stage, it arrives at the target position and stops.

4.3.4. Multi-robot cooperation

Multi-robot collaboration without human intervention is described in other sections, such as multi-robot systems with line-of-sight constrained communication [205], decentralized multi-UAV cooperative exploration [204], online cooperative path planning for multi-quadrotor maneuvering in unknown dynamic environments [147], formation shape generation for multi-UAV [40], multi-unmanned surface vehicle path planning under spatially varying currents [127], cooperative path planning for multiple shipboard aircraft [181]. However, in some tasks, the lack of guidance from human experience reduces the reliability of path planning.

For the path planning problem in hostile environments, the classical path planning method fails when the risk characteristics cannot be accurately labeled manually, to address this problem, Guo et al. [182] reduce the risk of generating paths based on the data-driven approach that constraints the growth direction of RRT*. The authors also mention that multi-UAV risk aversion and the need for further balance between unbiased and biased sampling also deserve attention.

Ramasamy et al. [243] investigate a multi-robot based plug-and-play system suitable for a multi-robot system that can be operated online during the manufacturing process, which utilizes an improved RRT algorithm to automatically plan the paths of multi-manipulator collaboration. The results show that the plug-in exhibits its superiority in terms of energy consumption, path quality, and path generation efficiency. The hyper-parameters involved in such customized plug-ins require large a priori datasets or manual guidance.

5. Challenges and future trends

5.1. Challenges

This paper reviews the RRT improvement strategies in the last three years in terms of branching strategy, sampling strategy, post-processing and model-driven RRT. From the above review, it can be seen that the research of RRT algorithm has two directions roughly: theoretical research and application research, the former is aimed at optimization, such as path length, time complexity, space complexity, etc., while the latter takes more consideration of various problems in real scenarios, such as model constraints, scenario constraints. Theoretically, the improved RRT algorithms are capable of accomplishing real-time path planning tasks for large-scale, complex scenes with multiple objectives, e.g., the literature [226] plans paths with 25 objectives on a map containing more than 800,000 nodes and 1,000,000 edges in only 0.44 s. Maps with narrow passages are generally considered challenging for RRT, however, literature [227] uses an improved RRT* algorithm to plan paths on maps containing multiple narrow passages, and due to the advantage of efficient path generation in RRT-based algorithm, the search time is less than 1 s, even when the algorithm has more than 10,000 iterations, and thus is sufficiently tailored to the application scenario. However, from the above literature, the application of improved RRT-based algorithms is still challenging, and the main issues are discussed in the following sections.

5.1.1. Modeling of unknown environments

The need to model unknown scenarios is largely due to GPS denial, in which case many researchers also load sensors on robots to build maps, such as sonar for seabed inspection [78,238] and tunnel inspection [233], LiDAR for indoor inspection [201], tunnel inspection [233] and fire guidance [202], and depth camera for indoor scene modeling [204]. This environment exploration strategy has been proven to be effective, for example, three UAVs each loaded with a depth camera take 209.4s to explore a 10 m*8 m*3 m room filled with various types of obstacles [204], whereas underwater robots loaded with sonar sensors detect more than 90 % of the 67 m*67 m*4 m seafloor area in a few minutes [238], but there are problems such as difficulty in selecting interest point regions, unstable modeling quality, and difficult trade-off between modeling accuracy and efficiency.

5.1.2. Limitation to highly dynamic environments

Although the RRT algorithm has been shown to have global convergence and asymptotic optimality [20], this is based on the globally static known environment, which means that improved RRT algorithms may not be able to cope with dynamically changing environments or partially (globally) unknown scenarios. Some literatures regard dynamically changing scenarios as a direction for future research, such as hazardous inspection tasks in open-air warehouse environments [237], solar power plant fault inspection [227], underground mining stopes [46], assembling parts with different geometries [200], some literatures considers dynamic scene changes, such as motion planning framework for the quadrotor [101,147], robot with nonlinear control affine dynamical system [160], however, the performance of the algorithm is not validated in real scenarios. Yuan et al. [161] conduct the robotic arm motion planning test to validate the dynamic obstacle avoidance performance of RRT in real scenarios, but the dynamic obstacles given in the experiment are added suddenly by human beings, and are not the dynamic obstacles that we usually think of as changing position over time, and this assumption is also unreasonable. For similar work, see Ref. [62], where the dynamic obstacles also change abruptly in that scenario. Such assumptions are not compatible with the design of moving obstacles in real scenarios.

Strictly speaking, the dynamic obstacle avoidance problem of RRT-based path planning algorithms in real highly dynamic environments remains unsolved due to reasons including the extremely fast accuracy degradation as the obstacle speed increases [228], extremely complex human vascular and tissue environment [211], safety issues encountered in overtaking [242].

5.1.3. Hyper-parameter settings

Due to the limitations of the algorithms themselves, many of the algorithm parameters need to be manually tuned for better performance in different scenarios. For example, the hyper-parameters of RRTs are different in different scenarios, and the tuning process of these hyper-parameters is unavoidable, and many studies have been done by manual tuning, such as search step size [52]. In the design of the cost function, since the evaluation of many tasks is multiple levels, for example, the evaluation of the arc welding robot path planning task consists of the minimum transmission path length, energy consumption, and joint smoothness [197], which are constrained by each other, it is also necessary to set the weights of these three metrics with the help of human experience. The choice of how the sensor acquires the data and the post-processing of the acquired data may also involve some hyper-parameters, for data-rich sensors such as LiDAR, experiments show that an increase in detection distance within a certain range improves the detection efficiency, but beyond this range, increasing the detection distance leads to a slight decrease in detection efficiency, while too large a detection range greatly increases the computational cost [201]. Properly filtering the local point cloud and selecting the appropriate point cloud downsampling accuracy also greatly improve the quality and efficiency of the scene description, but this still requires setting the parameters manually and implementing several experiments [203]. For multi-robot collaborative tasks, the optimal value for the number of robots also needs to be determined by several experiments [201]. In addition, the improved RRT may introduce new algorithms whose parameters need to be chosen manually in several experiments, for example, leaf clustering is one of the improvement strategies for RRT-based exploration, but unsupervised clustering still requires manual selection of cluster radius and cluster density [203]. In terms of frontier-based rescue mission, the selection of the frontier threshold is also critical [203].

5.1.4. No generalization of hyper-parameter settings

When the scenario is changed or the algorithm is adjusted, the manually-tuned parameters mentioned in Section 5.1.2 may also no longer be applicable, such as the step size of RRT [52,227], weights of multiple evaluation metrics under different missions [197], cluster radius and cluster density for leaf nodes of RRT-based algorithms [203].

5.1.5. Multi-robot collaboration task

Multi-robot collaboration involves more difficult tasks than individual robots, and the challenges are illustrated in several points.

(1) Centralized or decentralized architecture

The centralized architecture is easy to manage, but when the number of robots is too large, the amount of data in the central node increases dramatically [201]. Decentralized architecture is characterized by high performance, but data consistency is a challenge [128]. In addition, over-idealized assumptions about the decentralized architecture, including the fact that robots can brake instantaneously and that inter-robot communication is lossless and latency-free, lead to the framework that is theoretically feasible but far from satisfying the needs of practical applications [205].

(2) Efficient or high-precision sensors

Multiple UAVs loaded with depth cameras separately can detect known areas in a short time. However, low accuracy of scene modeling is inevitable due to the limitations of sensor accuracy. LiDAR is a sensor that allows for accurate scene modeling, but the real-time performance of the LiDAR-based RRT navigation is poor due to its large amount of data [204].

(3) Breadth or depth of exploration

In [203], the frontier threshold determines the level of exploration; Excessively low frontier thresholds result in many areas being potentially unexplored, and too high frontier thresholds result in many areas being thoroughly explored, which can lead to some relatively unimportant regions being over-explored, resulting in a significant increase in rescue time, so the search strategy is also crucial and challenging.

(4) Switching exploration targets back and forth

In the field of multi-objective exploration under multiple robots, either frontier-based methods or sampling-based methods suffers from the problem of one robot switching back and forth between two regions or two objectives, leading to difficulty in accomplishing the task [203].

(5) Rationalization of the dynamic division of tasks

Regions of the same size may vary greatly in the number of critical nodes, but this information may only be known gradually during the robot exploration process, so it is also important to assign tasks to each robot in advance and adjust task assignment strategy in real time. In addition, the problem that sample-based algorithms are fast in the early stages of detection and slow in the later stages of detection needs to be considered [204].

(6) Communication constraints

Since robots can only communicate with each other within line-of-sight, the safety hazards due to communication limitations in decentralized collaborative exploration tasks cannot be ignored. Victoria et al. [205] demonstrate the effectiveness of the improved RRT method to this problem, but over-idealized assumptions including that robots can be braked instantaneously, that inter-robot communication is lossless and latency-free, and static scenarios result in algorithms that are far from being adequate for practical applications.

5.1.6. *Poor real-time performance*

Although simulation experiments validate the real-time performance of RRT algorithms [79,226,227], such algorithms still perform unsatisfactorily in real application scenarios, for example, the improved path planning algorithm in coal mine roadways takes 86.12 s s under various constraints [167], due to the collision detection constraints between the six arms, the proposed AEB-RRT takes 1109 s, 307 s, 210.8 s and 447.5 s to generate weld trajectories of 831.5 mm, 353.7 mm, 554.0 mm and 543.7 mm, respectively, which is far from the requirements of coherent welding [52]. The average time for the three UAVs to explore an area of 10 m*8 m*3 m with obstacles is 209. 4s [204]. The assembly of lightweight structures for healthcare facilities is approximately 28 min [199].

5.1.7. *Restricted by upper limits in other areas*

For exploratory tasks, robots usually undertake the tasks of localization, mapping, partitioning and path planning simultaneously. Therefore, as the last tasks (path planning), the upper limit of the quality and efficiency of RRT is determined by the previous tasks, which means that inaccurate localization, mapping, and irrational task allocation all result in RRT being far away from the high-quality solution, and this problem cannot be solved at the path planning level alone.

5.1.8. *Unstable results*

Due to the stochastic nature of RRT, even in the same algorithm, the same set of hyper-parameters, and the same scenario, differences in initial values (e.g., initial pose of robots) lead to large differences in results [201], not to mention that real scenarios include a variety of disturbances such as communication constrain and dynamic obstacles.

5.1.9. *Large-scale environment*

Model-driven algorithms are the trend in the development of path planning incorporating artificial intelligence, but large-scale environments pose a challenge to these kinds of algorithms, for example, DQN's performance in large-scale (>400 cubic meters) environments is far from optimal [234]. In contrast, improved RRTs may still work well in large-scale scenarios, for example, RRT-Rope is designed with the expectation of finding suboptimal solutions in a short period of time, and results show the method works well in large-scale uncluttered environment [46]. However, this conclusion is only valid when only static obstacles are included and the scenario is globally known, so large-scale scenarios are still challenging for exploration tasks.

