《机器人学导论》课程设计结题汇报

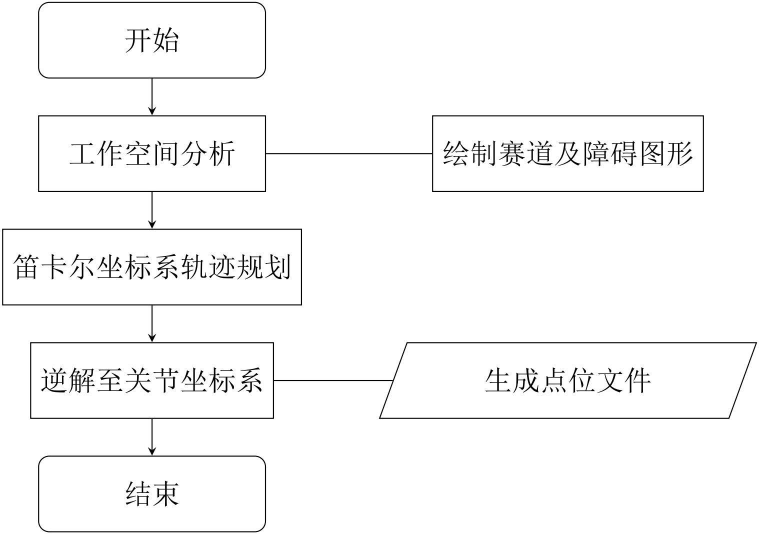
CCcircle, Oliver Wu

July 4, 2024

提纲

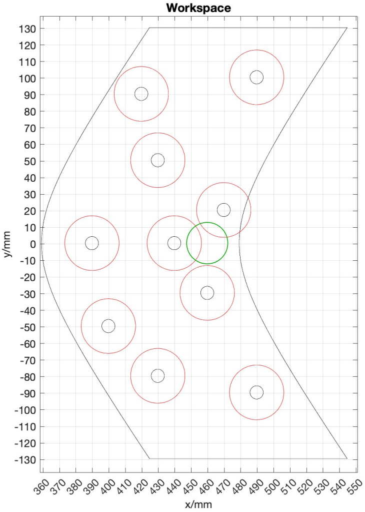
1. 功能模块分解
2. 工作空间分析
3. 笛卡尔坐标系轨迹规划
   - 方法一：LFPB（Linear Function with Parabolic Blends）
   - 方法二：B-样条曲线

功能模块分解

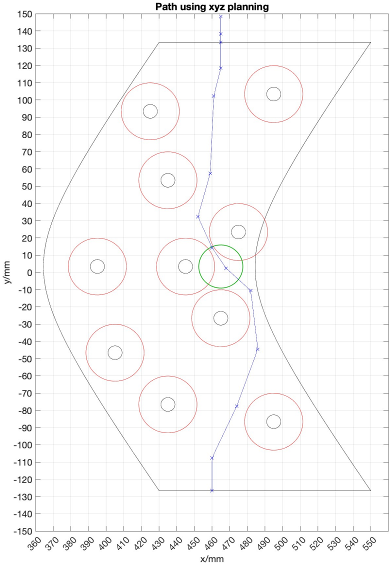


工作空间分析

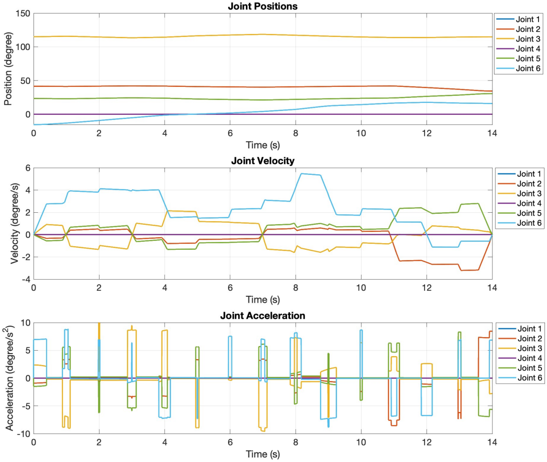
绘制赛道及障碍图形



LFPB（Linear Function with Parabolic Blends）



LFPB逆解结果图示



LFPB规划得出的关节位移、速度和加速度

B-样条曲线由多段四次Bezier曲线组成。四次Bezier曲线有5个控制点P_0, P_1, P_2, P_3和P_4。因为本次规划不需要考虑z坐标，所以曲线设为二维曲线，各控制点均为二维平面上的点（即2维向量）。这些控制点共同决定了曲线方程

$$p(t) = \begin{bmatrix} t^4 & t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} 1 & -4 & 6 & -4 & 1 \\ -4 & 12 & -12 & 4 & 0 \\ 6 & -12 & 6 & 0 & 0 \\ -4 & 4 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \\ P_4 \end{bmatrix}$$

