



南京航空航天大学

SE(2)机器人的运动规划

姓 名 张 康

学 号 BX2207004

学 院 民航学院

专 业 交通运输工程

班 级 机器人智能控制班

二〇二二年十一月

第 1 页

概述：运动规划是机器人自主完成任务的重要模块。作业基于 matlab 软件编程实现了一个二维平面的机器人运动规划框架。该框架十分灵活：可自由的调整障碍物的类别、数量及占用面积大小；使用的基于采样的快速扩展随机树(RRT)路径规划算法具有强大的全局搜索能力；利用多项式来平滑轨迹，使得轨迹的求解更稳定、高效。

1 地图生成器：

生成随机凸多边形障碍的方法见参考文献[1]，可视化流程见图 1。

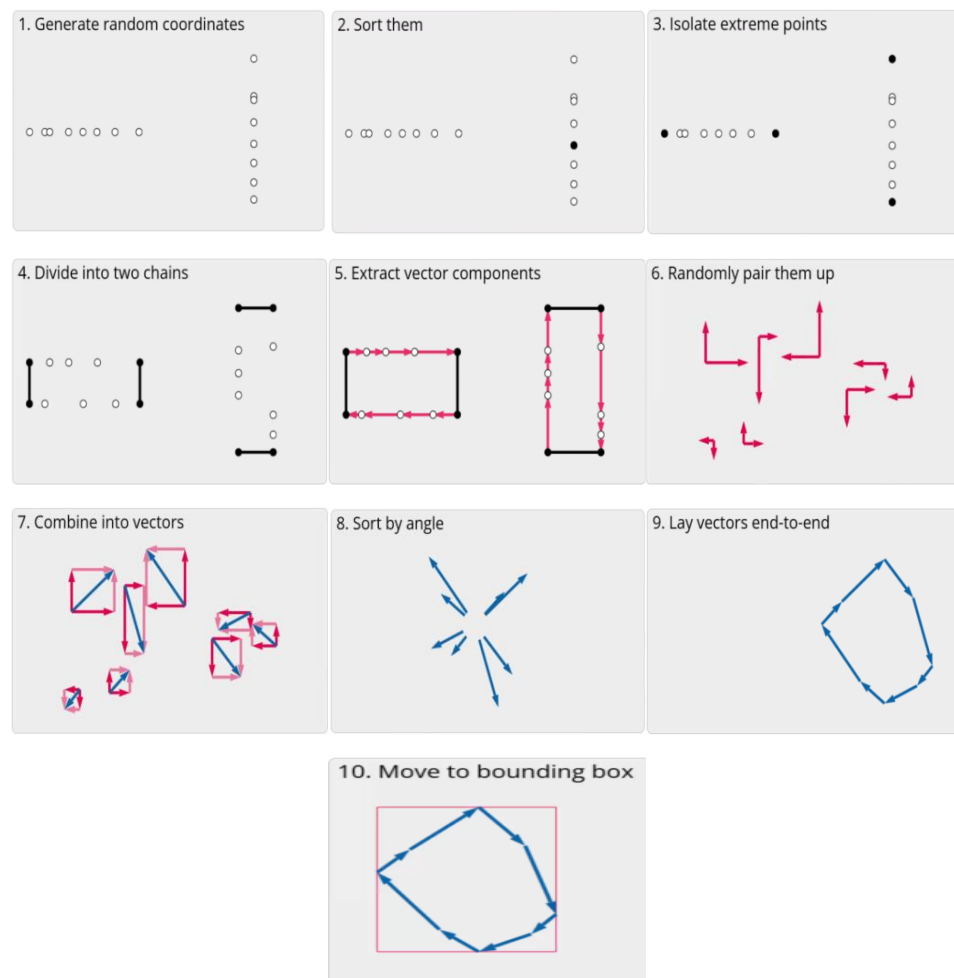


图1 随机凸多边形生成的流程图

判断点是否位于凸多边形障碍物内的条件：For all $\vec{edge} \times (\vec{p} - \vec{vertex}_i) > 0$
then is **in** else **out**.