5.1.10. *Stability and long-term performance involving continuous operation*

Stability is critical in safety-related fields, represented by human-related surgical robots. Surgical robots overcome the problems of poor precision, long operating times, surgeon fatigue, and lack of a three-dimensional precise field of view in traditional surgical procedures. Current surgical robots are auxiliary systems that are not fully intelligent enough to perform surgeries on their own, but needs to provide assistance to surgeons through remote control and mechanical drive, but the major concern of patients undergoing surgery and their families is the ability to ensure the absolute safety of the surgical robot. Although the surgical robot undergoes a long period of debugging and calibration before each surgery, and the surgery is operated by experienced surgeons, it is still difficult to ensure that the operation of the surgical robot is foolproof, and worse still, the surgical robot fails to deal with the regulation of special situations, and once a safety hazard occurs, the robotic arm cannot make adjustments independently like a clinically experienced surgeon to minimize the harm. Therefore, at present, most of the surgical robots still belong to the primary stage of semi-automatic collaboration, for the stability of the power system and control system requirements are extremely high, if there is a sudden power failure and restart or a malfunction of the control system during surgery, the surgical robotic arm is likely to cause irreparable damage to the patient's organs. Although the probability of a safety incident occurring is low, once it happens, the damage to the patient's life and health is irreparable.

Liu et al. [244] propose a long-term perspective planning strategy based on RRT*. which is motivated by the fact that (1) Co-awareness provides critical information beyond the field of view, (2) Planning using perspective long-range sensing techniques allows earlier response to obstacles, improving path quality and path generation efficiency, and (3) the branching property of RRT* is easy for real-time online path planning and preserves probabilistic completeness. The strategy investigates the fusion of different vehicle perception beliefs and proposes a cooperative perception-based cost map to represent the uncertainty and transmission delay. Results of campus experiments at the National University of Singapore demonstrate the improvement of the algorithm in remote sensing. Although the paper claims to validate the long-term performance, no quantitative metrics are actually given, and the experimental scenarios are only conducted on campus, and no indication is given of how long the algorithm runs for.

Thus, while much of the literature describes their work as stable and highly reliable, e.g., no failures in 1000 trials [192], in practice, even with very low failure rates (Consider hardware, communication failures, etc.), some robots may fail to achieve full autonomy, especially in safety-related tasks, such as surgical robots [206]. On the other hand, the long-term performance of RRTs in continuous operation is hardly reported, the only relevant literature retrieved is from 2013 [244], but although the authors mention the long-term performance, they do not give detailed data, so there is no way to know the reliability of RRT under long-term operation. Overall, various studies have shown that RRT has good long-term performance in small and medium-sized environments [52,167,204,234,235], but this is also limited to the algorithmic level. This suggests that, first, the long-term stability of RRT in large-scale unstructured scenarios remains to be verified; second, since RRT itself does not enable robot autonomy, the long-term stability of RRT in real-world applications is also related to the control (path-tracking stability), sensors (environment-awareness stability), communication (signal-awareness stability), and power supply.

5.1.11. Challenges to artificial intelligence approaches

Data-driven artificial intelligence methods achieve many difficult problems, such as inaccurate modeling, hardly evaluable cost functions, and various problems related to human beings. However, there are some problems with artificial intelligence itself, the most important of which is the poor interpretability, which leads to the increasing difficulty in designing and improving the model, and optimizing the hyper-parameters, and secondly, the solution of the path planning problem under artificial intelligence is closely related to the quality of the data, and it is very difficult to obtain a large amount of high-quality data in some scenarios, which also leads to the failure of this type of methods.

5.1.12. Ethics and morality

Ethics and morality are also factors that cannot be ignored. Common ethical and moral issues in autonomous driving include overspeeding, dangerous overtaking, and failure to maintain a safe distance, and literature [242] discusses an overtaking operation process while maintaining a safe distance. Unlike other tasks, when it comes to metrics in safety-related domains, it is important to design a reasonable framework that weighs the "public transportation rules" and "utility" rather than just optimizing the driver's interests (e.g., energy consumption, minimum time, etc.). In addition, in some extreme cases, when autonomous vehicles fail to find a path that avoid all the risks, they are caught in a dilemma, which generally consists of two tricky scenarios (1) the "other-other" dilemma, where the choice fails to take into account the safety of different people outside the car. For example, the autonomous car is driving normally according to the traffic rules, and suddenly there are two children running in front of the driveway. If there are only two choices at this point, either drive straight ahead and run over the two children, or avoid the children and dash to the sidewalk and run over pedestrians. (2) The "other-me" dilemma, where the choice fails to take into account the safety of people sitting in the car and people outside the car. For example, the autonomous car is driving normally according to the tunnel entrance, and suddenly a child runs into the lane ahead, if there are only two choices at this time, run over the child straight ahead or avoid the child to crash into the wall at the side of the tunnel, resulting in the death of the person in the car.

RRT, as a path planning method, is also an important process for intelligent military robots, but there may be ethical and legal issues. For example, whether military robots choose lethal weapons or weapons of mass destruction when performing tasks and making decisions independently; whether international humanitarian law is obeyed to ensure the safety of civilians and non-combatants, and how to define the responsibility of robots or human beings in human-robot collaboration of combat robots.

5.2. Future trends

5.2.1. Multi-type robot collaboration

Typical unmanned scenarios of cross-domain collaboration include sea-air unmanned cluster systems and air-ground unmanned cluster systems; the former is mostly used for maritime combat and joint operation missions, such as sea area alert patrol, ocean detection, target reconnaissance, and coordinated anti-submarine warfare; the latter consists of ground-based unmanned systems such as UAV and unmanned vehicles, and its missions include cross-domain collaborative reconnaissance, strikes, and searches. Specifically, for example, in Ref. [199], the authors mention that full automation of lightweight structure assembly can be achieved by ground-based robots assembling the shell while aerial-based robots assist in tightening the bolts. In the field of pavement construction, pavers and rollers collaborate with each other to automate the entire process of pavement compaction tasks, and I and our team also propose a bi-directional path planning strategy for the idle robot and the malfunction robot to address the problem of insufficient fuel that may be involved in pavement construction machinery [245].

5.2.2. Human-robot collaboration

In some safety-critical applications, human-robot collaboration is a major trend for technical and safety reasons, for example, emotionally, patients and their families do not trust surgeries that are completely dependent on robots; technically, the complexity of the human tissue and organ environment also poses a great challenge for the full autonomy of surgical robots, and more critically, there is an inevitable failure rate of the hardware, so the stability of human-robot collaboration in surgery is much better [207,208], such as removing porcine gallbladder using a master-slave robotic arm within the framework of human-robot collaboration [210]. Under some accuracy-sensitive path planning requirements, such as ore transportation [222], the smoothed paths may no longer be asymptotically optimal solutions under the RRT framework, and may even generate unreasonable local paths, which makes manual intervention extremely important. Certain experimental results show that better performance can be achieved with human-assisted path planning [241], which is one of the conclusions that human-robot collaboration is the future trends. Balancing unbiased and biased sampling based on human experience is also one of the important research directions for RRT improvement [182]. In conclusion, all kinds of robots still have a bright future under human-robot collaboration.

5.2.3. Real-time path planning

Real-time path planning plays a decisive role in the efficiency of task completion, and examples of applications where tasks cannot be completed efficiently due to poor real-time performance are mentioned in Section 5.1.6, such as path planning for a coherent welding task [52] and path planning for a coal mine tunnel [167]. In addition, in some safety-critical domains, such as autonomous driving, failure to perform real-time path optimization and feedback control for sudden obstacles increases the risk of accidents, and thus the development of real-time path planning is still urgent from the perspectives of work efficiency and safety.

5.2.4. Self-tuning of the hyper-parameters

The self-tuning of hyper-parameters is also a very important research direction as many hyper-parameters change with different application scenarios or even different maps in the same application scenario, which is discussed in detail in Section 5.1.3. At present, swarm optimization algorithms or artificial intelligence algorithms have been capable of self-tuning the hyper-parameters of some algorithms, but there are still many algorithms whose hyper-parameters still need to be adjusted manually, which leads to a large amount of resource consumption. Artificial intelligence-driven path planning algorithms also suffer from self-tuning of network model hyper-parameters. Therefore, the self-tuning of hyper-parameters is one of the future trends.

5.2.6. Application-scenario-oriented algorithm and hardware design

At the algorithmic level, each algorithm has its advantages and disadvantages, so improving the existing algorithms for specific application scenarios is also a future trend. For example, although the proposed RRT-Rope in Ref. [46] cannot handle pillar-like structural obstacles, the authors apply it to a mining robot path planning scenario without such obstacles, and also achieve satisfactory results. At the hardware level, targeted hardware design is also critical, for example, Alejandro et al. [235] specifically design a slender robot for the pipe inspection task and further improve the RRT algorithm for the robot's kinematic characteristics, and this optimization, which includes both hardware and software system design, is more effective. In Ref. [236], a novel robot is also designed for pipeline inspection of gas-insulated switchgear. In the field of autonomous driving, it is also particularly important to customize some strategies, such as maintaining a safe distance and triggering overtaking conditions, taking into account self-interests, ethics, laws, and local field of view constraints, and the customized data can also be obtained through model-driven RRTs in human experience [183]. In the field of multi-robot and smart city infrastructure, such as the full automation of lightweight structure assembly mentioned in Ref. [199], aerial robots used to assist in tightening bolts also require customized designs. Therefore, task- or application-scenario-oriented algorithm and hardware design is also a future research trend.

5.2.5. Uncertainties handling

Uncertainty is caused by various factors, such as unknown time-varying environments, unknown types of perturbations, and communication connection interruptions, etc. Therefore, enhancing the system's resistance to interference under various constraints is also one of the research trends, and the solution of this problem can also help to improve the instability of the results described in Section 5.1.8.

5.2.7. Highly dynamic environments handling

Highly dynamic environments are extremely challenging. At the algorithmic level, such environments not only require stable and reliable generated paths, but also require algorithms with high computational efficiency, strong dynamic optimization capability, and even predictability; at the hardware level, sensors are required to accurately capture the rapid changes in the scene, and the sampling period of various types of hardware is also a determining factor that affects the ability of the robot to cope with highly dynamic environments.

6. Conclusion

This paper mainly reviews the articles published in the field of RRT in the past three years, and the author believes that although RRT-based improved algorithms have advantages in large-scale scenarios, real-time performance, and uncertain environments, and some strategies that are difficult to be quantitatively described can be designed based on model-driven RRT, there are still the problems of difficult hyper-parameter design and weak generalization ability. In the practical application level, the reliability and accuracy of the hardware such as controllers, actuators, sensors, communication, power supply and data acquisition efficiency in large-scale unstructured scenarios is still a challenge. In the field of multi-robot collaboration and human-robot collaboration, there are still a lot of problems to be solved. In summary, the author believe that multi-type robot collaboration, human-robot collaboration, real-time path planning, self-tuning of hyper-parameters, task- or application-scenario-oriented algorithm and hardware design, and path planning in highly dynamic environments are the future trends.

Data availability statement

No data is used for the research described in the article.

CRediT authorship contribution statement

Tong Xu: Writing – review & editing, Writing – original draft, Methodology, Formal analysis, Conceptualization.

Declaration of Competing interest

The author declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Acknowledgements

The author is very grateful to the experienced reviewers with broad perspectives for their valuable suggestions for improving the quality of the paper.

