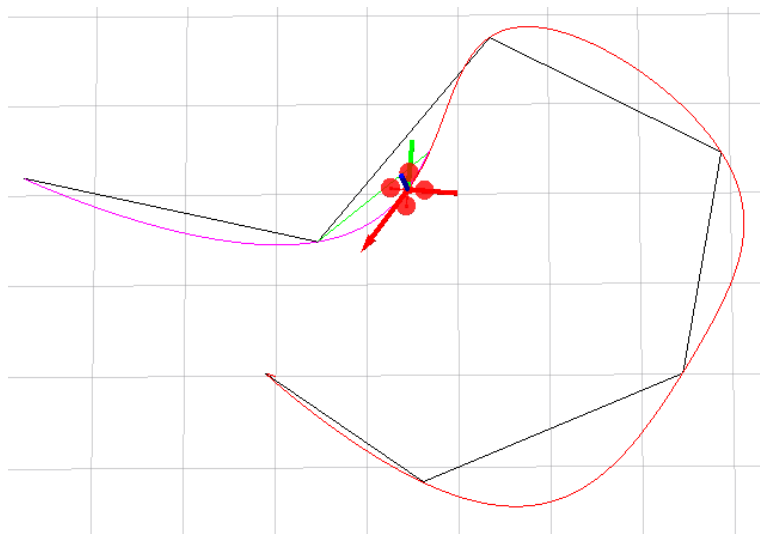


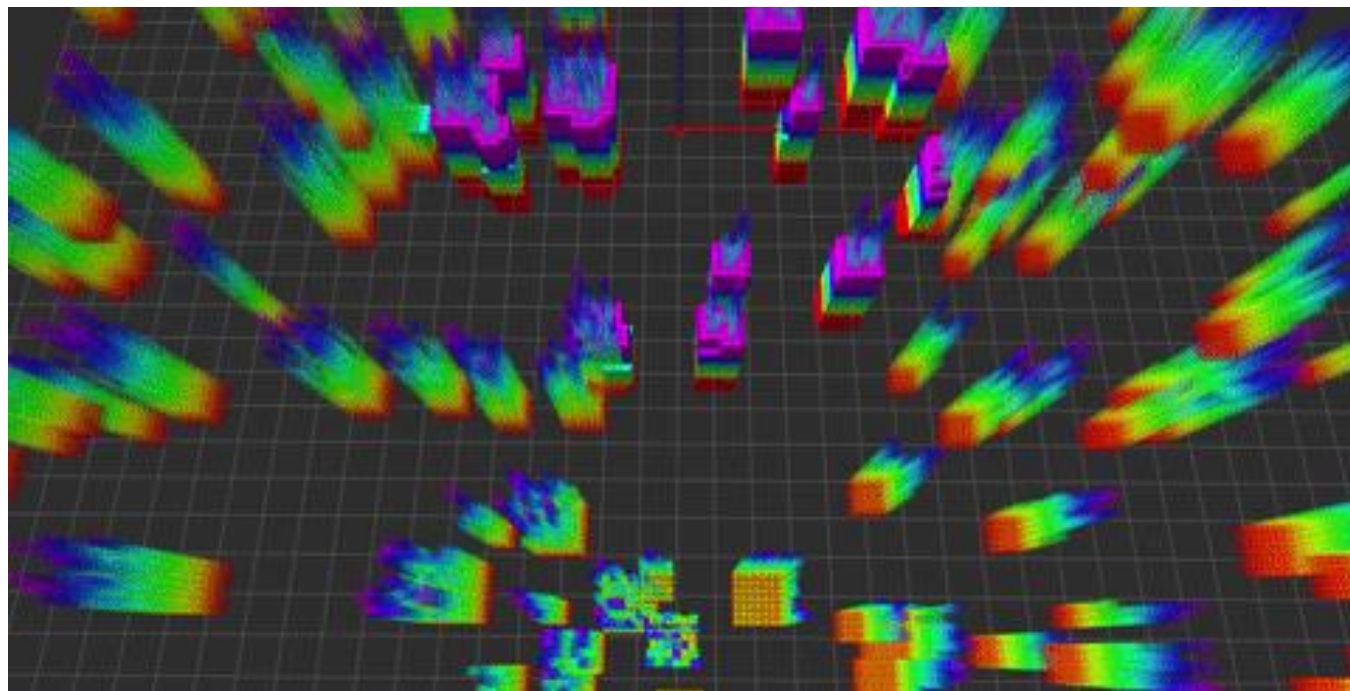
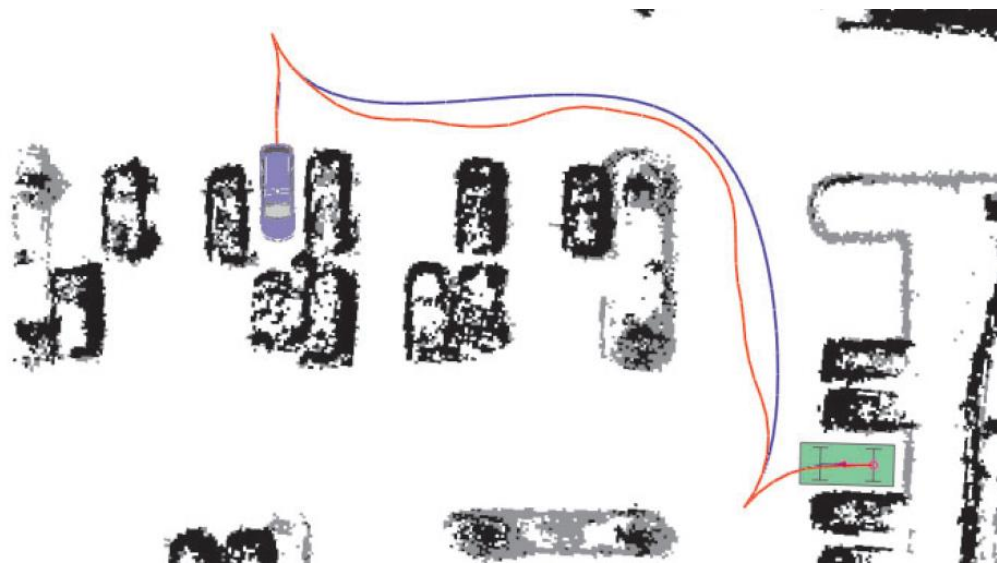


- 机器人的速度和动力学高阶量无法突变;
- 节约机器人运动过程的能量消耗;
- 保证轨迹执行时间的合理性;
- 确保机器人移动过程中的安全;



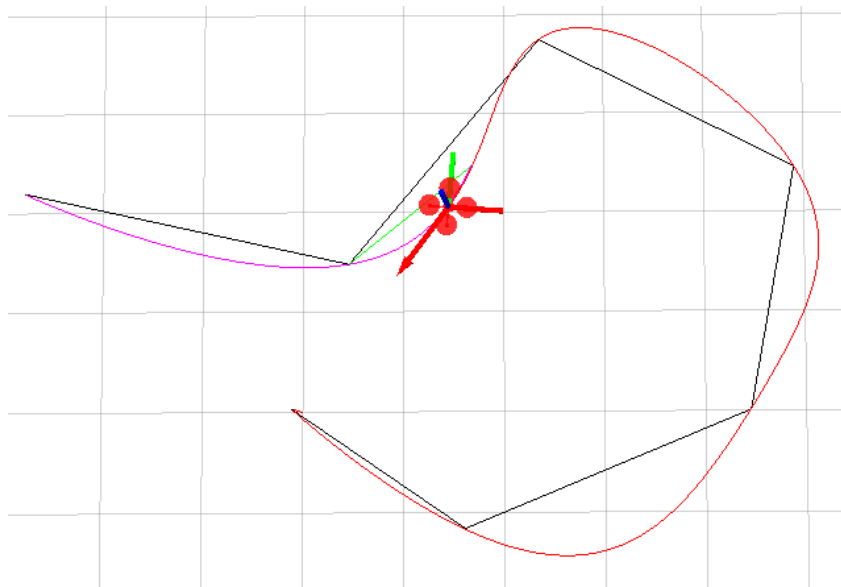


- Q: 我们有前端路径搜索, 为什么需要后端轨迹优化?
- Q: 前端Kinodynamic Path Finding 是可行的, 为什么后端轨迹优化是必要的?





- 边界条件：起始位置（朝向）
- 中间条件：航点位置（朝向）
 - 航点由路径规划得到（A*,RRT*等）
- 光滑性的标准
 - 通常通过最小化“输入”的变化率





四旋翼的状态和输入可以写成四个挑选出来的输出变量及其导数的代数函数

- 能够自动生成轨迹
- 在空间中任何具有平坦输出（输出具有合理的有界导数）的任意平滑轨迹都可以被旋翼跟踪
- 一种输出选择:

$$\sigma = [x, y, z, \psi]^T$$

- 在空间中有平坦输出的轨迹:

$$\sigma(t) = [T_0, T_M] \rightarrow \mathbb{R}^3 \times SO(2)$$



• 旋翼状态

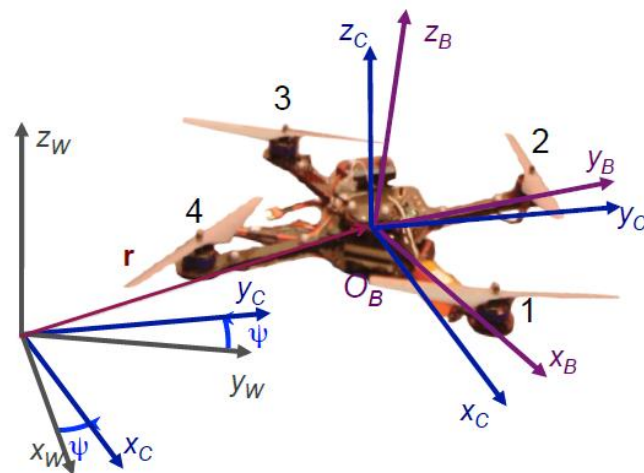
➤ 位置，姿态，线速度，角速度：

$$\mathbf{X} = [x, y, z, \phi, \theta, \psi, \dot{x}, \dot{y}, \dot{z}, \omega_x, \omega_y, \omega_z]^T$$

• 非线性动力学

牛顿方程:
$$m\ddot{\mathbf{p}} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + \mathbf{R} \begin{bmatrix} 0 \\ 0 \\ F_1 + F_2 + F_3 + F_4 \end{bmatrix} \quad \mathbf{u}_1$$

欧拉方程:
$$\mathbf{I} \cdot \begin{bmatrix} \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} + \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \times \mathbf{I} \cdot \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} l(F_2 - F_4) \\ l(F_3 - F_1) \\ M_1 - M_2 + M_3 - M_4 \end{bmatrix} \quad \begin{matrix} \mathbf{u}_2 \\ \mathbf{u}_3 \\ \mathbf{u}_4 \end{matrix}$$