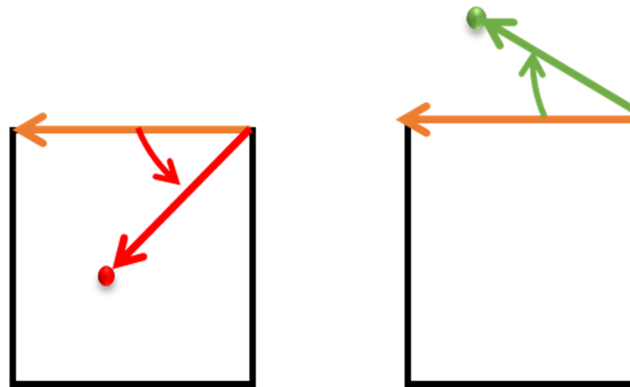
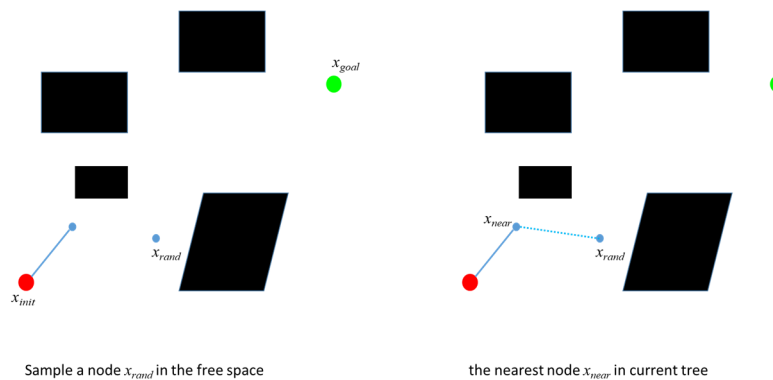


图2 判断示意图

其他细节详见代码 world 包。

2 路径规划：

使用基于采样的 RRT 家族算法^[2-4]作为路径规划包主要内容，而路径规划包的任务是负责生成一系列安全的 waypoints 作为前端搜索供给后端的轨迹优化包。细节见 plan 包。



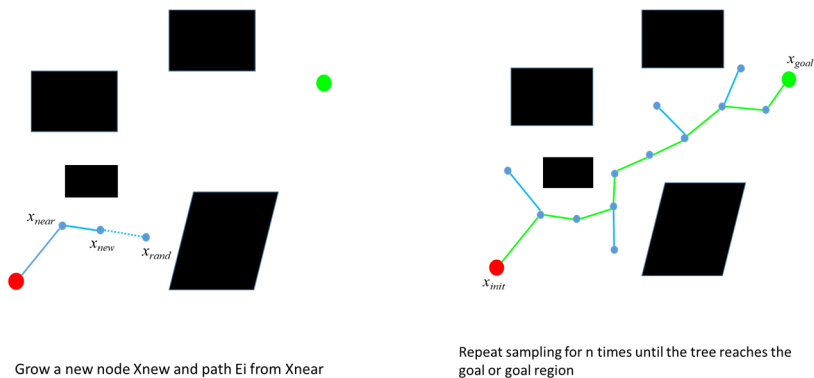


图3 RRT 算法过程

RRT*算法针对早期的 RRT 算法最优性的缺失，通过在每次扩展节点后的近邻优化操作，确保了着搜索树可以渐进地向着最优趋势生长，具有渐进最优性。Informed RRT*算法基于重要性采样思想，利用原 RRT*找到的初始解来构建一个超椭球子集的约束采样空间，椭球子集随着解不断逼近最优解而逐渐缩小，这提升了收敛速度，并很大程度上改善了采样类算法在狭窄通道场景下表现不佳的缺点。