References

[1] S.M. LaValle, J.J. Kuffner, Rapidly-exploring random trees: progress and prospects, ALGORITHMIC Comput. Robot. NEW Dir (2001) 293–308.
[2] S.M. LaValle, J.J. Kuffner, Randomized kinodynamic planning, Int J Rob Res 20 (2001) 378–400, https://doi.org/10.1177/02783640122067453.
[3] C. Urmson, R. Simmons, Approaches for heuristically biasing RRT growth, in: IROS 2003 Proc. 2003 IEEE/RSJ Int. Conf. Intell. Robot. Syst vols. 1–4, IEEE, IEEE, 2003, pp. 1178–1183.
[4] X. Chen, Y. Zhao, J. Fan, H. Liu, Three-dimensional UAV Track Planning Based on the GB-PQ-RRT*algorithm, 2023.
[5] A. Li, C. Niu, S. Li, X. Huang, C. Xu, G. Liu, Research on intelligent vehicle trajectory planning and control based on an improved terminal sliding mode, Appl. Sci. 12 (2022), https://doi.org/10.3390/app12052446.
[6] Y. Liu, W. Tao, S. Li, Y. Li, Q. Wang, A path planning method with a bidirectional potential field probabilistic step size RRT for a dual manipulator, Sensors 23 (2023), https://doi.org/10.3390/s23115172.
[7] S. Yu, J. Chen, G. Liu, X. Tong, Y. Sun, SOF-RRT*: an improved path planning algorithm using spatial offset sampling, Eng. Appl. Artif. Intell. 126 (2023), https://doi.org/10.1016/j.engappai.2023.106875.
[8] J. Dai, Y. Zhang, H. Deng, Novel potential guided bidirectional RRT*With direct connection strategy for path planning of redundant robot manipulators in joint space, IEEE Trans. Ind. Electron. 71 (2024) 2737–2747, https://doi.org/10.1109/TIE.2023.3269462.
[9] J. Fan, X. Chen, X. Liang, UAV trajectory planning based on bi-directional APF-RRT* algorithm with goal-biased, Expert Syst. Appl. 213 (2023), https://doi.org/10.1016/j.eswa.2022.119137.
[10] J. Fan, X. Chen, Y. Wang, X. Chen, UAV trajectory planning in cluttered environments based on PF-RRT* algorithm with goal-biased strategy, Eng. Appl. Artif. Intell. 114 (2022), https://doi.org/10.1016/j.engappai.2022.105182.
[11] D. Han, H. Nie, J. Chen, M. Chen, Optimal randomized path planning for redundant manipulators based on Memory-Goal-Biasing, Int J Adv Robot Syst 15 (2018), https://doi.org/10.1177/1729881418787049.
[12] S. Ganesan, S.K. Natarajan, A. Thondiyath, A novel goal-oriented sampling method for improving the convergence rate of sampling-based path planning for autonomous mobile robot navigation, Def Sci J 73 (2023) 322–331, https://doi.org/10.14429/dsj.73.17888.
[13] G. Noh, J. Park, D. Han, D. Lee, Selective goal aiming rapidly exploring random tree path planning for UAVs, Int J Aeronaut Sp Sci 22 (2021) 1397–1412, https://doi.org/10.1007/s42405-021-00406-7.
[14] G. Kang, Y.B. Kim, Y.H. Lee, H.S. Oh, W.S. You, H.R. Choi, Sampling-based motion planning of manipulator with goal-oriented sampling, Intell Serv Robot 12 (2019) 265–273, https://doi.org/10.1007/s11370-019-00281-y.
[15] M.F. Aslan, A. Durdu, K. Sabanci, Goal distance-based UAV path planning approach, path optimization and learning-based path estimation: GDRRT*, PSO-GDRRT* and BiLSTM-PSO-GDRRT, Appl. Soft Comput. 137 (2023), https://doi.org/10.1016/j.asoc.2023.110156.
[16] T. Xu, Y. Xu, D. Wang, S. Chen, W. Zhang, L. Feng, Path planning for autonomous articulated vehicle based on improved goal-directed rapid-exploring random tree, Math. Probl Eng. (2020), https://doi.org/10.1155/2020/7123164, 2020.
[17] X.U. Tong, C. Siwei, W. Dong, W.U. Ti, X.U. Yang, Z. Weigong, A novel path planning method for articulated road roller using Support vector machine and longest accessible path with course correction, IEEE Access 7 (2019) 182784–182795, https://doi.org/10.1109/ACCESS.2019.2959346.
[18] D.B.E. Moses, G. Anitha, Goal directed approach to autonomous motion planning for unmanned vehicles, Def Sci J 67 (2017) 45–49, https://doi.org/10.14429/dsj.67.10295.
[19] A.A. Bastapure, S.S. Chiddarwar, A. Goyal, in: S. Sharma, B. Subudhi, U.K. Sahu (Eds.), A Comparative Study of A*, RRT, and RRT* Algorithm for Path Planning in 2D Warehouse Configuration Space BT - Intelligent Control, Robotics, and Industrial Automation, Springer Nature Singapore, Singapore, 2023, pp. 95–107.
[20] S. Karaman, E. Frazzoli, Sampling-based algorithms for optimal motion planning, Int J Rob Res 30 (2011) 846–894, https://doi.org/10.1177/0278364911406761.
[21] Ding Jun, Yinxuan Zhou, Xia Huang, Kun Song, L.W. Shiqing Lu, An improved RRT* algorithm for robot path planning based on path expansion heuristic sampling Jun, J. Comput. Sci. 67 (2023) 1–12.
[22] W. Lu, H. Xiong, Z. Zhang, Z. Hu, T. Wang, Scalable optimal formation path planning for multiple interconnected robots via convex polygon trees, J Intell Robot Syst 109 (2023), https://doi.org/10.1007/s10846-023-01994-0.
[23] F.J. Yu, Y. Chen, Cyl-IRRT*: homotopy optimal 3D path planning for AUVs by biasing the sampling into a cylindrical informed subset, IEEE Trans. Ind. Electron. 70 (2023) 3985–3994, https://doi.org/10.1109/TIE.2022.3177801.
[24] M.M. Alam, T. Nishi, Z. Liu, T. Fujiwara, A novel sampling-based optimal motion planning algorithm for energy-efficient robotic pick and place, Energies 16 (2023), https://doi.org/10.3390/en16196910.