其中$0 \le t \le 1$。对其求导可写出速度方程

$$p'(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} 4 & -16 & 24 & -16 & 4 \\ -12 & 36 & -36 & 12 & 0 \\ 12 & -24 & 12 & 0 & 0 \\ -4 & 4 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \\ P_4 \end{bmatrix}.$$

类似可写出加速度、加加速度关于时间t和控制点的方程：

$$p''(t) = \begin{bmatrix} t^2 & t & 1 \end{bmatrix} \begin{bmatrix} 12 & -48 & 72 & -48 & 12 \\ -24 & 72 & -72 & 24 & 0 \\ 12 & -24 & 12 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \\ P_4 \end{bmatrix}$$

$$p'''(t) = \begin{bmatrix} t & 1 \end{bmatrix} \begin{bmatrix} 24 & -96 & 144 & -96 & 24 \\ -24 & 72 & -72 & 24 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \\ P_4 \end{bmatrix}$$

使用多段Bezier曲线拼接成B-样条曲线，确保各段均连接且连接处速度、加速度连续。

记共有n段轨迹（即有$n+1$个 via point），第i段轨迹方程为$P_i(t)$，则上述约束可表示为

$$p_i(1) = p_{i+1}(0), i = 1, \ldots, n-1$$

$$p_i'(1) = p_{i+1}'(0), i = 1, \ldots, n-1$$

$$p_i''(1) = p_{i+1}''(0), i = 1, \ldots, n-1$$

通过使via point作为每段首个（或末个）控制点，可满足其中第一个约束，这样整个曲线的控制参数有$2 \times (5-2)n = 6n$个；而上述第二和第三个约束共将产生$4(n-1)$个约束方程，可见约束方程数少于曲线的控制参数个数，剩余的自由度可以进行优化配置。

取优化目标函数：

$$\min \sum_{i=1}^{n} \int_{0}^{1} \|p_i'''(t)\|^2 \mathrm{d}t$$

将加加速度方程代入可将其化为

$$\min \sum_{i=1}^{n} \left(\|c_{i4}\|^2 + 4\langle c_{i4}, c_{i5} \rangle + \frac{16}{3}\|c_{i5}\|^2 \right)$$

其中$C_i = MP_i$，$M = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -4 & 4 & 0 & 0 & 0 \\ 6 & -12 & 6 & 0 & 0 \\ -4 & 12 & -12 & 4 & 0 \\ 1 & -4 & 6 & -4 & 1 \end{bmatrix}$，$P_i = \begin{bmatrix} P_{i0} \\ P_{i1} \\ P_{i2} \\ P_{i3} \\ P_{i4} \end{bmatrix}$

在此基础上，增加4个等式约束方程，即初始速度、加速度与末尾速度、加速度均为0：

$$p_1'(0) = p_1''(0) = p_n'(1) = p_n''(1) = 0$$

为防止速度、加速度过大，另增加若干限制的不等式约束方程：

$$|p_{ix}'(t)| \leq v_{\max}, |p_{iy}'(t)| \leq v_{\max}, i = 1, \ldots, n, \forall t \in [0, 1]$$

$$|p_{ix}''(t)| \leq a_{\max}, |p_{iy}''(t)| \leq a_{\max}, i = 1, \ldots, n, \forall t \in [0, 1]$$

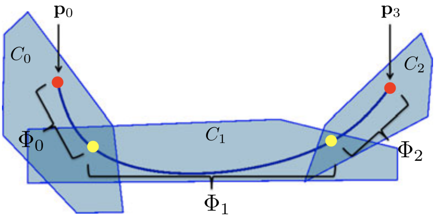
这些约束是连续的，因此，将每段的t在$[0, 1]$区间内按0.01的间隔离散化，以给出一系列确定的线性方程。

若直接衡量轨迹各点至障碍的距离，则将引入非线性约束，计算量大大增加。为简化计算，用凸N边形表示不会碰撞的空间，凸N边形可用一系列线性不等式来描述：

$$A_i p_i(t) \leq b_i, i = 1, \ldots, n, \forall t \in [0, 1]$$

其中各A_i为$N \times 2$矩阵，b_i为N维列向量。

这样，就将每段轨迹p_i约束在对应多边形(A_i, b_i)内。相邻两段的多边形有一定重叠，从中选取via points即可。同样将每段的t在$[0, 1]$区间内进行离散化。



最终，优化问题可表述为：

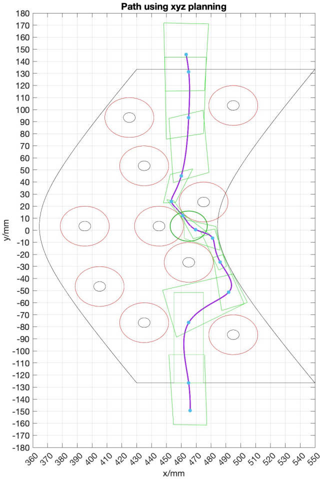
$$\min \sum_{i=1}^{n} \left(\|c_{i4}\|^2 + 4\langle c_{i4}, c_{i5} \rangle + \frac{16}{3}\|c_{i5}\|^2 \right)$$

s.t.