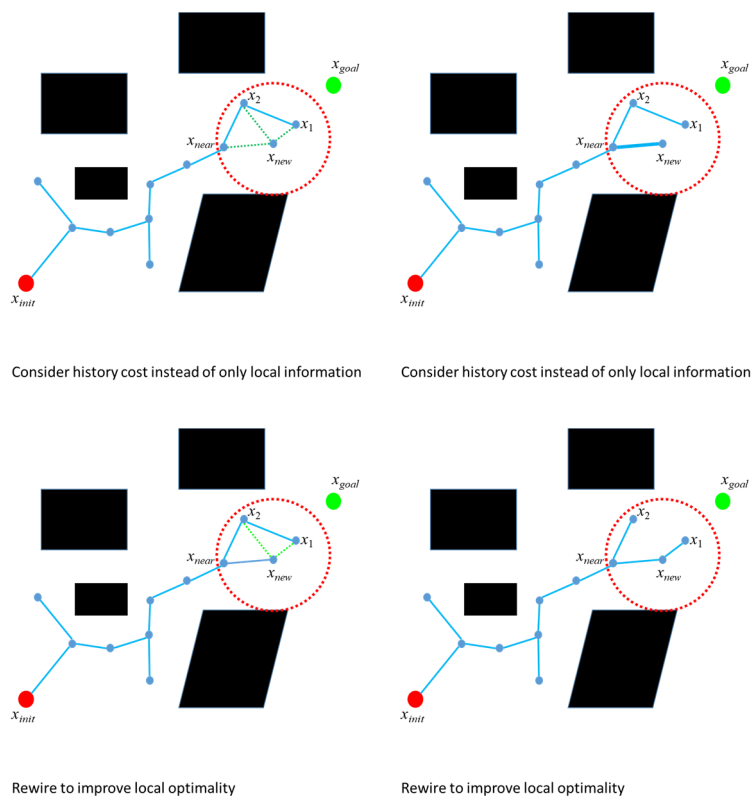


图4 RRT*的近邻优化过程

算法 1 Informed RRT*。

- 1) $V \leftarrow \{x_{init}\}, E, x^* \leftarrow \emptyset, T \leftarrow (V, E) \quad c_{best} = 0;$
- 2) for $iteration = 0$ to N do
- 3) $x_{rand} \leftarrow \text{HeuristicSample}(x^*, c_{best});$
- 4) $x_{nearest} \leftarrow \text{Nearest}(T, x_{rand});$
- 5) $x_{new} \leftarrow \text{Steer}(x_{nearest}, x_{rand});$
- 6) if $\text{CollisionCheck}(x_{new}, x_{nearest})$ then
- 7) $T \leftarrow \text{InsertNode}(x_{new}, T);$
- 8) $x_{nearest} \leftarrow \text{Nearest}(T, x_{rand});$
- 9) $x_{parent} \leftarrow \text{Chooseparent}(T, V_{near}, x_{new});$
- 10) $T \leftarrow \text{Rewire}(T, V_{near}, x_{parent});$
- 11) end
- 12) if $\text{PathFound}(T)$ then
- 13) $x^* \leftarrow \text{FindBestPath}(T, x_{goal});$
- 14) $c_{best} \leftarrow \text{Cost}(x^*);$
- 15) end
- 16) end
- 17) return T