[25] S. Odem, S. Hacohen, O. Medina, An RRT that uses MSR-equivalence for solving the self-reconfiguration task in lattice modular robots, IEEE Robot Autom Lett 8 (2023) 2922–2929, https://doi.org/10.1109/LRA.2023.3261705.
[26] I.-B. Jeong, S.-J. Lee, J.-H. Kim, Quick-RRT*: triangular inequality-based implementation of RRT* with improved initial solution and convergence rate, Expert Syst. Appl. 123 (2019) 82–90, https://doi.org/10.1016/j.eswa.2019.01.032.
[27] J.J. Kuffner, S.M. La Valle, RRT-connect: an efficient approach to single-query path planning, Proc - IEEE Int Conf Robot Autom 2 (2000) 995–1001, https://doi.org/10.1109/robot.2000.844730.
[28] B. Wu, W. Zhang, X. Chi, D. Jiang, Y. Yi, Y. Lu, A novel AGV path planning approach for narrow channels based on the Bi-RRT algorithm with a failure rate threshold, Sensors 23 (2023), https://doi.org/10.3390/s23177547.
[29] V. Rajendran, P. Carreno-Medrano, W. Fisher, D. Kulic, Human-aware RRT-connect: motion planning for safe human-robot collaboration, in: 2021 30TH IEEE Int. Conf. Robot Hum. Interact. Commun., IEEE, 2021, pp. 15–22.
[30] X.Y. Chen, X.J. You, J.C. Jiang, J.H. Ye, H.B. Wu, Trajectory planning of dual-robot cooperative assembly, Machines 10 (2022), https://doi.org/10.3390/machines10080689.
[31] B. Li, B. Chen, An adaptive rapidly-exploring random tree, IEEE/CAA J Autom Sin 9 (2022) 283–294, https://doi.org/10.1109/JAS.2021.1004252.
[32] A. Tahirovic, M. Ferizbegovic, Rapidly-exploring random vines (RRV) for motion planning in configuration spaces with narrow passages, in: 2018 IEEE Int. Conf. Robot. Autom., IEEE, 2018, pp. 7055–7062.
[33] L. Cao, L. Wang, Y. Liu, S. Yan, 3D trajectory planning based on the Rapidly-exploring Random Tree–Connect and artificial potential fields method for unmanned aerial vehicles, Int J Adv Robot Syst 19 (2022) 1–17, https://doi.org/10.1177/17298806221118867.
[34] L.H. Jiang, S.Y. Liu, Y.M. Cui, H.X. Jiang, Path planning for robotic manipulator in complex multi-obstacle environment based on improved_RRT, IEEE-ASME Trans MECHATRONICS 27 (2022) 4774–4785, https://doi.org/10.1109/TMECH.2022.3165845.
[35] Y. Li, S. Ma, Navigation of apple tree pruning robot based on improved RRT-connect algorithm, AGRICULTURE-BASEL 13 (2023), https://doi.org/10.3390/agriculture13081495.
[36] Y.Y. Chen, Y.X. Fu, B. Zhang, W. Fu, C.J. Shen, Path planning of the fruit tree pruning manipulator based on improved RRT-Connect algorithm, Int. J. Agric. Biol. Eng. 15 (2022) 177–188, https://doi.org/10.25165/j.ijabe.20221502.6249.
[37] L.A. Wang, X. Yang, Z.L. Chen, B.R. Wang, Application of the improved rapidly exploring random tree algorithm to an insect-like mobile robot in a narrow environment, Biomimetics 8 (2023), https://doi.org/10.3390/biomimetics8040374.
[38] J.F. Chang, N. Dong, D.H. Li, W.H. Ip, K.L. Yung, Skeleton extraction and greedy-algorithm-based path planning and its application in UAV trajectory tracking, IEEE Trans Aerosp Electron Syst 58 (2022) 4953–4964, https://doi.org/10.1109/TAES.2022.3198925.
[39] T. Elmokadem, A.V. Savkin, A hybrid approach for autonomous collision-free UAV navigation in 3D partially unknown dynamic environments, DRONES 5 (2021), https://doi.org/10.3390/drones5030057.
[40] T.H. Gong, Y. Yu, J.H. Song, Path planning for multiple unmanned vehicles (MUVs) formation shape generation based on dual RRT optimization, Actuators 11 (2022), https://doi.org/10.3390/act11070190.
[41] X. Cheng, J.M. Zhou, Z. Zhou, X.M. Zhao, J.J. Gao, T. Qiao, An improved RRT-Connect path planning algorithm of robotic arm for automatic sampling of exhaust emission detection in Industry 4.0, J Ind Inf Integr 33 (2023), https://doi.org/10.1016/j.jii.2023.100436.
[42] Z.J. Huan, J.M. Wang, L. Zhu, Z.X. Zhong, W.C. Ma, Z.F. Chen, Navigation and closed-loop control of magnetic microrobot in plant vein mimic environment, Front. Plant Sci. 14 (2023), https://doi.org/10.3389/fpls.2023.1133944.
[43] S. Li, D. Zhao, Y. Sun, J. Yang, S. Wang, Path planning algorithm based on the improved RRT-connect for home service robot arms, in: 2021 IEEE Int. Conf. Intell. Saf. Robot., IEEE, 2021, pp. 403–407.
[44] A. Spinos, M. Yim, A linking invariant for truss robot motion planning, IEEE Robot Autom Lett 7 (2022) 1424–1430, https://doi.org/10.1109/LRA.2021.3139941.
[45] J.G. Kang, D.W. Lim, Y.S. Choi, W.J. Jang, J.W. Jung, Improved RRT-connect algorithm based on triangular inequality for robot path planning, SENSORS 21 (2021), https://doi.org/10.3390/s21020333.
[46] L. Petit, A.L. Desbiens, RRT-Rope: a deterministic shortening approach for fast near-optimal path planning in large-scale uncluttered 3D environments, in: 2021 IEEE Int. Conf. Syst. MAN, Cybern, IEEE, 2021, pp. 1111–1118.
[47] J. Dai, Y. Zhang, H. Deng, Bidirectional RRT*-Based path planning for tight coordination of dual redundant manipulators, Machines 11 (2023), https://doi.org/10.3390/machines11020209.
[48] R. Singh, Optimized trajectory planning for the time efficient navigation of mobile robot in constrained environment, Int J Mach Learn Cybern 14 (2023) 1079–1103, https://doi.org/10.1007/s13042-022-01684-7.
[49] P. Sharma, A. Gupta, D. Ghosh, V. Honkote, G. Nandakumar, D. Ghose, PG-RRT: a Gaussian mixture model driven, kinematically constrained Bi-directional RRT for robot path planning, in: 2021 IEEE/RSJ Int. Conf. Intell. Robot. Syst., IEEE, 2021, pp. 3666–3673.
[50] H. Wang, X. Liu, S. Song, B. Li, X. Lu, J. Nie, X. Zhao, Improved RRT path planning algorithm based on growth evaluation, Int. Conf. Intell. Equip. Spec. Robot. (ICIESR 2021) (2021) 12127.
[51] T.H. Hsieh, S.Z. Wang, H.J. Gong, W. Liu, N. Xu, Sea ice warning visualization and path planning for ice navigation based on radar image recognition, J. Mar. Sci. Technol. 29 (2021) 277–286.
[52] X.W. Wang, J.B. Wei, X. Zhou, Z.L. Xia, X.S. Gu, AEB-RRT*: an adaptive extension bidirectional RRT* algorithm, Auton Robots 46 (2022) 685–704, https://doi.org/10.1007/s10514-022-10044-x.
[53] Q. Fu, X. Lan, Y. Ji, X. Sun, F. Ren, Heuristic RRT Fusion A* for 3D Path Planning of UAV, 2022.
[54] J. Li, L. Li, J. Qiang, H. Wang, Y. Cao, Fast path planning based on Bi-directional RRT{*}for mobile robot in complex maze environments, in: 2023 42nd Chinese Control Conference (CCC), 2023, pp. 4768–4772.
[55] M. Zhang, Y. Ma, N. Xi, Bi-RRT{*} based trajectory optimization and obstacle avoidance for a serial manipulator, in: 2021 IEEE 11th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER), 2021, pp. 163–168.
[56] Y. Shan, Z. Zhong, L. Fan, H. Kai, DeepParking: deep learning-based planning method for autonomous parking, in: X. Meng, X. Li, J. Xu, X. Zhang, Y. Fang, B. Zheng, Y. Li (Eds.), Spatial Data and Intelligence: 4th International Conference, SpatialDI 2023, Nanchang, China, April 13-15, 2023, Proceedings, 2023, pp. 21–33. Lecture Notes in Computer Science (13887.
[57] Y. Wang, Q. Yuan, Y. Guo, W. Han, Path planning for lunar rover based on Bi-RRT algorithm, in: 2021 Proc. 40TH Chinese Control Conf., 2021, pp. 4211–4216.
[58] L. Ye, J.L. Duan, Z. Yang, X.J. Zou, M.Y. Chen, S. Zhang, Collision-free motion planning for the litchi-picking robot, Comput. Electron. Agric. 185 (2021), https://doi.org/10.1016/j.compag.2021.106151.
[59] B. He, J. Lee, U. Topcu, L. Sentis, BP-RRT: barrier pair synthesis for temporal logic motion planning, in: 2020 59th IEEE Conference on Decision and Control (CDC), 2020, pp. 1404–1409.
[60] K. Hao, Y. Yang, Z. Li, Y. Liu, X. Zhao, CERRT: a mobile robot path planning algorithm based on RRT in complex environments, Appl. Sci. 13 (2023), https://doi.org/10.3390/app13179666.
[61] J. Guo, G. Lu, T. Chen, X. Rong, Y. Li, Z. Wang, H. Zhao, J. Zhang, Autonomous legged robot navigation with environment awareness system in complex outdoor environments, in: 2021 IEEE International Conference on Robotics and Biomimetics (ROBIO), 2021, pp. 1117–1122.
[62] J. Qi, H. Yang, H.X. Sun, MOD-RRT*: a sampling-based algorithm for robot path planning in dynamic environment, IEEE Trans. Ind. Electron. 68 (2021) 7244–7251, https://doi.org/10.1109/TIE.2020.2998740.
[63] Z.R. Xu, G.P. Kontoudis, K.G. Vamvoudakis, Online and robust intermittent motion planning in dynamic and changing environments, IEEE Trans NEURAL NETWORKS Learn Syst (2023), https://doi.org/10.1109/TNNLS.2023.3303811.
[64] H. Yin, L. Zhang, Path planning algorithm based on RRT-A* in 3D terrain, in: 2023 IEEE International Conference on Mechatronics and Automation, ICMA), 2023, pp. 2315–2319.