$$p_i'(1) = p_{i+1}'(0), i = 1, \ldots, n-1$$
$$p_i''(1) = p_{i+1}''(0), i = 1, \ldots, n-1$$
$$p_1'(0) = p_1''(0) = p_n'(1) = p_n''(1) = 0$$
$$A_i p_i(t) \le b_i, i = 1, \ldots, n, \forall t \in [0, 1]$$
$$|p_{ix}'(t)| \le v_{\max}, |p_{iy}'(t)| \le v_{\max}, i = 1, \ldots, n, \forall t \in [0, 1]$$
$$|p_{ix}''(t)| \le a_{\max}, |p_{iy}''(t)| \le a_{\max}, i = 1, \ldots, n, \forall t \in [0, 1]$$

这是一个带线性等式和不等式约束的二次规划问题。

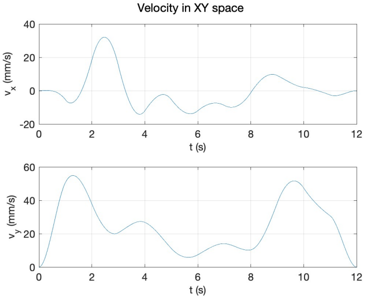
利用MATLAB的fmincon函数解之，得出轨迹：



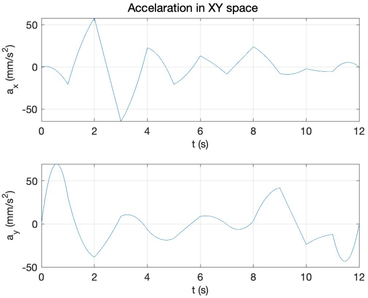
B-样条曲线

求解与结果

利用MATLAB的fmincon函数解之，得出轨迹：

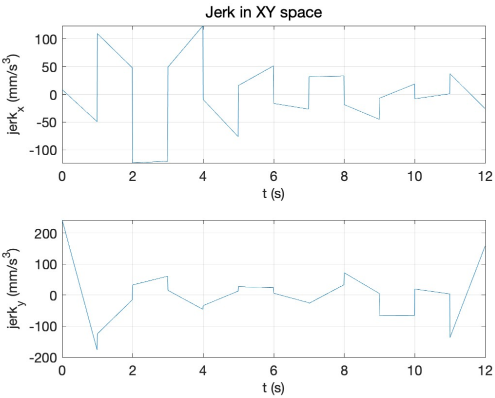


(a) 速度(Velocity)



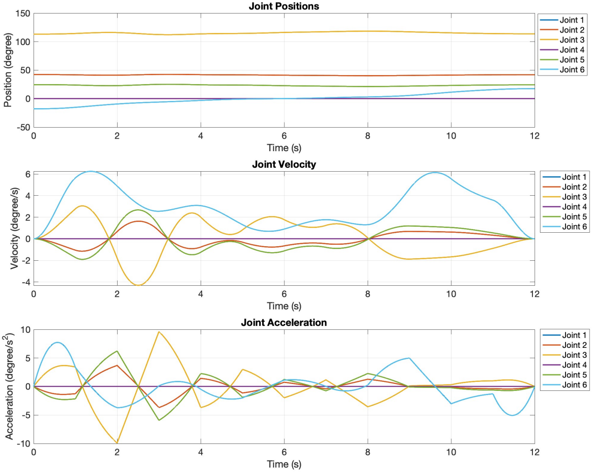
(b) 加速度(Acceleration)

利用MATLAB的fmincon函数解之，得出轨迹：



加加速度(Jerk)

B-样条曲线逆解结果图示



B-样条曲线规划得出的关节位移、速度和加速度

总结与收获

总结：

- 通过工作空间分析，选取了一系列路径点；
- 在笛卡尔坐标系下分别用LFPB和B-样条曲线生成轨迹；
- 将轨迹逆解至关节坐标系，生成了点位文件并能使机器人按规定完成任务。

收获：

- 巩固了正、逆运动学与轨迹规划知识；
- 学习了课外知识：B-样条曲线生成轨迹；
- 提升了利用MATLAB编程和仿真的能力。

参考文献 I

J. Craig.
Introduction to Robotics: Mechanics and Control. 2nd ed.
Prentice Hall, 2005.

KILIÇOĞLU, Şeyda, and Süleyman ŞENYURT
On the matrix representation of Bezier curves and derivatives in E3.
Sigma 41.5 (2023): 992-998.

Liu, Sikang, et al.
Planning dynamically feasible trajectories for quadrotors using safe flight corridors in 3-d complex environments.
IEEE Robotics and Automation Letters 2.3 (2017): 1688-1695.

谢谢！

Thank you!