• 旋翼状态

- 位置，姿态，线速度，角速度：

$$\mathbf{X} = [x, y, z, \phi, \theta, \psi, \dot{x}, \dot{y}, \dot{z}, \omega_x, \omega_y, \omega_z]^T$$

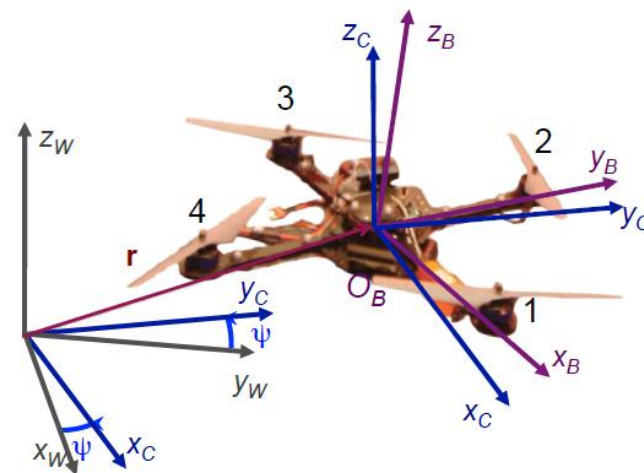
机体系下，机体旋转的
即时角速度

- 运动方程：

$$m\ddot{\mathbf{p}} = -mg\mathbf{z}_W + u_1\mathbf{z}_B.$$

$$\boldsymbol{\omega}_B = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}, \quad \dot{\boldsymbol{\omega}}_B = \mathbf{I}^{-1} \left[-\boldsymbol{\omega}_B \times \mathbf{I} \cdot \boldsymbol{\omega}_B + \begin{bmatrix} u_2 \\ u_3 \\ u_4 \end{bmatrix} \right]$$

- 位置，速度，加速度都是平坦输出 ($\boldsymbol{\sigma} = [x, y, z, \psi]^T$) 的简单导数





• 姿态角

- 旋翼姿态:

$$\mathbf{X} = [x, y, z, \phi, \theta, \psi, \dot{x}, \dot{y}, \dot{z}, \omega_x, \omega_y, \omega_z]^T$$

- 从运动方程中可得:

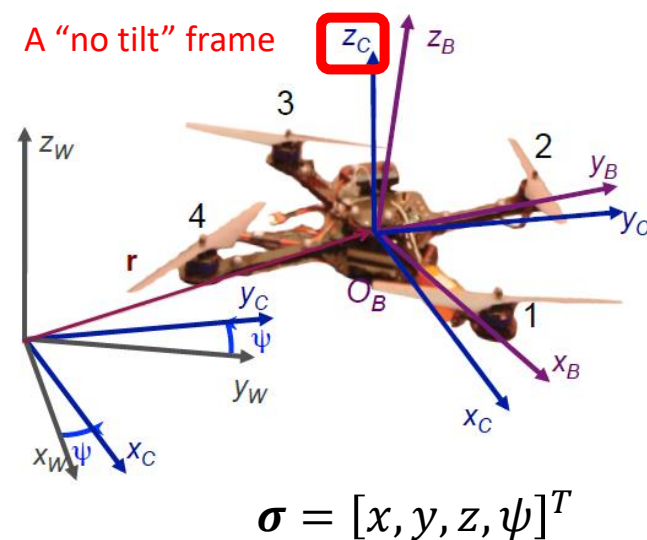
$$\mathbf{z}_B = \frac{\mathbf{t}}{\|\mathbf{t}\|}, \mathbf{t} = [\ddot{\sigma}_1, \ddot{\sigma}_2, \ddot{\sigma}_3 + g]^T$$

- 定义yaw的向量(Z-X-Y 欧拉角):

$$\mathbf{x}_C = [\cos \sigma_4, \sin \sigma_4, 0]^T$$

- 姿态角可以按如下公式获取

$$\mathbf{y}_B = \frac{\mathbf{z}_B \times \mathbf{x}_C}{\|\mathbf{z}_B \times \mathbf{x}_C\|}, \quad \mathbf{x}_B = \mathbf{y}_B \times \mathbf{z}_B \quad \mathbf{R}_B = [\mathbf{x}_B \quad \mathbf{y}_B \quad \mathbf{z}_B]$$





• 角速度

➤ 旋翼姿态:

$$\mathbf{X} = [x, y, z, \phi, \theta, \psi, \dot{x}, \dot{y}, \dot{z}, \omega_x, \omega_y, \omega_z]^T$$

➤ 对运动方程求导

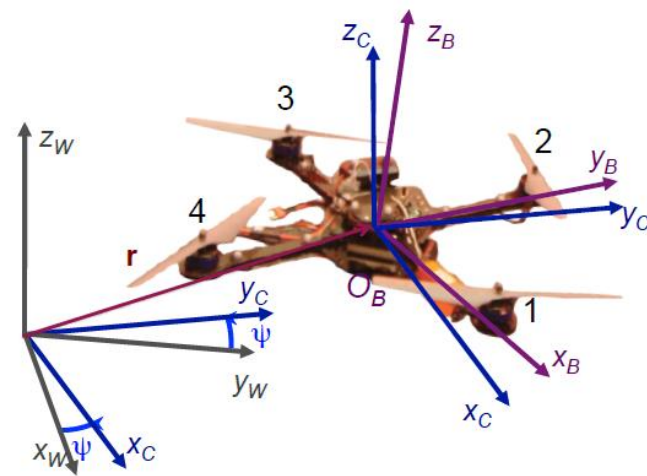
$$m\ddot{\mathbf{p}} = -mg\mathbf{z}_W + u_1\mathbf{z}_B.$$

$$m\dot{\mathbf{a}} = \dot{u}_1\mathbf{z}_B + \boxed{\boldsymbol{\omega}_{BW}} \times u_1\mathbf{z}_B$$

➤ 旋翼只有机体z轴方向有推力:

$$\dot{u}_1 = \mathbf{z}_B \cdot m\dot{\mathbf{a}}$$

世界系下，机体旋转的
即时角速度





• 角速度

➤ 旋翼姿态:

$$\mathbf{X} = [x, y, z, \phi, \theta, \psi, \dot{x}, \dot{y}, \dot{z}, \omega_x, \omega_y, \omega_z]^T$$

➤ 可推:

$$\mathbf{h}_\omega = \boldsymbol{\omega}_{BW} \times \mathbf{z}_B = \frac{m}{u_1} (\dot{\mathbf{a}} - (\mathbf{z}_B \cdot \dot{\mathbf{a}}) \mathbf{z}_B).$$

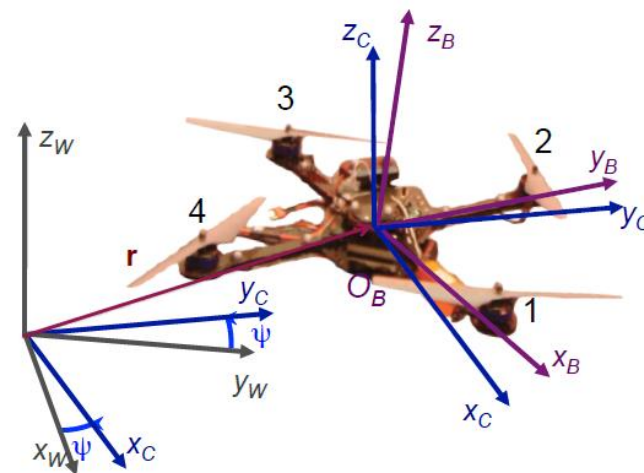
这是将 $\frac{m}{u_1} \dot{\mathbf{a}}$ 投影到 $x_B - y_B$ 平面

➤ 由于

$$\boldsymbol{\omega}_{BW} = \omega_x \mathbf{x}_B + \omega_y \mathbf{y}_B + \omega_z \mathbf{z}_B$$

➤ 沿着 \mathbf{x}_B 和 \mathbf{y}_B 方向的角速度:

$$\omega_x = -\mathbf{h}_\omega \cdot \mathbf{y}_B, \quad \omega_y = \mathbf{h}_\omega \cdot \mathbf{x}_B$$





• 角速度

➤ 旋翼姿态:

$$\mathbf{X} = [x, y, z, \phi, \theta, \psi, \dot{x}, \dot{y}, \dot{z}, \omega_x, \omega_y, \omega_z]^T$$

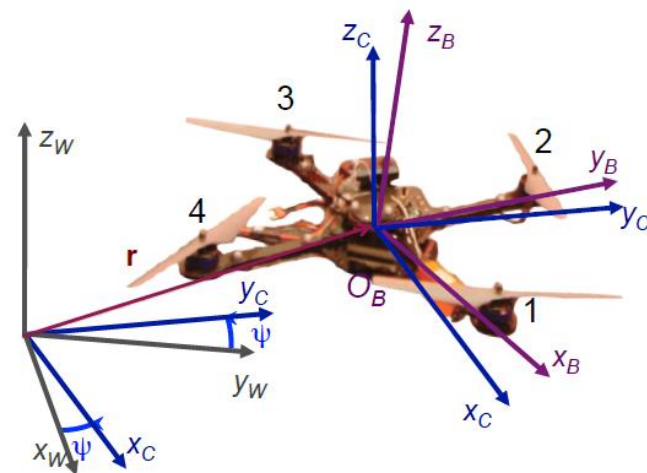
➤ 可推:

$$\mathbf{h}_\omega = \boldsymbol{\omega}_{BW} \times \mathbf{z}_B = \frac{m}{u_1} (\dot{\mathbf{a}} - (\mathbf{z}_B \cdot \dot{\mathbf{a}}) \mathbf{z}_B).$$

这是将 $\frac{m}{u_1} \dot{\mathbf{a}}$ 投影到 $x_B - y_B$ 平面

➤ 由于 $\boldsymbol{\omega}_{BW} = \boldsymbol{\omega}_{BC} + \boldsymbol{\omega}_{CW}$, $\boldsymbol{\omega}_{BC}$ 没有 z_B 分量:

$$\omega_z = \boldsymbol{\omega}_{BW} \cdot \mathbf{z}_B = \boldsymbol{\omega}_{CW} \cdot \mathbf{z}_B = \dot{\psi} \mathbf{z}_W \cdot \mathbf{z}_B$$





• 总结

- 旋翼姿态:

$$\mathbf{X} = [x, y, z, \phi, \theta, \psi, \dot{x}, \dot{y}, \dot{z}, \omega_x, \omega_y, \omega_z]^T$$

- 平坦的输出:

$$\boldsymbol{\sigma} = [x, y, z, \psi]^T$$

- 位置, 速度, 加速度:

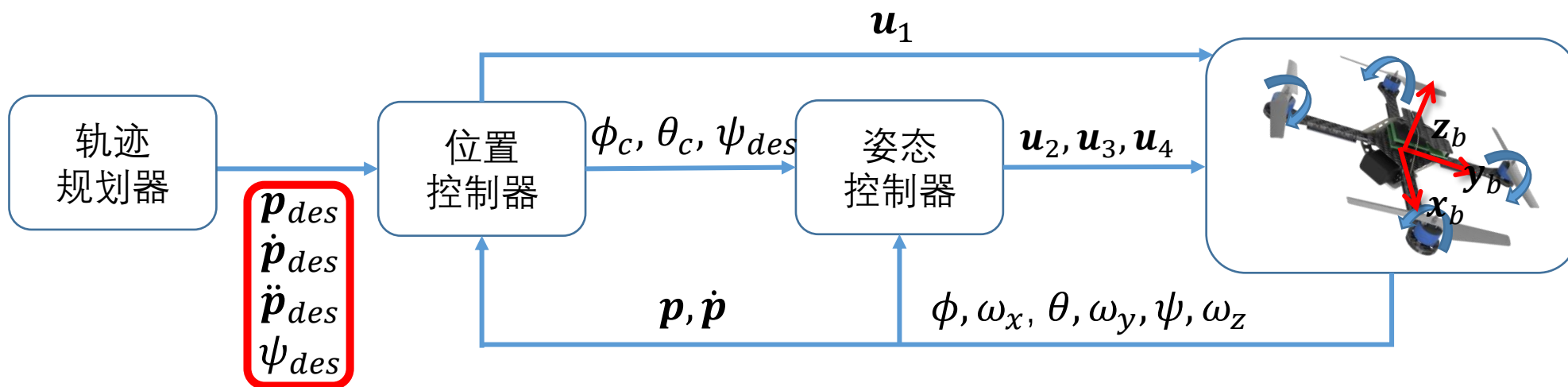
平坦输出的导数

- 姿态:

$$\mathbf{x}_C = [\cos\sigma_4, \sin\sigma_4, 0]^T \quad \mathbf{R}_B = [\mathbf{x}_B \ \mathbf{y}_B \ \mathbf{z}_B]$$

- 角速度:

$$\omega_x = -\mathbf{h}_\omega \cdot \mathbf{y}_B, \quad \omega_y = \mathbf{h}_\omega \cdot \mathbf{x}_B, \quad \omega_z = \dot{\psi} \mathbf{z}_w \cdot \mathbf{z}_B$$



• 非线性动力学

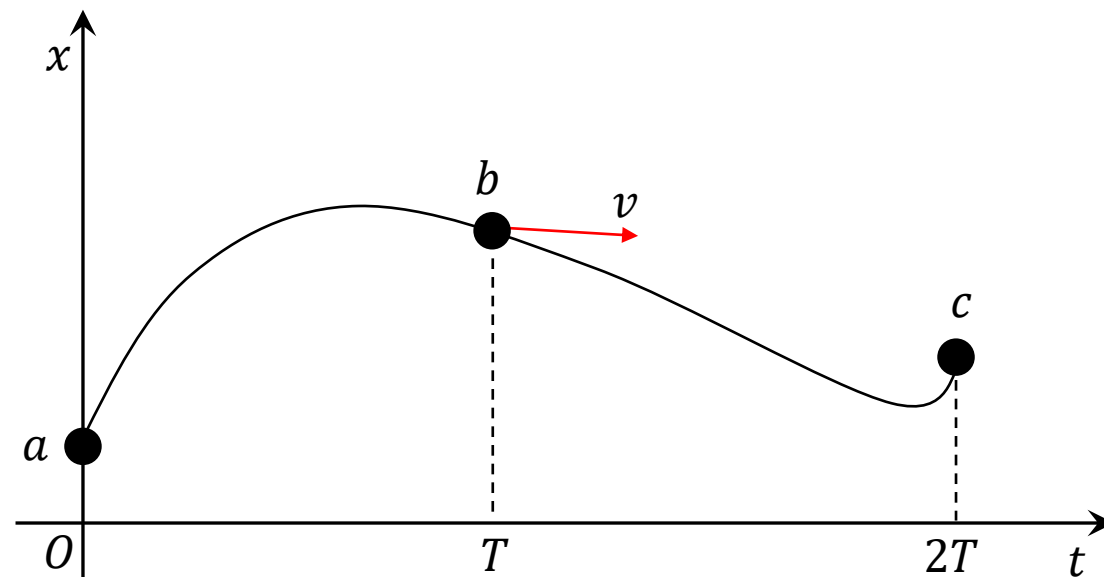
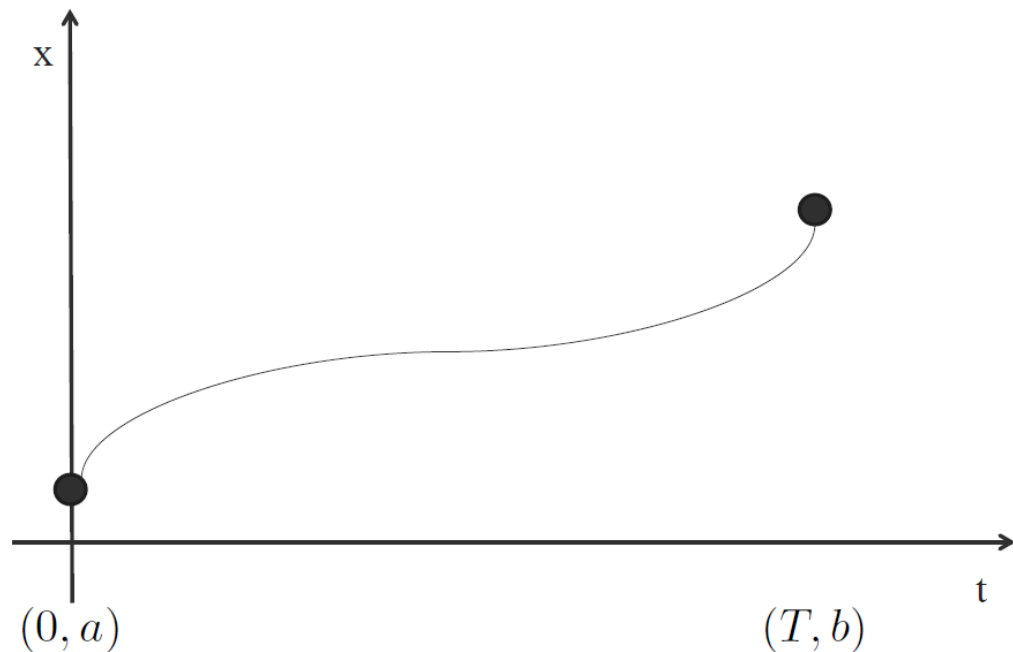
牛顿方程:
$$m\ddot{\mathbf{p}} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + \mathbf{R} \begin{bmatrix} 0 \\ 0 \\ \boxed{F_1 + F_2 + F_3 + F_4} \end{bmatrix} \mathbf{u}_1$$

欧拉方程:
$$\mathbf{I} \cdot \begin{bmatrix} \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} + \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \times \mathbf{I} \cdot \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} \boxed{l(F_2 - F_4)} \\ \boxed{l(F_3 - F_1)} \\ \boxed{M_1 - M_2 + M_3 - M_4} \end{bmatrix} \begin{matrix} \mathbf{u}_2 \\ \mathbf{u}_3 \\ \mathbf{u}_4 \end{matrix}$$



- 设计一条轨迹 $x(t)$ 使得:
 - $x(0) = a$
 - $x(T) = b$
- 轨迹光滑性通过轨迹参数化形式保证

} 边界条件





- 5th 多项式轨迹:

$$x(t) = p_5 t^5 + p_4 t^4 + p_3 t^3 + p_2 t^2 + p_1 t + p_0$$



轨迹参数化

- 边界条件

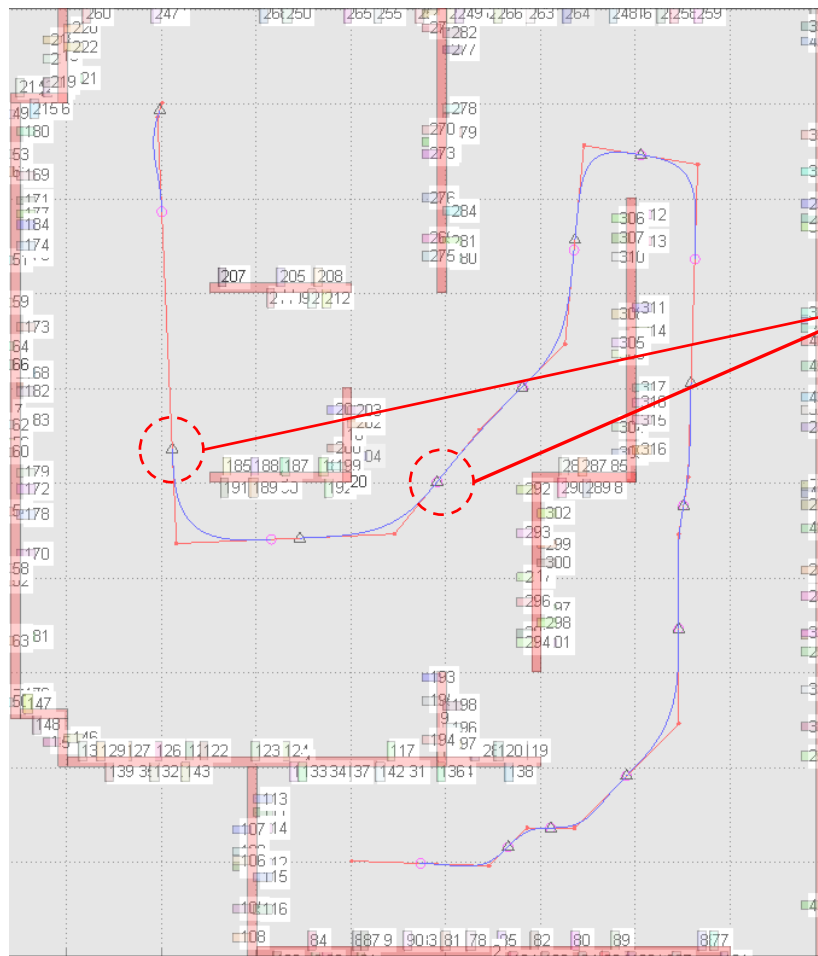
	位置	速度	加速度
t = 0	a	0	0
t = T	b	0	0

- 求解:

$$\begin{bmatrix} a \\ b \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ T^5 & T^4 & T^3 & T^2 & T & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 5T^4 & 4T^3 & 3T^2 & 2T & 1 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 20T^3 & 12T^2 & 6T & 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_5 \\ p_4 \\ p_3 \\ p_2 \\ p_1 \\ p_0 \end{bmatrix}$$



- 光滑直线段的转角处
- 倾向于以 v 作匀速运动
- 倾向于零加速度
- 短的直线段需要特殊处理



中间条件



- 在每个维度上仍使用相同的参数化形式:

- $$x(t) = p_5 t^5 + p_4 t^4 + p_3 t^3 + p_2 t^2 + p_1 t + p_0$$

- 边界条件

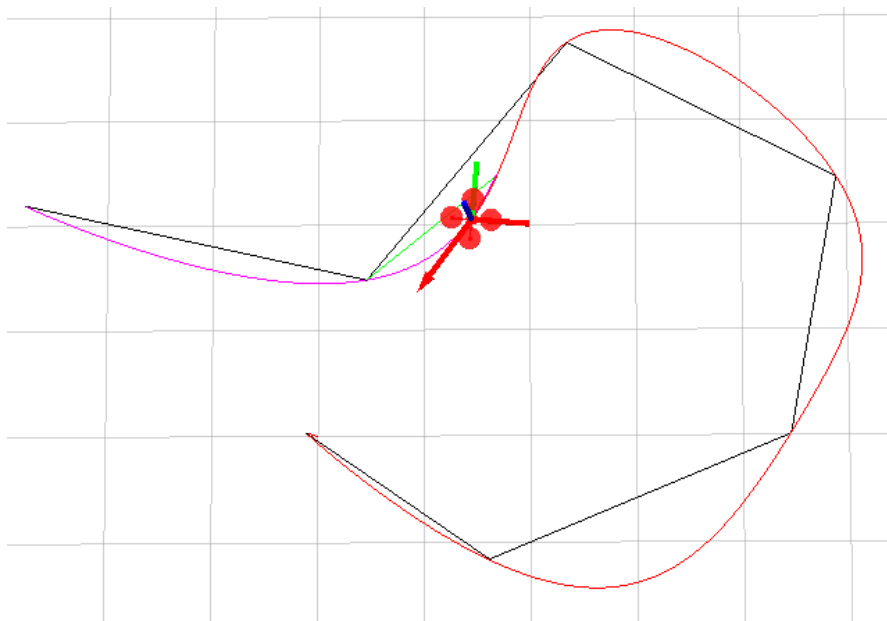
	位置	速度	加速度
$t = 0$	a	v_0	0
$t = T$	b	v_T	0

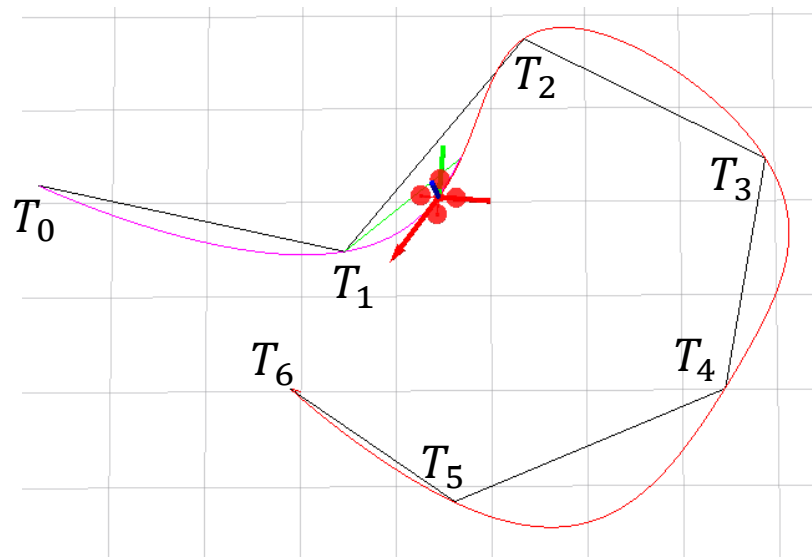
- 求解:

$$\begin{bmatrix} a \\ b \\ v_0 \\ v_T \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ T^5 & T^4 & T^3 & T^2 & T & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 5T^4 & 4T^3 & 3T^2 & 2T & 1 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 20T^3 & 12T^2 & 6T & 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_5 \\ p_4 \\ p_3 \\ p_2 \\ p_1 \\ p_0 \end{bmatrix}$$



- 边界条件: 起点、目标位置 (方向)
- 中间状态: 路径点位置 (方向)
 - 路径点可以通过路径搜索 (A*、RRT*等) 找到
- 光滑约束
 - 通常转化为最小化 “输入” 的变化率





$$f(t) = \begin{cases} f_1(t) \doteq \sum_{i=0}^N p_{1,i} t^i & T_0 \leq t \leq T_1 \\ f_2(t) \doteq \sum_{i=0}^N p_{2,i} t^i & T_1 \leq t \leq T_2 \\ \vdots & \vdots \\ f_M(t) \doteq \sum_{i=0}^N p_{M,i} t^i & T_{M-1} \leq t \leq T_M \end{cases}$$

问题定义：

- 轨迹的**初始（或初末）**状态给定；
- 轨迹在**预定时刻**需经过**预设航点**；
- 轨迹的**4阶导数**处处存在
- 最小化轨迹**4阶导数平方幅值积分**。

重要性质：

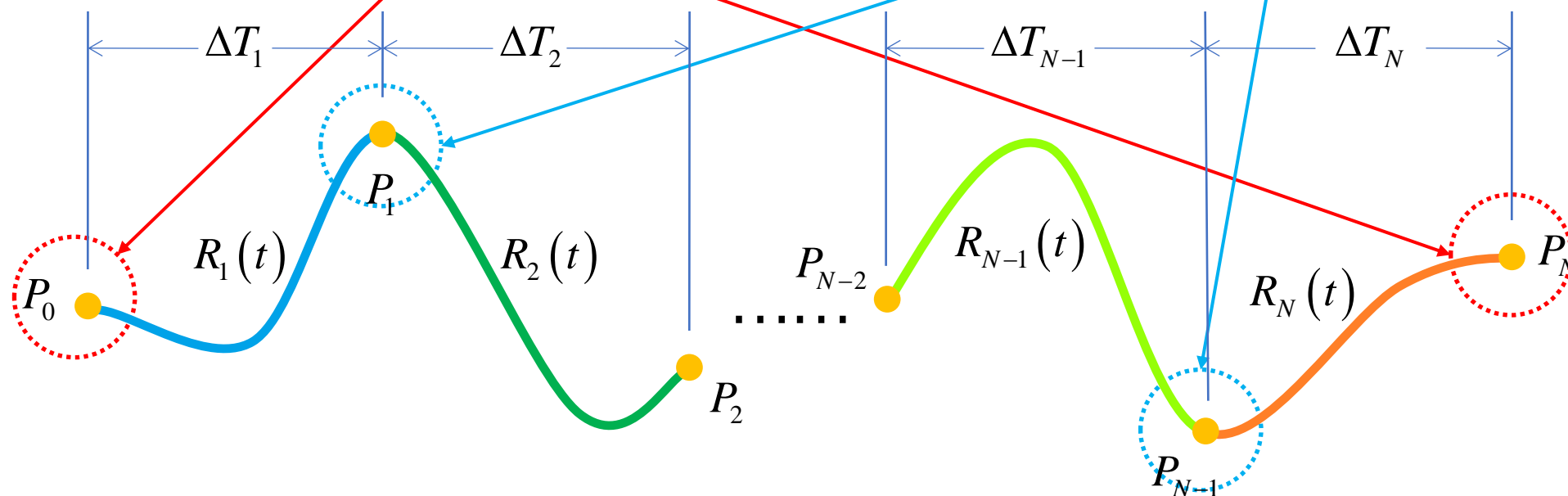
- 最优轨迹的参数化形式为**分段多项式**；
- 每段多项式的次数不超过 $2 \times 4 - 1 = 7$ ，即**每段均可以被7次多项式表征**。



约束:

- 导数给定值约束:
$$\begin{cases} f_j^{(k)}(T_{j-1}) = x_{0,j}^{(k)} \\ f_j^{(k)}(T_j) = x_{T,j}^{(k)} \end{cases}$$