[65] Y. Wang, W.S. Jiang, Z. Luo, L. Yang, Y.Q. Wang, Path planning of a 6-DOF measuring robot with a direction guidance RRT method, Expert Syst. Appl. 238 (2024), https://doi.org/10.1016/j.eswa.2023.122057.
[66] C. Liu, Q.C. Feng, Z.L. Tang, X.Y. Wang, J.P. Geng, L.J. Xu, Motion planning of the citrus-picking manipulator based on the TO-RRT algorithm, AGRICULTURE-BASEL 12 (2022), https://doi.org/10.3390/agriculture12050581.
[67] Y.Z. Shang, F. Liu, P. Qin, Z.Z. Guo, Z. Li, Research on path planning of autonomous vehicle based on RRT algorithm of Q-learning and obstacle distribution, Eng. Comput. 40 (2023) 1266–1286, https://doi.org/10.1108/EC-11-2022-0672.
[68] Y. Wang, D. Liu, H.M. Zhao, Y.L. Li, W.M. Song, M.L. Liu, L. Tian, X.H. Yan, Rapid citrus harvesting motion planning with pre-harvesting point and quad-tree, Comput. Electron. Agric. 202 (2022), https://doi.org/10.1016/j.compag.2022.107348.
[69] H.H. Shen, W.F. Xie, J.Y. Tang, T. Zhou, Adaptive manipulability-based path planning strategy for industrial robot manipulators, IEEE-ASME Trans MECHATRONICS 28 (2023) 1742–1753, https://doi.org/10.1109/TMECH.2022.3231467.
[70] Q. Diao, J. Zhang, M. Liu, J. Yang, A disaster relief UAV path planning based on APF-IRRT* fusion algorithm, Drones 7 (2023), https://doi.org/10.3390/drones7050323.
[71] J. Peng, Y. Chen, Y. Duan, Y. Zhang, J. Ji, Y. Zhang, Towards an online RRT-based path planning algorithm for ackermann-steering vehicles, in: 2021 IEEE Int. Conf. Robot. Autom. (ICRA 2021), IEEE, 2021, pp. 7407–7413.
[72] Q. Li, H. Zhao, M. Zhang, Z. Sun, A path planning algorithm for mobile robots based on DGABI-RRT, Intell. Robot. Appl. ICIRA 2021, PT IV (13016) (2021) 554–564.
[73] J. Nasir, F. Islam, U. Malik, Y. Ayaz, O. Hasan, M. Khan, M.S. Muhammad, RRT{*}-SMART: a rapid convergence implementation of RRT, Int J Adv Robot Syst 10 (2013), https://doi.org/10.5772/56718.
[74] F. Islam, J. Nasir, U. Malik, Y. Ayaz, O. Hasan, RRT{*}-Smart: rapid convergence implementation of RRT{*} towards optimal solution, in: 2012 IEEE International Conference on Mechatronics and Automation (ICMA), 2012, pp. 1651–1656.
[75] C. Chung, C.H. Yang, A 1.5-μJ/Task path-planning processor for 2-D/3-D autonomous navigation of microrobots, IEEE J Solid-State Circuits 56 (2021) 112–122, https://doi.org/10.1109/JSSC.2020.3037138.
[76] F. Grothe, V.N. Hartmann, A. Orthey, M. Toussaint, ST-RRT*: asymptotically-optimal bidirectional motion planning through space-time, in: 2022 IEEE Int. Conf. Robot. Autom. (ICRA 2022), IEEE, 2022, pp. 3314–3320.
[77] J. Zhang, J. Li, H. Yang, X. Feng, G. Sun, Complex environment path planning for unmanned aerial vehicles, Sensors 21 (2021) 1–29, https://doi.org/10.3390/s21155250.
[78] L. Zacchini, M. Franchi, A. Ridolfi, Sensor-driven autonomous underwater inspections: a receding-horizon RRT-based view planning solution for AUVs, J F Robot 39 (2022) 499–527, https://doi.org/10.1002/rob.22061.
[79] R. Xu, Y. Fan, Z. Li, S. Zhu, H. Shang, A. Gao, Time-optimal attitude planning for spacecraft with movable parts using second order cone programming, Aerosp Sci Technol 141 (2023), https://doi.org/10.1016/j.ast.2023.108589.
[80] H.Y. Long, G. Li, F.L. Zhou, T.F. Chen, Cooperative dynamic motion planning for dual manipulator arms based on RRT*Smart-AD algorithm, SENSORS 23 (2023), https://doi.org/10.3390/s23187759.
[81] Y. Song, Y. Chen, J. Gao, Y. Wang, G. Pan, Collision avoidance strategy for unmanned surface vessel considering actuator faults using kinodynamic rapidly exploring random tree-smart and radial basis function neural network-based model predictive control, J. Mar. Sci. Eng. 11 (2023), https://doi.org/10.3390/jmse11061107.
[82] J. Wang, B. Li, M.Q.H. Meng, Kinematic constrained Bi-directional RRT with efficient branch pruning for robot path planning, Expert Syst. Appl. 170 (2021) 1–7, https://doi.org/10.1016/j.eswa.2020.114541.
[83] J. Swedeen, G. Droge, R. Christensen, Fillet-based RRT * : a rapid convergence implementation of RRT* for curvature constrained vehicles, J Intell Robot Syst Theory Appl 108 (2023), https://doi.org/10.1007/s10846-023-01846-x.
[84] H. Ji, H. Xie, C. Wang, H. Yang, E-RRT*: path planning for hyper-redundant manipulators, IEEE Robot Autom Lett 8 (2023) 8128–8135, https://doi.org/10.1109/LRA.2023.3325716.
[85] W.H. Xue, B.C. Wang, X.X. Huang, B. Yang, Z.K. Wen, H. Zhang, S. Li, Spacecraft attitude maneuver planning with multi-sensor pointing constraints using improved RRT-star algorithm, Adv Sp Res 72 (2023) 1485–1495, https://doi.org/10.1016/j.asr.2023.04.024.
[86] W. Lan, X. Jin, T.L. Wang, H. Zhou, Improved RRT algorithms to solve path planning of multi-glider in time-varying ocean currents, IEEE Access 9 (2021) 158098–158115, https://doi.org/10.1109/ACCESS.2021.3130367.
[87] Q. Chai, Y. Wang, RJ-RRT: improved RRT for path planning in narrow passages, Appl. Sci. 12 (2022), https://doi.org/10.3390/app122312033.
[88] H. Tu, Y. Deng, Q. Li, M. Song, X. Zheng, Improved RRT global path planning algorithm based on Bridge Test, Rob Auton Syst 171 (2024) 1–10, https://doi.org/10.1016/j.robot.2023.104570.
[89] Z. Wu, Z. Meng, W. Zhao, Z. Wu, Fast-RRT: a RRT-based optimal path finding method, Appl. Sci. 11 (2021), https://doi.org/10.3390/app112411777.
[90] Z. Luo, Q. Li, J. Qiu, C. Zhu, Graft-RRT: an efficient method to find and converge global paths through intermediate points, in: 2022 International Conference on Machine Learning and Knowledge Engineering (MLKE), 2022, pp. 256–260.
[91] Y. Gan, B. Zhang, C. Ke, X. Zhu, W. He, T. Ihara, Research on robot motion planning based on RRT algorithm with nonholonomic constraints, Neural Process. Lett. 53 (2021) 3011–3029, https://doi.org/10.1007/s11063-021-10536-4.
[92] J.D. Gammell, S.S. Srinivasa, T.D. Barfoot, Informed RRT*: optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic, IEEE Int Conf Intell Robot Syst (2014) 2997–3004, https://doi.org/10.1109/IROS.2014.6942976.
[93] G. Uzun, A. Ozdemir, S. Bogosyan, n-Sliced informed RRT*: intelligent sampling-based path planning in high eccentricity informed ellipsis, in: 2022 IEEE 31ST Int. Symp. Ind. Electron, IEEE, 2022, pp. 741–746.
[94] Y. Chang, H. Liang, P. Zhao, Z. Li, J. Wang, On-road trajectory planning with spatio-temporal informed RRT*, in: Proc. 2022 IEEE Int. Conf. MECHATRONICS Autom. (IEEE ICMA 2022), IEEE, 2022, pp. 1425–1431.
[95] M.N. Vu, M. Schwegel, C. Hartl-Nesic, A. Kugi, Sampling-based trajectory (re)planning for differentially flat systems: application to a 3D gantry crane, IFAC Pap. 55 (2022) 33–40.
[96] J. Dai, D.F. Li, J.W. Zhao, Y.Q. Li, Autonomous navigation of robots based on the improved informed-RRT* algorithm and DWA, J Robot (2022), https://doi.org/10.1155/2022/3477265, 2022.
[97] S. Fathi, S. Fathi, V. Balali, Time-space conflict management in construction sites using discrete event simulation (DES) and path planning in unity, Appl. Sci. 13 (2023), https://doi.org/10.3390/app13148128.
[98] P.T. Kyaw, A.V. Le, P. Veerajagadheswar, M.R. Elara, T.T. Thu, N.H.K. Nhan, P.V. Duc, M.B. Vu, Energy-efficient path planning of reconfigurable robots in complex environments, IEEE Trans. Robot. 38 (2022) 2481–2494, https://doi.org/10.1109/TRO.2022.3147408.
[99] T.T. Enevoldsen, C. Reinartz, R. Galeazzi, COLREGs-informed RRT* for collision avoidance of marine crafts, in: Proc - IEEE Int Conf Robot Autom 2021-May, 2021, pp. 8083–8089, https://doi.org/10.1109/ICRA48506.2021.9560909.
[100] W. Li, M. Ren, Y. Zhu, Y. Zhang, S. Zhou, Y. Zhou, CC-BRRT: a path planning algorithm based on central circle sampling bidirectional RRT, WEB Inf. Syst. Appl. (WISA 2021) 12999 (2021) 430–441.
[101] B. Liu, S.M. Wei, Online safe trajectory generation of quadrotors autonomous landing in complex dynamic environments, Intell. Robot. Appl. (ICIRA 2022), PT IV (13458) (2022) 517–529.
[102] Z.W. Zhang, Y.W. Jia, Q.Q. Su, X.T. Chen, B.P. Fu, ATS-RRT*: an improved RRT* algorithm based on alternative paths and triangular area sampling, Adv. Robot. 37 (2023) 605–620, https://doi.org/10.1080/01691864.2023.2174817.
[103] X.C. Zhou, W. Zheng, Z.W. Li, P.L. Wu, Y.J. Sun, Enhancing path planning efficiency for underwater gravity matching navigation with a novel three-dimensional along-path obstacle profiling algorithm, Remote Sens 15 (2023), https://doi.org/10.3390/rs15235579.
[104] K.H. Wang, S.F. Li, Y. Wang, J. Xi, Smooth-RRT*: an improved motion planner for underwater robot, in: 2022 27TH ASIA PACIFIC Conf. Commun. (APCC 2022) Creat. Innov. Commun. Technol. POST-PANDEMIC ERA, IEEE, 2022, pp. 353–358.

[105] X. Guo, A. Li, P. Duan, L. Zhu, X. Wang, An efficient RRT{*}-based path planning for mobile robot with fast convergence rate, in: 2023 IEEE International Conference on Mechatronics and Automation (ICMA), 2023, pp. 2213–2217.
[106] Y.M. Liang, H.Y. Zhao, CCPF-RRT*: an improved path planning algorithm with consideration of congestion, Expert Syst. Appl. 228 (2023), https://doi.org/10.1016/j.eswa.2023.120403.
[107] B. Wang, D. Ju, F. Xu, C. Feng, Bi-RRT*: an improved bidirectional RRT* path planner for robot in two-dimensional space, IEEJ Trans. Electr. Electron. Eng. 18 (2023) 1639–1652, https://doi.org/10.1002/tee.23898.
[108] B.P. Wang, D.Y. Ju, F.Z. Xu, C. Feng, G.L. Xun, CAF-RRT*: a 2D path planning algorithm based on circular arc fillet method, IEEE Access 10 (2022) 127168–127181, https://doi.org/10.1109/ACCESS.2022.3226465.
[109] A.H. Qureshi, Y. Ayaz, Potential functions based sampling heuristic for optimal path planning, Auton Robots 40 (2016) 1079–1093, https://doi.org/10.1007/s10514-015-9518-0.
[110] Y. Li, W. Wei, Y. Gao, D. Wang, Z. Fan, PQ-RRT*: an improved path planning algorithm for mobile robots, Expert Syst. Appl. 152 (2020), https://doi.org/10.1016/j.eswa.2020.113425.
[111] B. Wang, D. Ju, Q. Li, C. Feng, SS-RRT{*}: a safe and smoothing path planner for mobile robot in static environment, in: 2022 China Automation Congress (CAC), 2022, pp. 3677–3682.
[112] C. Zhuge, Q. Wang, J. Liu, L. Yao, An improved Q-RRT* algorithm based on virtual light, Comput. Syst. Sci. Eng. 39 (2021) 107–119, https://doi.org/10.32604/csse.2021.016273.
[113] Z. Zhang, H. Liu, Q. Deng, J. Jiang, G. Li, Improved RRT{*} algorithm based on node density diffusion strategy, in: 2023 IEEE International Conference on Unmanned Systems (ICUS), 2023, pp. 573–579.
[114] L.F. Jia, Y.P. Huang, T. Chen, Y.X. Guo, Y.C. Yin, J. Chen, MDA plus RRT: a general approach for resolving the problem of angle constraint for hyper-redundant manipulator, Expert Syst. Appl. 193 (2022), https://doi.org/10.1016/j.eswa.2021.116379.
[115] I. Jeong, Y. Jang, J. Park, Y.K. Cho, Motion planning of mobile robots for autonomous navigation on uneven ground surfaces, J. Comput. Civ. Eng. 35 (2021), https://doi.org/10.1061/(ASCE)CP.1943-5487.0000963.
[116] Q.H. Li, J.H. Wang, H.M. Li, B.P. Wang, C. Feng, Fast-RRT*: an improved motion planner for mobile robot in two-dimensional space, IEEJ Trans. Electr. Electron. Eng. 17 (2022) 200–208, https://doi.org/10.1002/tee.23502.
[117] A.M. Novo, L. Lu, P. Campoy, FAST RRT* 3D-sliced planner for autonomous exploration using MAVs, Unmanned Syst. 10 (2022) 175–186, https://doi.org/10.1142/S2301385022500108.
[118] B. Liao, F.Y. Wan, Y. Hua, R.R. Ma, S.R. Zhu, X.L. Qing, F-RRT*: an improved path planning algorithm with improved initial solution and convergence rate, Expert Syst. Appl. 184 (2021), https://doi.org/10.1016/j.eswa.2021.115457.
[119] J.P. Cong, J.B. Hu, Y.Y. Wang, Z.H. He, L.X. Han, M.Y. Su, FF-RRT*: a sampling-improved path planning algorithm for mobile robots against concave cavity obstacle, COMPLEX Intell Syst 9 (2023) 7249–7267, https://doi.org/10.1007/s40747-023-01111-6.
[120] X. Qiu, Y.H. Li, R. Jin, Z. Zhao, J.J. Li, D.L. Lu, L.H. Ma, Improved F-RRT* algorithm for flight-path optimization in hazardous weather, Int J Aerosp Eng (2022), https://doi.org/10.1155/2022/1166968, 2022.
[121] R. Zhang, R. Chai, S. Chai, Y. Xia, A. Tsourdos, Design and practical implementation of a high efficiency two-layer trajectory planning method for AGV, IEEE Trans. Ind. Electron. 71 (2024) 1811–1822, https://doi.org/10.1109/TIE.2023.3250847.
[122] Y.Y. Liu, C.J. Li, H.X. Yu, C.H. Song, NT-ARS-RRT: a novel non-threshold adaptive region sampling RRT algorithm for path planning, J KING SAUD Univ Inf Sci 35 (2023), https://doi.org/10.1016/j.jksuci.2023.101753.
[123] C. Zong, X. Han, D. Zhang, Y. Liu, W. Zhao, M. Sun, Research on local path planning based on improved RRT algorithm, Proc. Inst. Mech. Eng. - Part D J. Automob. Eng. 235 (2021) 2086–2100, https://doi.org/10.1177/0954407021993623.
[124] J. Xu, K.S. Park, Kinematic performance-based path planning for cable-driven parallel robots using modified adaptive RRT, Microsyst Technol Nanosyst STORAGE Process Syst 28 (2022) 2325–2336, https://doi.org/10.1007/s00542-022-05319-3.
[125] P.L. Zhao, Y.H. Chang, W.K. Wu, H.Y. Luo, Z.X. Zhou, Y.P. Qiao, Y. Li, C.H. Zhao, Z.A. Huang, B.J. Liu, X.J. Liu, S. He, D.H. Guo, Dynamic RRT: fast feasible path planning in randomly distributed obstacle environments, J Intell Robot Syst 107 (2023), https://doi.org/10.1007/s10846-023-01823-4.
[126] J. Leu, G. Zhang, L. Sun, M. Tomizuka, Efficient robot motion planning via sampling and optimization, in: 2021 Am. Control Conf., IEEE, 2021, pp. 4196–4202.
[127] J.B. Yu, Z.H. Chen, Z.Y. Zhao, P. Yao, J.P. Xu, A traversal multi-target path planning method for multi-unmanned surface vessels in space-varying ocean current, Ocean Eng 278 (2023), https://doi.org/10.1016/j.oceaneng.2023.114423.
[128] X. Wang, J. Gao, X. Zhou, X. Gu, Path planning for the gantry welding robot system based on improved RRT, Robot Comput Integr Manuf 85 (2024), https://doi.org/10.1016/j.rcim.2023.102643.
[129] A.H. Qureshi, Y. Ayaz, Intelligent bidirectional rapidly-exploring random trees for optimal motion planning in complex cluttered environments, Rob Auton Syst 68 (2015) 1–11, https://doi.org/10.1016/j.robot.2015.02.007.
[130] L. Sun, X. Duan, K. Zhang, P. Xu, X. Zheng, Q. Yu, Y. Luo, Improved path planning algorithm for mobile robots, Soft Comput. 27 (2023) 15057–15073, https://doi.org/10.1007/s00500-023-08674-z.
[131] Q.L. Zhanga, Z.P. Wang, J.G. Duana, J.Y. Qina, Robot motion planning with orientational constraints based on offline sampling datasets, J. Comput. Methods Sci. Eng. 22 (2022) 1545–1557, https://doi.org/10.3233/JCM-226140.
[132] Z. Zeng, C. Xiong, X. Yuan, H. Zhou, Y. Bai, Y. Jin, D. Lu, L. Lian, Information-driven path planning for hybrid aerial underwater vehicles, IEEE J. Ocean. Eng. 48 (2023) 689–715, https://doi.org/10.1109/JOE.2023.3267783.
[133] G.H. Huang, Q.L. Ma, Research on path planning algorithm of autonomous vehicles based on improved RRT algorithm, Int J Intell Transp Syst Res 20 (2022) 170–180, https://doi.org/10.1007/s13177-021-00281-2.
[134] L.H. Li, Z.H. Peng, C.X. Wen, P.Q. Shang, J.Q. Cui, Joint path and multi-hop communication node location planning in cluttered environment, J Adv Comput Intell Intell INFORMATICS 27 (2023) 664–672, https://doi.org/10.20965/jaciii.2023.p0664.
[135] N. Wen, R. Zhang, G. Liu, J. Wu, X. Qu, Online planning low-cost paths for unmanned surface vehicles based on the artificial vector field and environmental heuristics, Int J Adv Robot Syst 17 (2020) 1–18, https://doi.org/10.1177/1729881420969076.
[136] W.L. Zhang, L. Shan, L. Chang, Y.W. Dai, SVF-RRT*: a stream-based VF-RRT* for USVs path planning considering ocean currents, IEEE Robot Autom Lett 8 (2023) 2413–2420, https://doi.org/10.1109/LRA.2023.3245409.
[137] J.G. Kang, J.W. Jung, Post triangular rewiring method for shorter RRT robot path planning, Int J FUZZY Log Intell Syst 21 (2021) 213–221, https://doi.org/10.5391/IJFIS.2021.21.3.213.
[138] J.-G. Kang, Y.-S. Choi, J.-W. Jung, A method of enhancing rapidly-exploring random tree robot path planning using midpoint interpolation, Appl. Sci. 11 (2021).
[139] T.W. Kang, J.G. Kang, J.W. Jung, A bidirectional interpolation method for post-processing in sampling-based robot path planning, SENSORS 21 (2021), https://doi.org/10.3390/s21217425.
[140] Z.Y. Zhou, J. Zhao, Z.Q. Zhang, X.H. Li, Motion planning method of redundant dual-chain manipulator with multiple constraints, J Intell Robot Syst 108 (2023), https://doi.org/10.1007/s10846-023-01921-3.
[141] B. Benno van den, B. Brito, M. Alirezaei, J. Alonso-Mora, Curvature aware motion planning with closed-loop rapidly-exploring random trees, 2021 IEEE Intelligent Vehicles Symposium, in: (IV), 2021, pp. 1024–1030.
[142] J.N. Wang, J.J. Li, J.L. Yang, X.Z. Meng, T.J. Fu, Automatic parking trajectory planning based on random sampling and nonlinear optimization, J FRANKLIN INSTITUTE-ENGINEERING Appl Math 360 (2023) 9579–9601, https://doi.org/10.1016/j.jfranklin.2023.06.037.
[143] D. Wang, S.T. Zheng, Y.X. Ren, D.J. Du, Path planning based on the improved RRT* algorithm for the mining truck, C Mater Contin 71 (2022) 3571–3587, https://doi.org/10.32604/cmc.2022.022183.
[144] S. Mao, P. Yang, D. Gao, C. Bao, Z. Wang, A motion planning method for unmanned surface vehicle based on improved RRT algorithm, J. Mar. Sci. Eng. 11 (2023), https://doi.org/10.3390/jmse11040687.