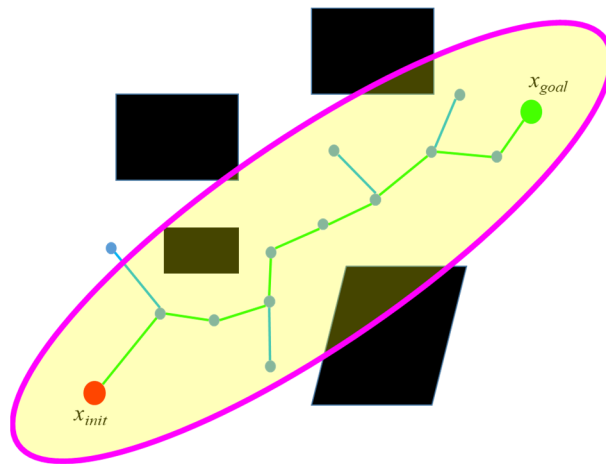


图5 Informed RRT*生成超椭球子集过程

通常情况下，由前端找到的路径不能被直接执行，因为它可能是不连续的或包含不自然的转向。因此，需要进一步对路径进行优化，生成一条安全、连续、可行的轨迹。

3 轨迹优化

大多数路径搜索算法构造的是不含时间信息的几何轨迹。为了满足运动学、动力学的可行性，需要对初始几何轨迹进行时间上的参数化，这是轨迹生成的主要目标。一般来说，轨迹生成问题是在满足安全性和动力学可行性约束的情况下最小化总控制成本等目标函数，基本等价于一个优化问题，而将前端搜索的低维空间路径点信息作为输入，可以很大程度上减少后端轨迹优化的求解压力，这也是当前运动规划方法中比较常见的范式。

较有代表性的工作是文献[5]中的 **minimum-snap** 轨迹规划方法，它利用微分平坦变换通过高阶多项式函数表示具有时空信息的轨迹。通过计算每一段轨迹的多项式系数，使总体的多项式轨迹满足首末状态、中间点约束，保证在中间点轨迹的前后段左右极限的高阶导数相同，使分段多项式光滑连续，同时最小化代价函数。

多项式轨迹 $p(t)$:

$$p(t) = \mathbf{t} \cdot \mathbf{c}; \mathbf{t} = [1 \quad t \quad t^2 \dots t^{N-1}]; \mathbf{c} = [c_0 \dots c_{N-1}]^T$$

优化目标:

$$J(T) = \int_0^T c_0 P(t)^2 + c_1 P'(t)^2 + c_2 P''(t)^2 + \dots + c_N P^{(N)}(t)^2 dt = \mathbf{p}^T Q(T) \mathbf{p}$$

$$J_{total} = \begin{bmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_M \end{bmatrix}^T \begin{bmatrix} Q_1(T_1) & & \\ & \ddots & \\ & & Q_M(T_M) \end{bmatrix} \begin{bmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_M \end{bmatrix}^T$$

连续性约束:

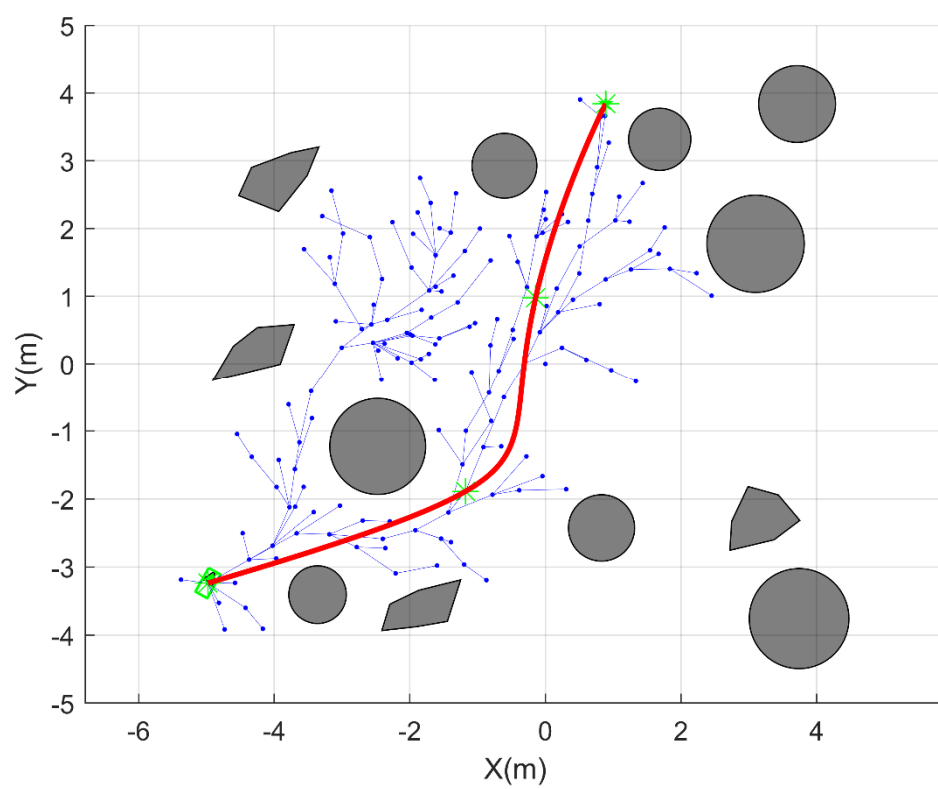
$$A_i \mathbf{p}_i = \mathbf{d}_i, \quad A_i = \begin{bmatrix} A_0 \\ A_T \end{bmatrix}_i, \quad \mathbf{d}_i = \begin{bmatrix} d_0 \\ d_T \end{bmatrix}_i$$