- 高阶连续性约束:
$$f_j^{(k)}(T_j) = f_{j+1}^{(k)}(T_j)$$



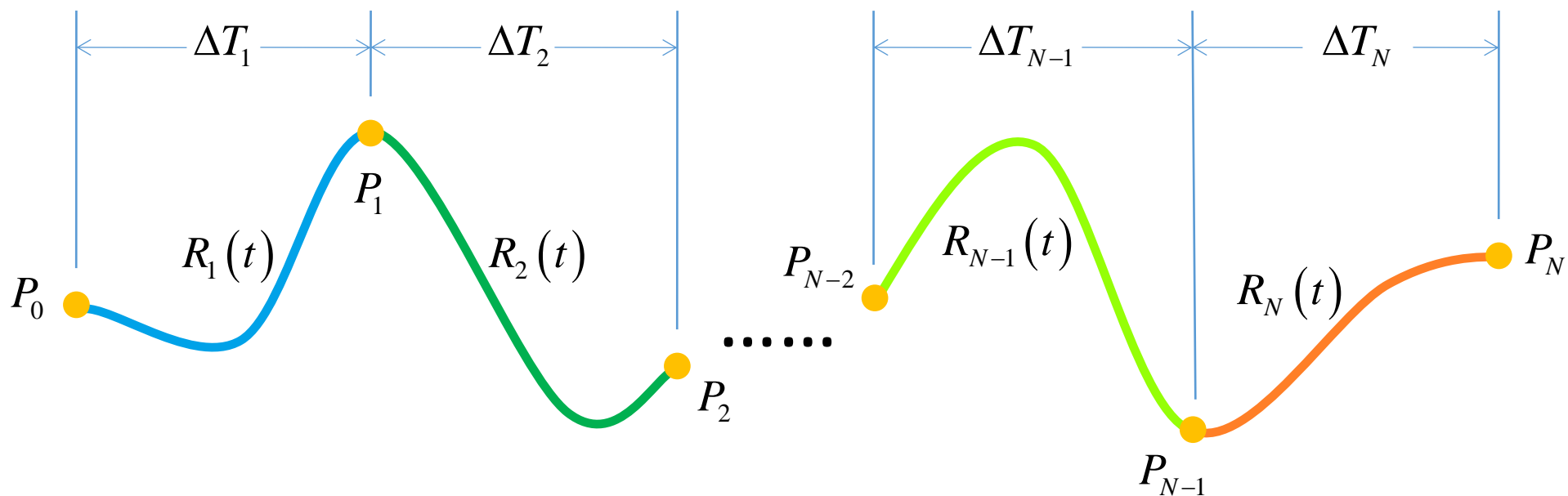


如何合理的确定轨迹的阶数？

- 确保轨迹多少阶光滑。
- 确保轨迹多少阶连续。
- 最小化控制输入。

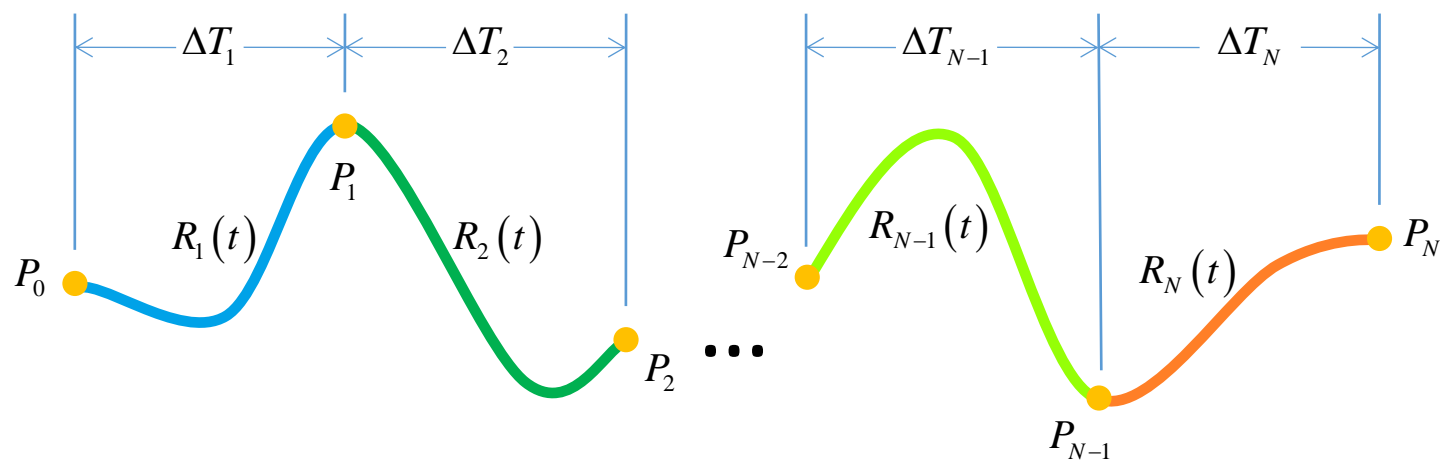
光滑性意味着它的导数是连续的！

施加的约束独立！





- 确保单段平滑多项式轨迹的最小次数:
 - Minimum jerk: $N = 2 * 3(\text{jerk}) - 1 = 5$
 - Minimum snap: $N = 2 * 4(\text{snap}) - 1 = 7$
- 确保多段平滑多项式轨迹的最小次数:



Minimum jerk:

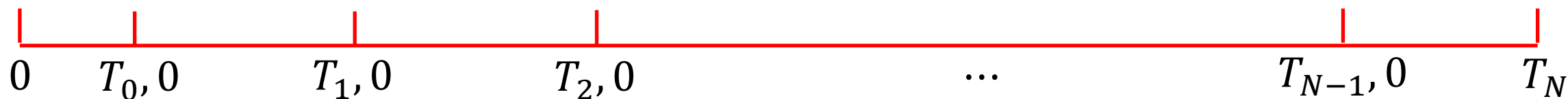
约束数量: $3 + 3 + (k-1) = k + 5$

未知数数量: $(N+1) * k$

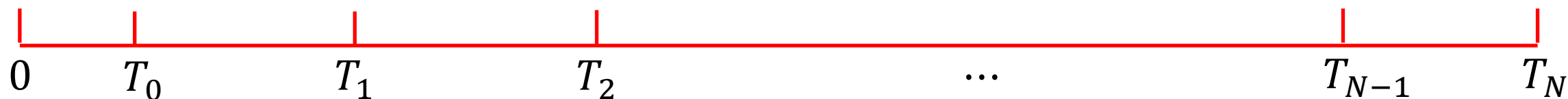
$$(N + 1) * k = k + 5 \quad N = \frac{5}{k}$$



两种时间参数方案



方案1：相对每一段初始的时间 优点：数值稳定性高



方案2：相对第一段初始的时间 优点：数学表达清楚



目标函数的解析表达

半正定二次型

$$f(t) = \sum_i p_i t^i$$

$$\Rightarrow f^{(4)}(t) = \sum_{i \geq 4} i(i-1)(i-2)(i-3)t^{i-4}p_i$$

$$\Rightarrow \left(f^{(4)}(t)\right)^2 = \sum_{i \geq 4, l \geq 4} i(i-1)(i-2)(i-3)l(l-1)(l-2)(l-3)t^{i+l-8}p_i p_l$$

$$\Rightarrow J(T) = \int_{T_{j-1}}^{T_j} \left(f^{(4)}(t)\right)^2 dt = \sum_{i \geq 4, l \geq 4} \frac{i(i-1)(i-2)(i-3)l(l-1)(l-2)(l-3)}{i+l-7} (T_j^{i+l-7} - T_{j-1}^{i+l-7}) p_i p_l$$

$$\Rightarrow J(T) = \int_{T_{j-1}}^{T_j} \left(f^{(4)}(t)\right)^2 dt = \begin{bmatrix} \vdots \\ p_i \\ \vdots \end{bmatrix}^T \begin{bmatrix} \vdots & \dots & \frac{i(i-1)(i-2)(i-3)l(l-1)(l-2)(l-3)}{i+l-7} T^{i+l-7} & \dots & \vdots \end{bmatrix} \begin{bmatrix} \vdots \\ p_l \\ \vdots \end{bmatrix}$$

$$J_j(T) = \mathbf{p}_j^T \mathbf{Q}_j \mathbf{p}_j$$



指定导数值约束：0阶导给定即航点坐标

$$\bullet f_j^{(k)}(T_j) = x_j^{(k)}$$

$$\Rightarrow \sum_{i \geq k} \frac{i!}{(i-k)!} T_j^{i-k} p_{j,i} = x_{T,j}^{(k)}$$

$$\Rightarrow \begin{bmatrix} \cdots & \frac{i!}{(i-k)!} T_j^{i-k} & \cdots \end{bmatrix} \begin{bmatrix} \vdots \\ p_{j,i} \\ \vdots \end{bmatrix} = x_{T,j}^{(k)}$$

$$\Rightarrow \begin{bmatrix} \cdots & \frac{i!}{(i-k)!} T_{j-1}^{i-k} & \cdots \\ \cdots & \frac{i!}{(i-k)!} T_j^{i-k} & \cdots \end{bmatrix} \begin{bmatrix} \vdots \\ p_{j,i} \\ \vdots \end{bmatrix} = \begin{bmatrix} x_{0,j}^{(k)} \\ x_{T,j}^{(k)} \end{bmatrix}$$

$$\Rightarrow \mathbf{A}_j \mathbf{p}_j = \mathbf{d}_j$$

$$x(t) = p_5 t^5 + p_4 t^4 + p_3 t^3 + p_2 t^2 + p_1 t + p_0$$

$$x(0) = \cdots, x(T) = \cdots$$

$$\dot{x}(0) = \cdots, \dot{x}(T) = \cdots$$

$$\ddot{x}(0) = \cdots, \ddot{x}(T) = \cdots$$

$$\vdots$$

$$p_0 = \cdots,$$

$$p_5 T^5 + p_4 T^4 + p_3 T^3 + p_2 T^2 + p_1 T + p_0 = \cdots$$

$$[T^5, T^4, T^3, T^2, T, 1] \begin{bmatrix} p_5 \\ p_4 \\ p_3 \\ p_2 \\ p_1 \\ p_0 \end{bmatrix} = \cdots$$



高阶连续性约束：保证两段之间的4阶导连续

$$f_j^{(k)}(T_j) = f_{j+1}^{(k)}(T_j)$$

$$\Rightarrow \sum_{i \geq k} \frac{i!}{(i-k)!} T_j^{i-k} p_{j,i} - \sum_{l \geq k} \frac{l!}{(l-k)!} T_j^{l-k} p_{j+1,l} = 0$$

$$\Rightarrow \left[\dots \quad \frac{i!}{(i-k)!} T_j^{i-k} \quad \dots \quad -\frac{l!}{(l-k)!} T_j^{l-k} \quad \dots \right] \begin{bmatrix} \vdots \\ p_{j,i} \\ \vdots \\ p_{j+1,l} \\ \vdots \end{bmatrix} = 0$$

$$\Rightarrow [\mathbf{A}_j \quad -\mathbf{A}_{j+1}] \begin{bmatrix} \mathbf{p}_j \\ \mathbf{p}_{j+1} \end{bmatrix} = 0$$



线性等式约束的二次规划

$$\begin{aligned} \min \quad & \begin{bmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_M \end{bmatrix}^T \begin{bmatrix} \mathbf{Q}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Q}_M \end{bmatrix} \begin{bmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_M \end{bmatrix} \\ \text{s.t.} \quad & \mathbf{A}_{eq} \begin{bmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_M \end{bmatrix} = \mathbf{d}_{eq} \end{aligned}$$

最小化Snap的轨迹生成是一个典型的**凸优化**问题。



凸方程

- 一个方程 $f: R^n \rightarrow R$ 被称作凸方程, 如果域 $\text{dom } f$ 是凸, 并且对于任意 $x, y \in \text{dom } f$ 和 $0 \leq \theta \leq 1$,

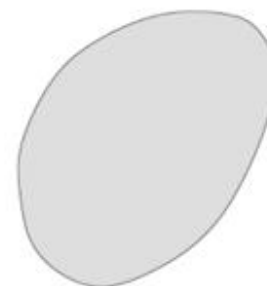
$$f(\theta x + (1 - \theta)y) \leq \theta f(x) + (1 - \theta)f(y)$$



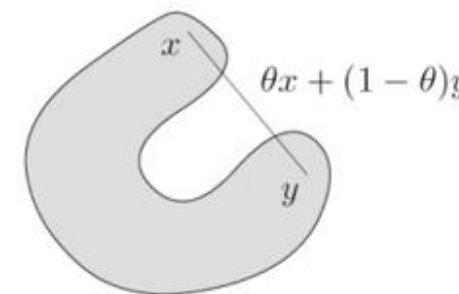
- 如果不等关系对 $0 < \theta < 1$ 严格成立, 那么 f 是严格凸的
- 如果 $-f$ 是凸, 那么 f 是凹的

凸集

- 一个集合 $C \in R^n$ 被称作凸集, 如果任意两点间的线段在集合中: 对于任意 $x, y \in C$ 且 $0 \leq \theta \leq 1$, $\theta x + (1 - \theta)y \in C$.



convex



non-convex



- 标准形式的优化问题

$$\begin{array}{ll} \underset{x}{\text{minimize}} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0 \quad i = 1, \dots, m \\ & h_i(x) = 0 \quad i = 1, \dots, p \end{array}$$

$x \in R^n$ 是优化变量

$f_0: R^n \rightarrow R$ 是目标函数

$f_i: R^n \rightarrow R, i = 1, \dots, m$ 是不等式约束

$h_i: R^n \rightarrow R, i = 1, \dots, p$ 是等式约束

- 标准形式的凸优化问题

$$\begin{array}{ll} \underset{x}{\text{minimize}} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0 \quad i = 1, \dots, m \\ & Ax = b \end{array}$$

f_0, f_1, \dots, f_m 是凸函数且等式约束是仿射变换.