[145] J. Wang, Y. Luo, X. Tan, Path planning for automatic guided vehicles (AGVs) fusing MH-RRT with improved TEB, Actuators 10 (2021) 1–19, https://doi.org/10.3390/act10120314.
[146] Q.Y. Gu, R. Zhen, J.L. Liu, C. Li, An improved RRT algorithm based on prior AIS information and DP compression for ship path planning, Ocean Eng 279 (2023), https://doi.org/10.1016/j.oceaneng.2023.114595.
[147] Z.Y. Jia, P. Lin, J.L. Liu, L.Y. Liang, Online cooperative path planning for multi-quadrotors in an unknown dynamic environment, Proc Inst Mech Eng PART G-JOURNAL Aerosp Eng 236 (2022) 567–582, https://doi.org/10.1177/09544100211016615.
[148] G. Vailland, V. Gouranton, M. Babel, Cubic bézier local path planner for non-holonomic feasible and comfortable path generation, in: Proc - IEEE Int Conf Robot Autom 2021-May, 2021, pp. 7894–7900, https://doi.org/10.1109/ICRA48506.2021.9560854.
[149] Y. Sun, C. Zhang, C. Liu, Collision-free and dynamically feasible trajectory planning for omnidirectional mobile robots using a novel B-spline based rapidly exploring random tree, Int J Adv Robot Syst 18 (2021) 1–16, https://doi.org/10.1177/17298814211016609.
[150] M. Kim, J. Ahn, J. Park, TargetTree-RRT*: continuous-curvature path planning algorithm for autonomous parking in complex environments, IEEE Trans. Autom. Sci. Eng. (2022), https://doi.org/10.1109/TASE.2022.3225821.
[151] Y. Guo, X. Liu, Q. Jia, X. Liu, W. Zhang, HPO-RRT*: a sampling-based algorithm for UAV real-time path planning in a dynamic environment, Complex Intell Syst 9 (2023) 7133–7153, https://doi.org/10.1007/s40747-023-01115-2.
[152] C. Zhao, Y. Zhu, Y. Du, F. Liao, C.-Y. Chan, A novel direct trajectory planning approach based on generative adversarial networks and rapidly-exploring random tree, IEEE Trans Intell Transp Syst 23 (2022) 17910–17921, https://doi.org/10.1109/TITS.2022.3164391.
[153] R. Reyes, I. Becerra, R. Murrieta-Cid, S. Hutchinson, Visual-RRT: integrating IBVS as a steering method in an RRT planner☆, Rob Auton Syst 169 (2023) https://doi.org/10.1016/j.robot.2023.104525.
[154] S. Ramasamy, K. Eriksson, B. Perumal, S. Peralippatt, F. Danielsson, Optimized path planning by adaptive RRT* algorithm for constrained environments considering energy, in: 2021 26TH IEEE Int. Conf. Emerg. Technol. Fact. Autom, IEEE, 2021.
[155] A.R. Pedram, J. Stefan, R. Funada, T. Tanaka, Rationally inattentive path-planning via RRT, in: 2021 Am. Control Conf., IEEE, 2021, pp. 3440–3446.
[156] M. Lee, G. Noh, J. Park, D. Lee, Real-time directed rapidly exploring random tree path planning for air collision avoidance, J Aerosp Inf Syst (2022), https://doi.org/10.2514/1.I011016.
[157] L.T. Zhou, N.P. Wu, H. Chen, Q.E. Wu, Y.B. Lu, RRT*-Fuzzy dynamic window approach (RRT*-FDWA) for collision-free path planning, Appl. Sci. 13 (2023), https://doi.org/10.3390/app13095234.
[158] Y. Liu, Y. Zhao, S. Yan, C. Song, F. Li, A sampling-based algorithm with the Metropolis acceptance criterion for robot motion planning, Sensors 22 (2022), https://doi.org/10.3390/s22239203.
[159] S. Safaoui, B.J. Gravell, V. Renganathan, T.H. Summers, Risk-averse RRT* planning with nonlinear steering and tracking controllers for nonlinear robotic systems under uncertainty, in: 2021 IEEE/RSJ Int. Conf. Intell. Robot. Syst., IEEE, 2021, pp. 3681–3688.
[160] C.Y. Tao, H.M. Kim, N. Hovakimyan, RRT guided model predictive path integral method, in: 2023 Am. Control Conf. ACC, IEEE, 2023, pp. 776–781.
[161] C.R. Yuan, C.G. Shuai, W.Q. Zhang, A dynamic multiple-query RRT planning algorithm for manipulator obstacle avoidance, Appl. Sci. 13 (2023), https://doi.org/10.3390/app13063394.
[162] G. Yi, C. Zhou, Y. Cao, H. Hu, Hybrid assembly path planning for complex products by reusing a priori data, Mathematics 9 (2021) 1–13, https://doi.org/10.3390/math9040395.
[163] S. Katiyar, A. Dutta, Dynamic path planning over CG-Space of 10DOF Rover with static and randomly moving obstacles using RRT* rewiring, Robotica 40 (2022) 2610–2629, https://doi.org/10.1017/S0263574721001843.
[164] S. Primatesta, A. Osman, A. Rizzo, MP-RRT$^{\#}$: a model predictive sampling-based motion planning algorithm for unmanned aircraft systems, J Intell Robot Syst 103 (2021), https://doi.org/10.1007/s10846-021-01501-3.
[165] B. Zhao, C.D. Wu, X. Zhao, R.H. Sun, Y. Jiang, Research on hybrid navigation algorithm and multi-objective cooperative planning method for electric inspection robot, Energy Rep. 9 (2023) 805–813, https://doi.org/10.1016/j.egyr.2023.05.204.
[166] C.C. Lee, K.T. Song, Path Re-planning design of a cobot in a dynamic environment based on current obstacle configuration, IEEE Robot Autom Lett 8 (2023) 1183–1190, https://doi.org/10.1109/LRA.2023.3236577.
[167] H. Wang, G. Li, J. Hou, L. Chen, N. Hu, A path planning method for underground intelligent vehicles based on an improved RRT* algorithm, Electron 11, https://doi.org/10.3390/electronics11030294, 2022.
[168] H.X. Xi, Obstacle avoidance trajectory planning of redundant robots based on improved Bi-RRT, Int J Syst Assur Eng Manag 14 (2023) 548–557, https://doi.org/10.1007/s13198-021-01255-z.
[169] X.U. Tong, C. Siwei, W. Dong, W.U. Ti, X.U. Yang, Z. Weigong, A Novel Path Planning Method for Articulated Road Roller Using Support Vector Machine and Longest Accessible Path with Course Correction, vol. 7, 2019, https://doi.org/10.1109/ACCESS.2019.2959346.
[170] D. Belter, J. Wietrzykowski, P. Skrzypczynski, Employing natural terrain semantics in motion planning for a multi-legged robot, J Intell Robot Syst 93 (2019) 723–743, https://doi.org/10.1007/s10846-018-0865-x.
[171] J. Pan, D. Manocha, Fast probabilistic collision checking for sampling-based motion planning using locality-sensitive hashing, Int J Rob Res 35 (2016) 1477–1496, https://doi.org/10.1177/0278364916640908.
[172] L. Janson, B. Ichter, M. Pavone, Deterministic sampling-based motion planning: optimality, complexity, and performance, Int J Rob Res 37 (2018) 46–61, https://doi.org/10.1177/0278364917714338.
[173] C. Peng, O. Donca, G. Castillo, A. Hereid, Safe bipedal path planning via control barrier functions for polynomial shape obstacles estimated using logistic regression, in: Proceedings - IEEE International Conference on Robotics and Automation, IEEE, 2023, pp. 3649–3655.
[174] J. Cano, Y. Yang, B. Bodin, V. Nagarajan, M. O'Boyle, Automatic parameter tuning of motion planning algorithms, in: IEEE International Conference on Intelligent Robots and Systems, 2018, pp. 8103–8109.
[175] P. Baldoni, J. McMahon, E. Plaku, Leveraging neural networks to guide path planning: improving dataset generation and planning efficiency, in: IEEE International Conference on Automation Science and Engineering, IEEE, 2022, pp. 667–674.
[176] H. Nichols, M. Jimenez, Z. Goddard, M. Sparapany, B. Boots, A. Mazumdar, Adversarial sampling-based motion planning, IEEE Robot Autom Lett 7 (2022) 4267–4274, https://doi.org/10.1109/LRA.2022.3148464.
[177] N. Ma, J. Wang, J. Liu, M.Q.H. Meng, Conditional generative adversarial networks for optimal path planning, IEEE Trans Cogn Dev Syst 14 (2022) 662–671, https://doi.org/10.1109/TCDS.2021.3063273.
[178] J.K. Wang, W.Z. Chi, C.M. Li, C.Q. Wang, M.Q.H. Meng, Neural RRT*: learning-based optimal path planning, IEEE Trans. Autom. Sci. Eng. 17 (2020) 1748–1758, https://doi.org/10.1109/TASE.2020.2976560.
[179] A. Chaulwar, Sampling algorithms combination with machine learning for efficient safe trajectory planning, in: International Journal of Machine Learning and Computing, 2021, pp. 1–11.
[180] Z. Ding, J. Liu, W. Chi, J. Wang, G. Chen, L. Sun, PRTIRL based socially adaptive path planning for mobile robots, Int J Soc Robot 15 (2023) 129–142, https://doi.org/10.1007/s12369-022-00924-8.
[181] Z. Shang, Z. Mao, H. Zhang, M. Xu, Collaborative path planning of multiple carrier-based aircraft based on multi-agent reinforcement learning, in: Proc - IEEE Int Conf Mob Data Manag 2022-June, 2022, pp. 512–517, https://doi.org/10.1109/MDM55031.2022.00108.
[182] J. Guo, W. Xia, X. Hu, H. Ma, Feedback RRT* algorithm for UAV path planning in a hostile environment, Comput. Ind. Eng. 174 (2022) 108771, https://doi.org/10.1016/j.cie.2022.108771.
[183] J. Yu, A. Arab, J. Yi, X. Pei, X. Guo, Hierarchical framework integrating rapidly-exploring random tree with deep reinforcement learning for autonomous vehicle, Appl. Intell. 53 (2023) 16473–16486, https://doi.org/10.1007/s10489-022-04358-7.
[184] H.T.L. Chiang, J. Hsu, M. Fiser, L. Tapia, A. Faust, RL-RRT: kinodynamic motion planning via learning reachability estimators from RL policies, IEEE Robot Autom Lett 4 (2019) 4298–4305, https://doi.org/10.1109/LRA.2019.2931199.