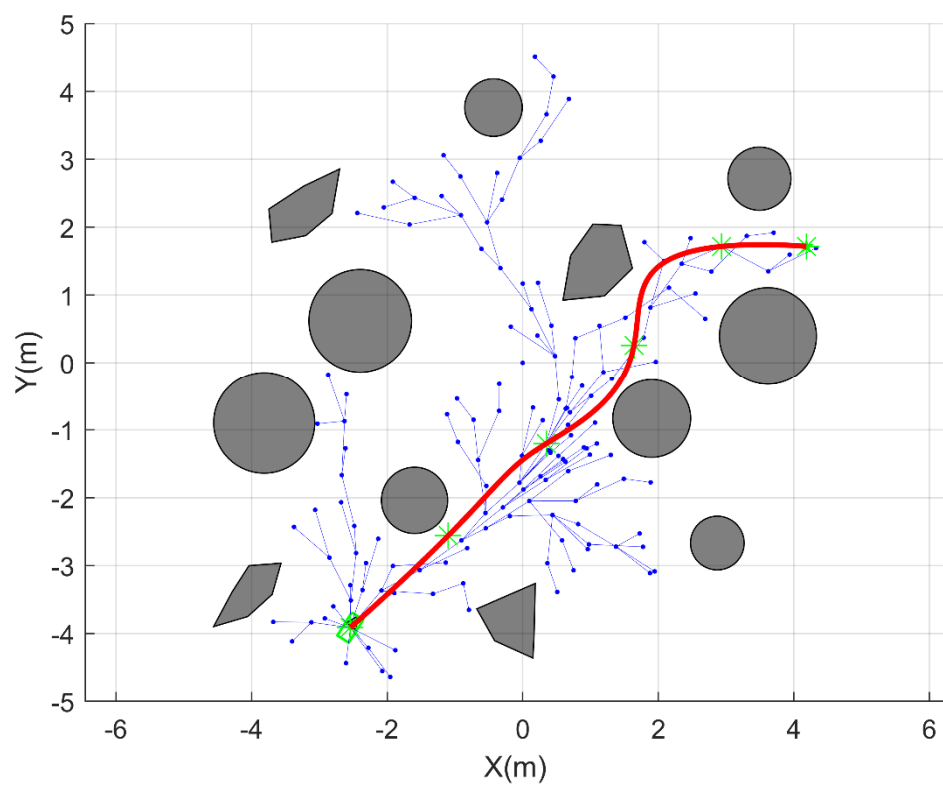
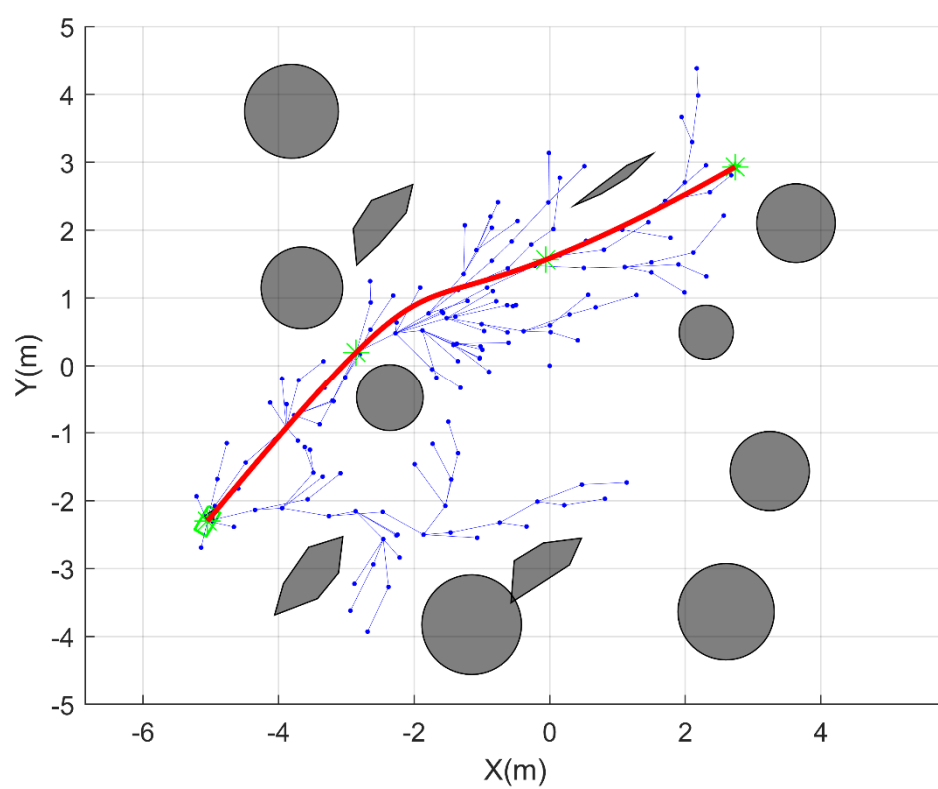
$$A_{T,i} \mathbf{p}_i = A_{0,i+1} \mathbf{p}_{i+1}$$