- **局部和全局最优**: 任何凸问题的局部最优点是也是全局最优
- 大多数问题在被表达时是非凸的
- 将问题以凸的形式表达是一门艺术, 没有系统的方法



线性规划 Linear Programming (LP)

$$\begin{array}{ll} \underset{x}{\text{minimize}} & c^T x + d \\ \text{subject to} & Gx \leq h \\ & Ax = b \end{array}$$

- 凸问题：仿射的目标函数和约束方程

二次约束的二次规划

Quadratically Constrained QP (QCQP)

$$\begin{array}{ll} \underset{x}{\text{minimize}} & (1/2)x^T P_0 x + q_0^T x + r_0 \\ \text{subject to} & (1/2)x^T P_i x + q_i^T x + r_i \leq 0 \quad i = 1, \dots, m \\ & Ax = b \end{array}$$

- 凸问题（假设 $P_i \in S^n \geq 0$ ）：凸二次型目标函数和约束方程

二次规划 Quadratic Programming (QP)

$$\begin{array}{ll} \underset{x}{\text{minimize}} & (1/2)x^T P x + q^T x + r \\ \text{subject to} & Gx \leq h \\ & Ax = b \end{array}$$

- 凸问题（假设 $P \in S^n \geq 0$ ）：凸二次型目标函数和仿射约束方程

二次锥规划

Second-Order Cone Programming (SOCP)

$$\begin{array}{ll} \underset{x}{\text{minimize}} & f^T x \\ \text{subject to} & \|A_i x + b_i\| \leq c_i^T x + d_i \quad i = 1, \dots, m \\ & Fx = g \end{array}$$

- 凸问题：线性目标函数和二阶锥约束
- 对于 A_i 行向量，退化成LP
- 对于 $c_i = 0$ ，退化成QCQP
- 比QCQP和LP更具一般性



- 直接优化多项式轨迹在数值上不稳定
- 更倾向于替代变量，使得优化每段端点的微分
- 我们有 $M_j \mathbf{p}_j = \mathbf{d}_j$, M_j 是一个映射矩阵，将多项式系数映射至微分

$$J = \begin{bmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_M \end{bmatrix}^T \begin{bmatrix} \mathbf{Q}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Q}_M \end{bmatrix} \begin{bmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_M \end{bmatrix} \quad J = \begin{bmatrix} \mathbf{d}_1 \\ \vdots \\ \mathbf{d}_M \end{bmatrix}^T \begin{bmatrix} \mathbf{M}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{M}_M \end{bmatrix}^{-T} \begin{bmatrix} \mathbf{Q}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Q}_M \end{bmatrix} \begin{bmatrix} \mathbf{M}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{M}_M \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{d}_1 \\ \vdots \\ \mathbf{d}_M \end{bmatrix}$$



$$J = \begin{bmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_M \end{bmatrix}^T \begin{bmatrix} \mathbf{Q}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Q}_M \end{bmatrix} \begin{bmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_M \end{bmatrix} \quad + \quad \mathbf{M}_j \mathbf{p}_j = \mathbf{d}_j$$



$$J = \begin{bmatrix} \mathbf{d}_1 \\ \vdots \\ \mathbf{d}_M \end{bmatrix}^T \begin{bmatrix} \mathbf{M}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{M}_M \end{bmatrix}^{-T} \begin{bmatrix} \mathbf{Q}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Q}_M \end{bmatrix} \begin{bmatrix} \mathbf{M}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{M}_M \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{d}_1 \\ \vdots \\ \mathbf{d}_M \end{bmatrix}$$

$$x(t) = p_5 t^5 + p_4 t^4 + p_3 t^3 + p_2 t^2 + p_1 t + p_0$$

$$x'(t) = 5p_5 t^4 + 4p_4 t^3 + 3p_3 t^2 + 2p_2 t + p_1$$

$$x''(t) = 20p_5 t^3 + 12p_4 t^2 + 6p_3 t + 2p_2$$

$$M = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ T^5 & T^4 & T^3 & T^2 & T & 1 \\ 5T^4 & 4T^3 & 3T^2 & 2T & 1 & 0 \\ 20T^3 & 12T^2 & 6T & 2 & 0 & 0 \end{bmatrix}$$



- 使用选择矩阵C来分离自由变量(\mathbf{d}_P) 和受约束变量(\mathbf{d}_F)
 - 自由变量：微分不受指定，只受连续性约束

$$\mathbf{C}^T \begin{bmatrix} \mathbf{d}_F \\ \mathbf{d}_P \end{bmatrix} = \begin{bmatrix} \mathbf{d}_1 \\ \vdots \\ \mathbf{d}_M \end{bmatrix} \quad \longrightarrow \quad J = \begin{bmatrix} \mathbf{d}_F \\ \mathbf{d}_P \end{bmatrix}^T \underbrace{\mathbf{C}\mathbf{M}^{-T}\mathbf{Q}\mathbf{M}^{-1}\mathbf{C}^T}_{\mathbf{R}} \begin{bmatrix} \mathbf{d}_F \\ \mathbf{d}_P \end{bmatrix} = \begin{bmatrix} \mathbf{d}_F \\ \mathbf{d}_P \end{bmatrix}^T \begin{bmatrix} \mathbf{R}_{FF} & \mathbf{R}_{FP} \\ \mathbf{R}_{PF} & \mathbf{R}_{PP} \end{bmatrix} \begin{bmatrix} \mathbf{d}_F \\ \mathbf{d}_P \end{bmatrix}$$

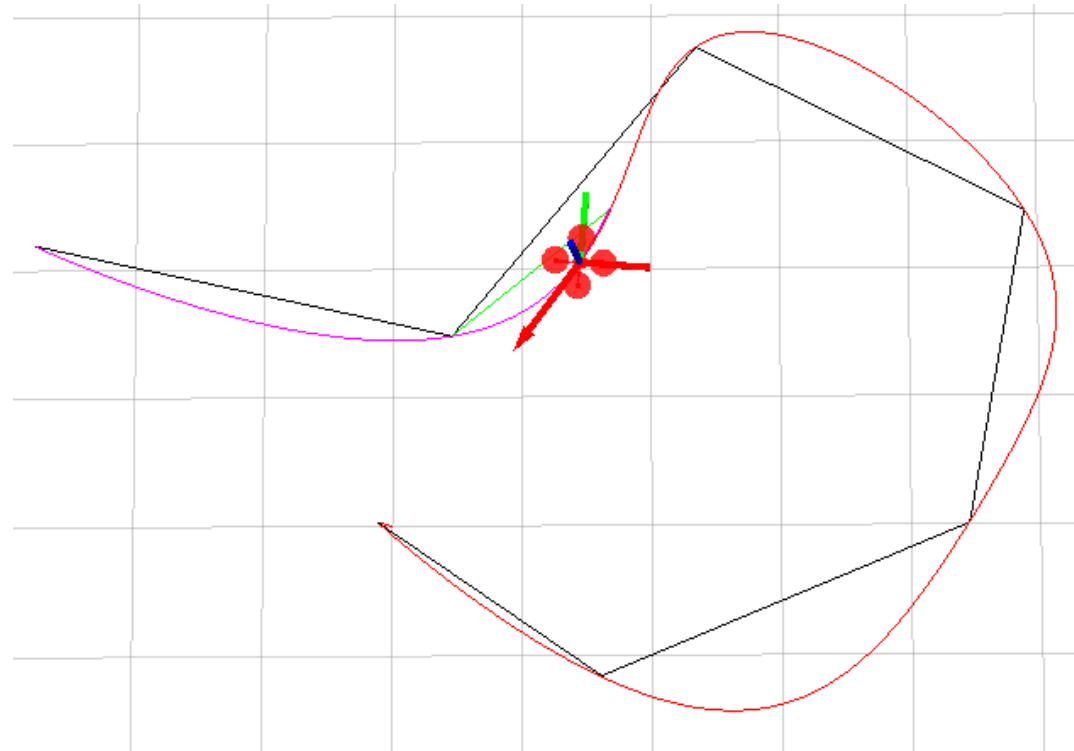
- 转变为一个不受约束的二次规划问题，可直接闭式求解

$$J = \mathbf{d}_F^T \mathbf{R}_{FF} \mathbf{d}_F + \mathbf{d}_F^T \mathbf{R}_{FP} \mathbf{d}_P + \mathbf{d}_P^T \mathbf{R}_{PF} \mathbf{d}_F + \mathbf{d}_P^T \mathbf{R}_{PP} \mathbf{d}_P$$