[185] M. Shah, N. Patel, FNPG-NH: a reinforcement learning framework for flexible needle path generation with nonholonomic constraints, IEEE Robot Autom Lett 8 (2023) 5854–5861, https://doi.org/10.1109/LRA.2023.3300576.
[186] C. Zhu, T. Yu, Q. Chang, Task-oriented safety field for robot control in human-robot collaborative assembly based on residual learning, Expert Syst. Appl. 238 (2024), https://doi.org/10.1016/j.eswa.2023.121946.
[187] G. Xu, L. Zhang, M. Chen, B. He, Hybrid frontier detection strategy for autonomous exploration in multi-obstacles environment, in: 2021 IEEE Int Conf Robot Biomimetics, ROBIO, vol. 2021, 2021, pp. 1909–1915, https://doi.org/10.1109/ROBIO54168.2021.9739463.
[188] G. Rajendran, V. Uma, B. O'Brien, Unified robot task and motion planning with extended planner using ROS simulator, J KING SAUD Univ Inf Sci 34 (2022) 7468–7481, https://doi.org/10.1016/j.jksuci.2021.07.002.
[189] Q. Chen, L. Qi, Z. Hua, Z. Yang, Y. Sun, A hierarchical autonomous exploration algorithm for large-scale and complex environments with mobile robot, in: ACM Int Conf Proceeding Ser, 2023, pp. 238–243, https://doi.org/10.1145/3598151.3598191.
[190] W. Wang, J. Li, Z. Bai, Z. Wei, J. Peng, Toward optimization of AGV path planning: an RRT*-ACO algorithm, IEEE Access 12 (2024) 18387–18399, https://doi.org/10.1109/ACCESS.2024.3359748.
[191] J. Liu, G.Q. Xiao, F. Wu, X.K. Liao, K.L. Li, AAPP: an accelerative and adaptive path planner for robots on GPU, IEEE Trans. Comput. 72 (2023) 2336–2349, https://doi.org/10.1109/TC.2023.3248274.
[192] A. Linard, I. Torre, E. Bartoli, A. Sleat, I. Leite, J. Tumova, Real-time RRT* with signal temporal logic preferences, in: IEEE Int Conf Intell Robot Syst, 2023, pp. 8621–8627, https://doi.org/10.1109/IROS55552.2023.10341993.
[193] K. Naderi, J. Rajamäki, P. Hämäläinen, RT-RRT*: a real-time path planning algorithm based on RRT*, in: Proceedings of the 8th ACM SIGGRAPH Conference on Motion in Games, Association for Computing Machinery, New York, NY, USA, 2015, pp. 113–118.
[194] J. Silveira, K. Cabral, S. Givigi, J.A. Marshall, Real-time fast marching tree for mobile robot motion planning in dynamic environments, in: Proc - IEEE Int Conf Robot Autom 2023-May, 2023, pp. 7837–7843, https://doi.org/10.1109/ICRA48891.2023.10160595.
[195] J. Yu, C. Chen, A. Arab, J. Yi, X. Pei, X. Guo, RDT-RRT: real-time double-tree rapidly-exploring random tree path planning for autonomous vehicles, Expert Syst. Appl. 240 (2024) 122510, https://doi.org/10.1016/j.eswa.2023.122510.
[196] J. Wang, Y. Luo, X. Tan, Path planning for automatic guided vehicles (AGVs) fusing MH-RRT with improved TEB, Actuators 10 (2021), https://doi.org/10.3390/act10120314.
[197] X. Zhou, X.W. Wang, X.S. Gu, An approach for solving the three-objective arc welding robot path planning problem, Eng. Optim. 55 (2023) 650–667, https://doi.org/10.1080/0305215X.2021.2021196.
[198] X.W. Wang, J.B. Wei, X. Zhou, Z.L. Xia, X.S. Gu, Dual-objective collision-free path optimization of arc welding robot, IEEE Robot Autom Lett 6 (2021) 6353–6360, https://doi.org/10.1109/LRA.2021.3092267.
[199] J. Shu, W. Li, Y. Gao, Collision-free trajectory planning for robotic assembly of lightweight structures, Autom Constr 142 (2022) 104520, https://doi.org/10.1016/j.autcon.2022.104520.
[200] A. Ali, J.Y. Lee, Integrated motion planning for assembly task with part manipulation using re-grasping, Appl. Sci. 10 (2020), https://doi.org/10.3390/app10030749.
[201] Q. Bi, X. Zhang, J. Wen, Z. Pan, S. Zhang, R. Wang, J. Yuan, CURE: a hierarchical framework for multi-robot autonomous exploration inspired by centroids of unknown regions, IEEE Trans. Autom. Sci. Eng. PP (2023) 1–14, https://doi.org/10.1109/TASE.2023.3285300.
[202] X. Li, B.Y. Li, Z. Pan, Y. Wang, Design of fire guidance and rescue system based on ROS platform, in: ITOEC 2023 - IEEE 7th Inf Technol Mechatronics Eng Conf, 2023, pp. 601–605, https://doi.org/10.1109/ITOEC57671.2023.10291767.
[203] N. Pérez-Higueras, A. Jardón, A. Rodríguez, C. Balaguer, 3D exploration and navigation with optimal-RRT planners for ground robots in indoor incidents, SENSORS 20 (2020), https://doi.org/10.3390/s20010220.
[204] J. Gui, T. Yu, B. Deng, X. Zhu, W. Yao, Decentralized multi-UAV cooperative exploration using dynamic centroid-based area partition, Drones 7 (2023) 1–15, https://doi.org/10.3390/drones7060337.
[205] V. Tuck, Y.V. Pant, S.A. Seshia, S.S. Sastry, DEC-LOS-RRT: decentralized path planning for multi-robot systems with line-of-sight constrained communication, in: CCTA 2021 - 5th IEEE Conf Control Technol Appl, 2021, pp. 103–110, https://doi.org/10.1109/CCTA48906.2021.9659247.
[206] Y.K. Zhang, Z.Y. Ju, H. Zhang, Z. Qi, 3-D path planning using improved RRT* algorithm for robot-assisted flexible needle insertion in multilayer tissues, IEEE Can J Electr Comput Eng 45 (2022) 50–62, https://doi.org/10.1109/ICJECE.2021.3120324.
[207] J. Li, W. Yuan, X. Gao, Y. Liao, T. Yu, X. He, H. Liu, Design and motion planning of a dexterous flexible cranial puncture robot, in: 2022 IEEE Int Conf Robot Biomimetics, ROBIO, vol. 2022, 2022, pp. 1586–1590, https://doi.org/10.1109/ROBIO55434.2022.10011954.
[208] N. Preda, A. Manurung, O. Lambercy, R. Gassert, M. Bonfè, Motion planning for a multi-arm surgical robot using both sampling-based algorithms and motion primitives, in: IEEE Int Conf Intell Robot Syst 2015-Decem, 2015, pp. 1422–1427, https://doi.org/10.1109/IROS.2015.7353554.
[209] J. Chen, J. Yan, Y. Qiu, H. Fang, J. Chen, S.S. Cheng, A cross-entropy motion planning framework for hybrid continuum robots, IEEE Robot Autom Lett 8 (2023) 8200–8207, https://doi.org/10.1109/LRA.2023.3325777.
[210] T. Song, B. Pan, G. Niu, Y. Fu, A novel closed-form solutions method based on the product of exponential model for the minimally invasive surgical robot, Ind Rob 49 (2022) 934–946, https://doi.org/10.1108/IR-10-2021-0244.
[211] Q.G. Fan, G.M. Cui, Z.Q. Zhao, J. Shen, Obstacle avoidance for microrobots in simulated vascular environment based on combined path planning, IEEE Robot Autom Lett 7 (2022) 9801–9808, https://doi.org/10.1109/LRA.2022.3191540.
[212] J. Guo, Y. Sun, S. Guo, A training system for vascular interventional surgeons based on local path planning, in: 2021 IEEE Int Conf Mechatronics Autom ICMA, vol. 2021, 2021, pp. 1328–1333, https://doi.org/10.1109/ICMA52036.2021.9512808.
[213] T. Rybus, M. Wojtunik, F.L. Basmadji, Optimal collision-free path planning of a free-floating space robot using spline-based trajectories, Acta Astronaut. 190 (2022) 395–408, https://doi.org/10.1016/j.actaastro.2021.10.012.
[214] T. Rybus, J. Prokopczuk, M. Wojtunik, K. Aleksiejuk, J. Musiał, Application of Bidirectional Rapidly Exploring Random Trees (BiRRT) Algorithm for Collision-free Trajectory Planning of Free-Floating Space Manipulator, 2022.
[215] M. Yu, J. Luo, M. Wang, D. Gao, Spline-RRT*: coordinated motion planning of dual-arm space robot, IFAC-PapersOnLine 53 (2020) 9820–9825, https://doi.org/10.1016/j.ifacol.2020.12.2685.
[216] T. Rybus, K. Seweryn, Application of Rapidly-exploring Random Trees (RRT) algorithm for trajectory planning of free-floating space manipulator, in: 2015 10th Int Work Robot Motion Control RoMoCo, vol. 2015, 2015, pp. 91–96, https://doi.org/10.1109/RoMoCo.2015.7219719.
[217] J.R.S. Benevides, V. Grassi, Autonomous path planning of free-floating manipulators using RRT-based algorithms, in: Proc - 12th LARS Lat Am Robot Symp 3rd SBR Brazilian Robot Symp LARS-SBR 2015 - Part Robot Conf 2015, 2016, pp. 139–144, https://doi.org/10.1109/LARS-SBR.2015.47.
[218] H. Zhang, Z. Wang, A new motion planning algoritm for free-floating space robot, in: Proc 35th Chinese Control Decis Conf CCDC, vol. 2023, 2023, pp. 3275–3280, https://doi.org/10.1109/CCDC58219.2023.10326885.
[219] H. Zhang, Y. Tang, Z.Z. B, A Novel Motion Planning Algorithm Based on RRT-Connect and Bidirectional Approach for Free-Floating Space Robot, Springer Nature, Singapore, 2023.
[220] Y. Chen, Z. He, S. Li, Real-time trajectory planning for free-floating space robot in close range using continuous thrust, J Phys Conf Ser 1074 (2018) 1–8, https://doi.org/10.1088/1742-6596/1074/1/012015.
[221] W.G. Serrantola, V. Grassi, Trajectory planning for a dual-arm planar free-floating manipulator using RRTControl, in: 2019 19th Int Conf Adv Robot ICAR, vol. 2019, 2019, pp. 394–399, https://doi.org/10.1109/ICAR46387.2019.8981596.
[222] Z. Shao, Y Bin Moon, Y.C. Yang, M. Kumral, Underground haulage network design using HDBSCAN and RRT algorithms built on Dubins path, Mining, Metall Explor 40 (2023) 773–786, https://doi.org/10.1007/s42461-023-00777-3.
[223] X. Zhang, M. Huang, M. Lei, H. Tian, X. Chen, C. Tian, Improved rapid-expanding-random-tree-based trajectory planning on drill ARM of anchor drilling robots, Machines 11 (2023) 858.
[224] W. Zhao, X. Wang, Y. Liu, Path planning for 5-Axis CMM inspection considering path reuse, Machines 10 (2022), https://doi.org/10.3390/machines10110973.