通过以上可得到一个一般形式的 (QP) 二次规划问题。细节见 `minimumsnap` 包。

4 结果

随机三次的仿真结果：





仿真运行时，主函数可选择 `mainsim.m` 或者 `image` 包下的 `fig1.m`，上来就报错的话，就多运行几次，鼠标左键设定位置方向，右键退出仿真。

参考文献:

- [1] Valtr P. Probability thatn random points are in convex position[J]. Discrete & Computational Geometry, 1995, 13(3-4): 637-643.
- [2] LAVALLE S M, KUFFNER J J, Jr. Rapidly-exploring random trees: progress and prospects[M]// DONALD B R, LYNCH K, RUS D. Algorithmic and Computational Robotics: New Directions 2000 WAFR. Natick, MA: A K Peters, 2001: 293-308.
- [3] KARAMAN S, FRAZZOLI E. Sampling-based algorithms for optimal motion planning[J]. The International Journal of Robotics Research, 2011, 30(7): 846-894.
- [4] Gammell J D, Barfoot T D, Srinivasa S S. Informed sampling for asymptotically optimal path planning[J]. IEEE Transactions on Robotics, 2018, 34(4): 966-984.
- [5] Mellinger D, Kumar V. Minimum snap trajectory generation and control for quadrotors[C]//2011 IEEE international conference on robotics and automation. IEEE, 2011: 2520-2525.

代码附录:

<https://github.com/robotzk/2d-planning-frame>