$$\mathbf{d}_P^* = -\mathbf{R}_{PP}^{-1} \mathbf{R}_{FP}^T \mathbf{d}_F$$

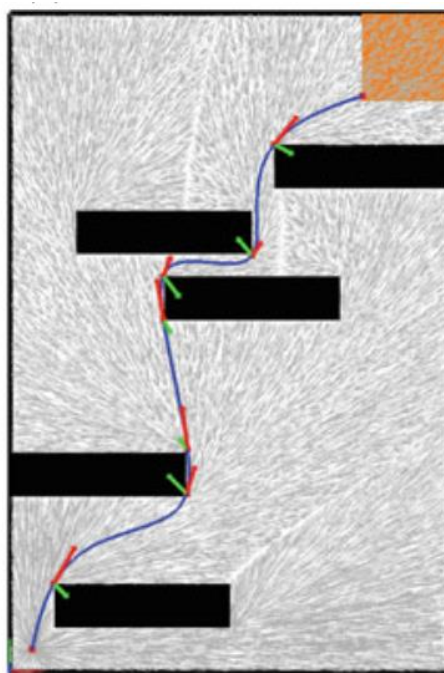


- 最终轨迹

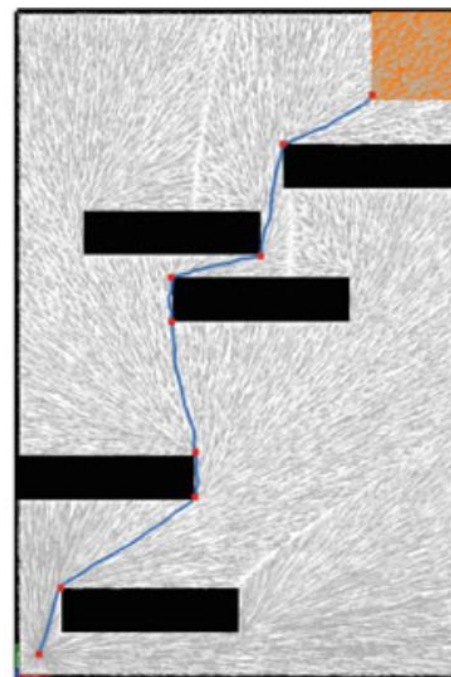


问：怎样获得航点？

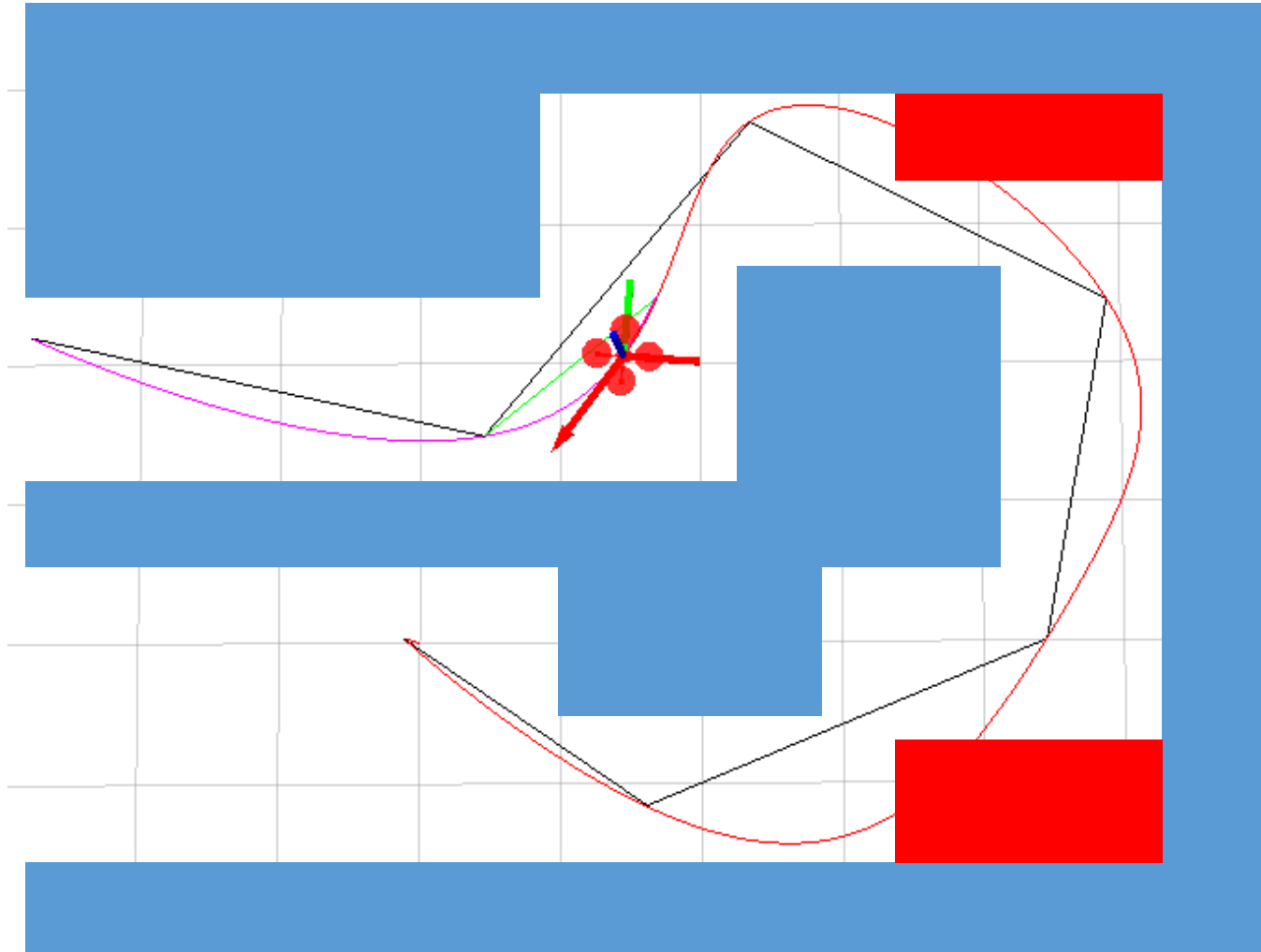
- 路径规划+ 轨迹生成
 - 低复杂度：路径规划的问题维度低



给定航点的轨迹生成具有闭式解

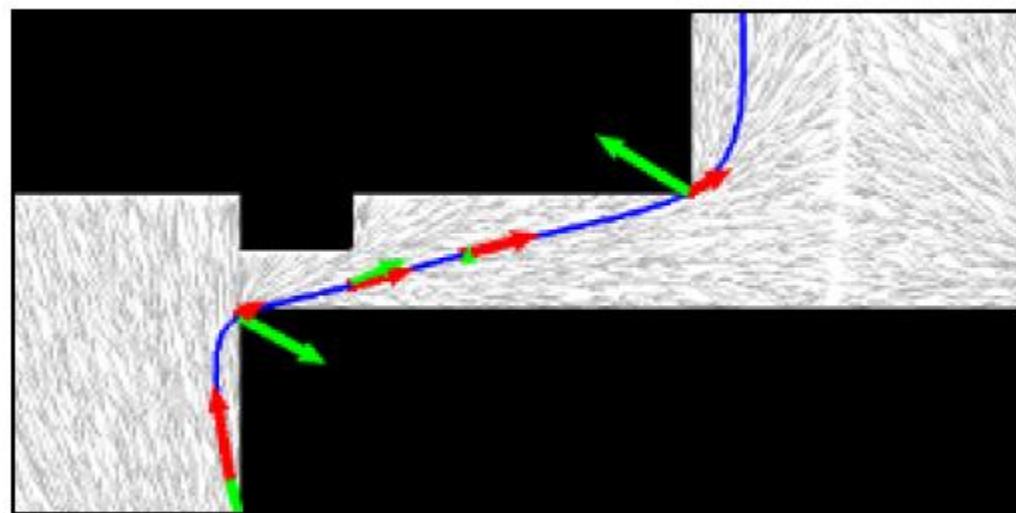
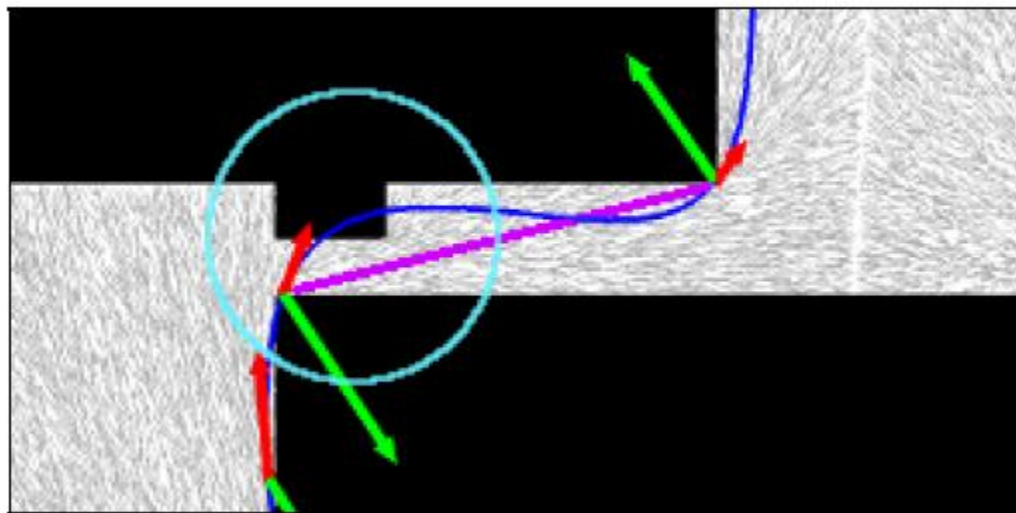


如何获取无碰撞的航点：路径规划



Q: 如何确保轨迹无碰撞?

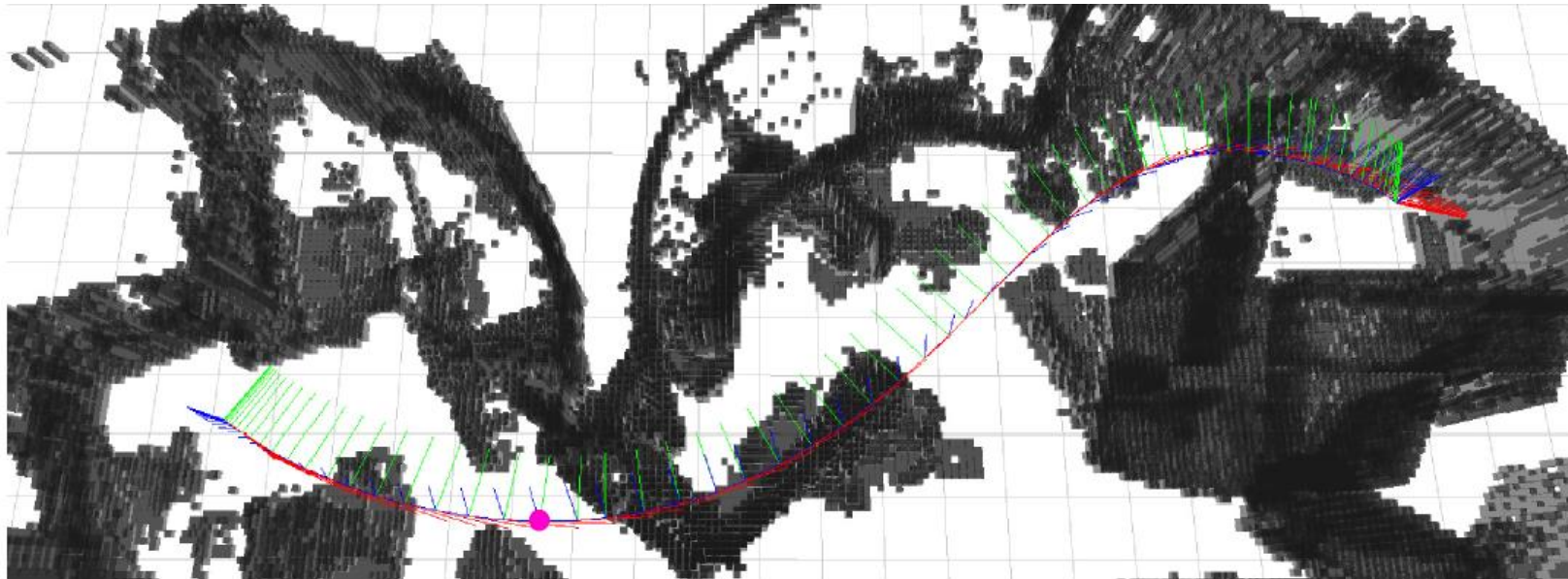
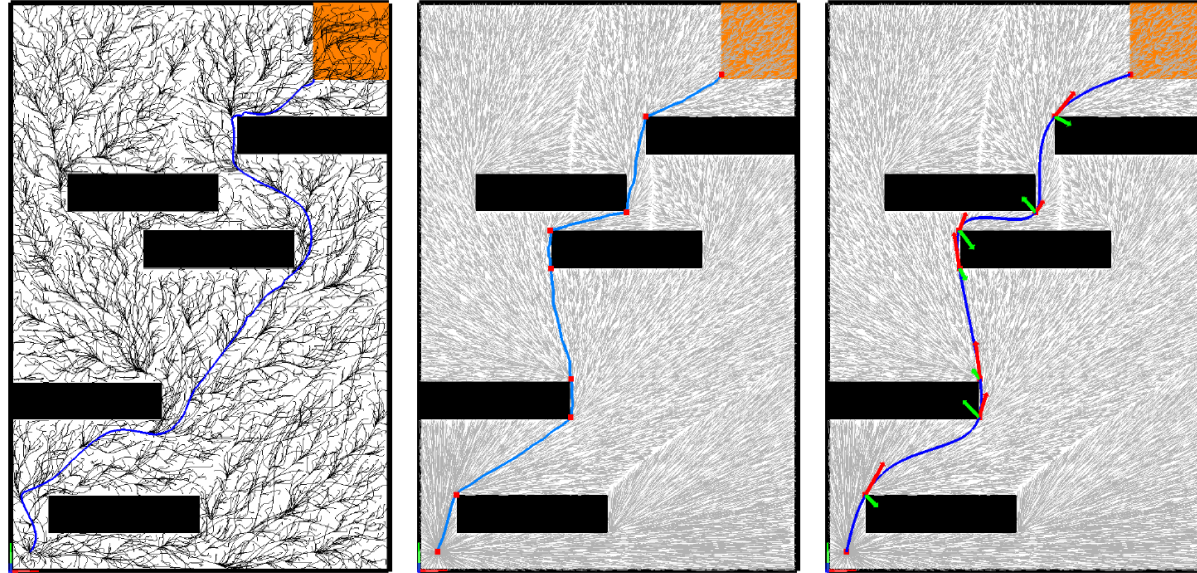
- 初始路径无碰撞
- 在初始路径的航点上利用闭式解生成轨迹
- 轨迹的碰撞部分则迭代地加入无碰撞的中间航点