[225] Z. Jianning, L. Jiaxin, D. Songyi, G. Rui, Beetle Antennae Search guided RRT* for path planning of GIS inspection and maintenance robot, in: Proc - 2021 6th Int Conf Autom Control Robot Eng CACRE, vol. 2021, 2021, pp. 102–107, https://doi.org/10.1109/CACRE52464.2021.9501393.
[226] J.K. Huang, Y. Tan, D. Lee, V.R. Desaraju, J.W. Grizzle, Informable multi-objective and multi-directional RRT*System for robot path planning, in: Proc - IEEE Int Conf Robot Autom 2023-May, 2023, pp. 5666–5673, https://doi.org/10.1109/ICRA48891.2023.10160838.
[227] F. Wang, Y. Gao, Z. Chen, X. Gong, D. Zhu, W. Cong, A path planning algorithm of inspection robots for solar power plants based on improved RRT, ELECTRONICS 12 (2023), https://doi.org/10.3390/electronics12214455.
[228] K.L. Hsu, Obstacle avoidance path scheme of snake robot based on bidirectional fast expanding random tree algorithm, J. King Saud Univ. Sci. 34 (2022), https://doi.org/10.1016/j.jksus.2022.101975.
[229] X.H. Yang, X. Ye, J.N. Cao, R. Yan, X.J. Guo, J.S. Huang, High-altitude inspection technology of substation based on fusion of unmanned aerial vehicle and multiple sensors, SENSORS Mater 34 (2022) 3191–3212, https://doi.org/10.18494/SAM3933.
[230] Z. Fang, Optimized UAV navigation overcoming LoS obstructions for maximized power grid tower inspections in mountainous terrains, in: 2023 IEEE Int Conf Robot Biomimetics, ROBIO, vol. 2023, 2023, pp. 1–6, https://doi.org/10.1109/ROBIO58561.2023.10354672.
[231] G. Tang, C. Wang, Z. Zhang, S. Men, UAV path planning for container terminal yard inspection in a port environment, J. Mar. Sci. Eng. 12 (2024), https://doi.org/10.3390/jmse12010128.
[232] G. Tang, P. Liu, Z. Hou, C. Claramunt, P. Zhou, Motion planning of UAV for port inspection based on extended RRT* algorithm, J. Mar. Sci. Eng. 11 (2023), https://doi.org/10.3390/jmse11040702.
[233] Y. Zhou, C. Cai, J. Yao, Y. Zou, Y. Yang, Design of autonomous navigation system for quadrotor in subway tunnel, in: IEEE Int Conf Control Autom ICCA 2020-Octob, 2020, pp. 187–192, https://doi.org/10.1109/ICCA51439.2020.9264455.
[234] GGR.d. Castro, G.S. Berger, A. Cantieri, M. Teixeira, J. Lima, A.I. Pereira, M.F. Pinto, Adaptive path planning for fusing rapidly exploring random trees and deep reinforcement learning in an agriculture dynamic environment UAVs, Agric. For. 13 (2023), https://doi.org/10.3390/agriculture13020354.
[235] A. Suarez, A. Caballero, A. Garofano, P.J. Sanchez-Cuevas, G. Heredia, A. Ollero, Aerial manipulator with rolling base for inspection of pipe arrays, IEEE Access 8 (2020) 162516–162532, https://doi.org/10.1109/ACCESS.2020.3021126.
[236] T. Liu, W. Li, X. Sun, W. Wang, Y. Fang, X. Zhu, L. Zhong, G. Liu, J. Zhao, Mechanical design and path planning of A robotic arm for GIS pipeline inspection, in: 2020 IEEE Int Conf Mechatronics Autom ICMA, vol. 2020, 2020, pp. 1371–1376, https://doi.org/10.1109/ICMA49215.2020.9233665.
[237] J. Zhang, Y. An, J. Cao, S. Ouyang, L. Wang, UAV trajectory planning for complex open storage environments based on an improved RRT algorithm, IEEE Access 11 (2023) 23189–23204, https://doi.org/10.1109/ACCESS.2023.3252018.
[238] L. Zacchini, A. Ridolfi, B. Allotta, Receding-horizon sampling-based sensor-driven coverage planning strategy for AUV seabed inspections, in: 2020 IEEE/OES Auton Underw Veh Symp AUV, vol. 2020, 2020, pp. 1–6, https://doi.org/10.1109/AUV50043.2020.9267903.
[239] Y.H. Wu, J.C. Liu, Y.G. Wei, D. An, Y.H. Duan, W.S. Li, B.K. Li, Y.R. Chen, Q. Wei, Intelligent control method of underwater inspection robot in netcage, Aquac Res 53 (2022) 1928–1938, https://doi.org/10.1111/are.15721.
[240] M. Henderson, T.D. Ngo, RRT-SMP: socially-encoded motion primitives for sampling-based path planning, in: 2021 30TH IEEE Int. Conf. Robot Hum. Interact. Commun., IEEE, 2021, pp. 330–336.
[241] K. Merckaert, B. Convens, M.M. Nicotra, B. Vanderborght, Real-time constraint-based planning and control of robotic manipulators for safe human–robot collaboration, Robot Comput Integr Manuf 87 (2024) 102711, https://doi.org/10.1016/j.rcim.2023.102711.
[242] D. Armstrong, A. Jonasson, AM-RRT*: informed sampling-based planning with assisting metric, in: 2021 IEEE Int. Conf. Robot. Autom. (ICRA 2021), IEEE, 2021, pp. 10093–10099.
[243] S. Ramasamy, K.M. Eriksson, F. Danielsson, M. Ericsson, Sampling-based path planning algorithm for a plug & produce environment, Appl. Sci. 13 (2023) 12114, https://doi.org/10.3390/app132212114.
[244] W. Liu, S.W. Kim, Z.J. Chong, X.T. Shen, M.H. Ang, Motion planning using cooperative perception on urban road, in: IEEE Conference on Robotics, Automation and Mechatronics, RAM - Proceedings, IEEE, 2013, pp. 130–137.
[245] T. Xu, D. Wang, W. Zhang, Multi-roller trajectory planning under the dynamic obstacle in pavement construction, in: 2021 7th Int Conf Control Autom Robot ICCAR, vol. 2021, 2021, pp. 36–40, https://doi.org/10.1109/ICCAR52225.2021.9463473.