RRT*+最小化Snap轨迹生成

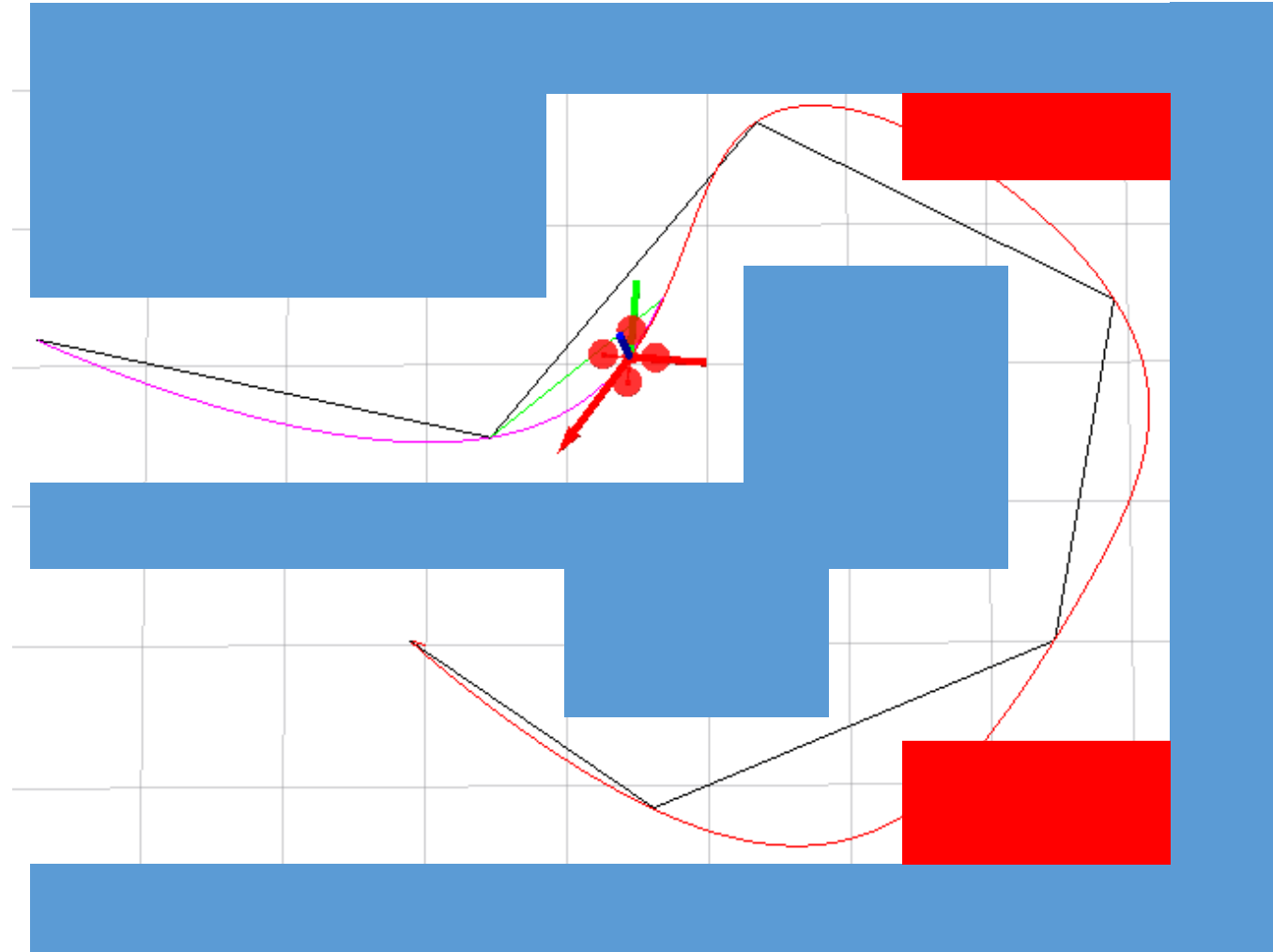
浙江大学·控制学院





Better solutions?

浙江大学 · 控制学院



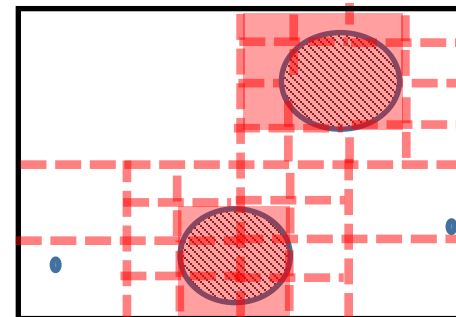


保证避障的平滑轨迹生成

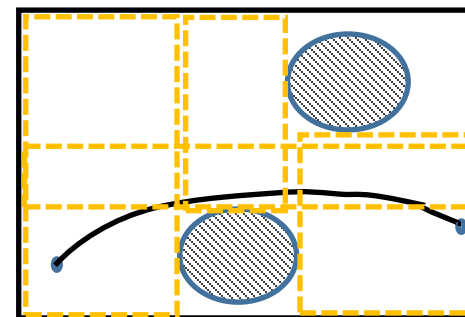
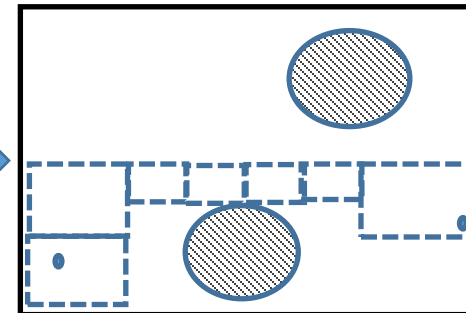
浙江大学·控制学院



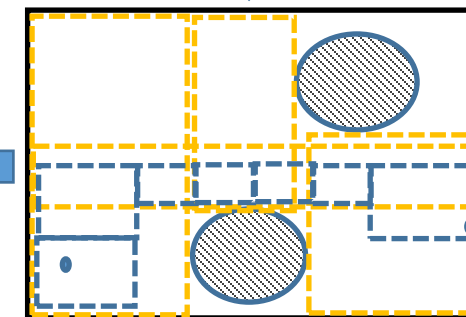
Step 1. 障碍物检测



Step 2. 寻找飞行走廊



Step 4. 在走廊内生成轨迹



Step 3. 膨胀飞行走廊

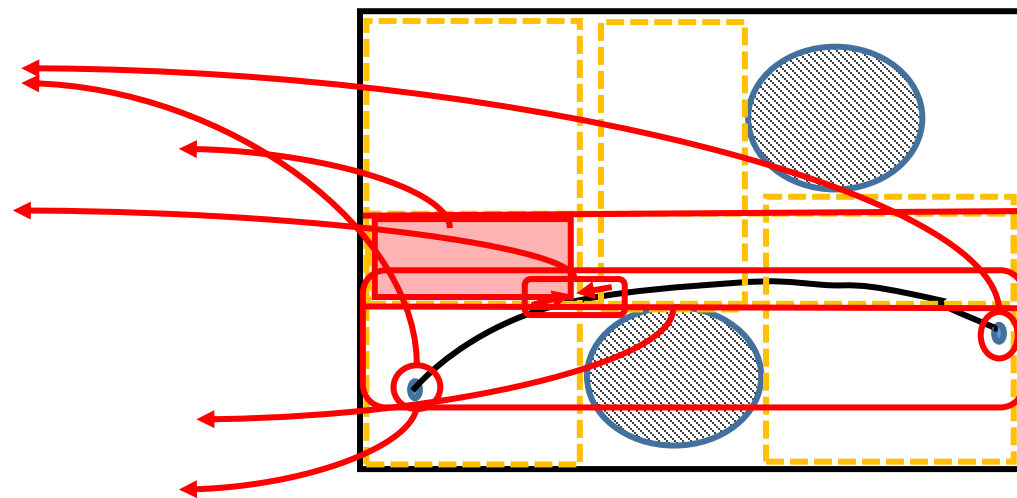


- **点**线性约束:

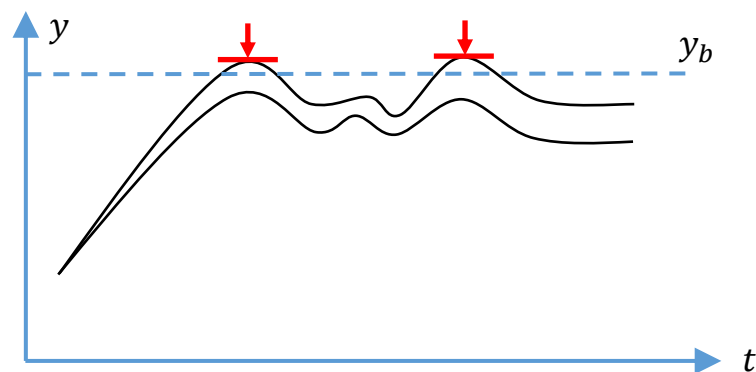
- 起点终点的状态约束($\mathbf{A}\mathbf{p} = \mathbf{b}$)
- 中间轨迹点状态约束($\mathbf{A}\mathbf{p} = \mathbf{b}, \mathbf{A}\mathbf{p} \leq \mathbf{b}$)
- 分段连续性约束($\mathbf{A}\mathbf{p}_i = \mathbf{A}\mathbf{p}_{i+1}$)

- **区间**线性约束:

- 飞行走廊边界约束($\mathbf{A}(t)\mathbf{p} \leq \mathbf{b}, \forall t \in [t_l, t_r]$)
- 动力学约束($\mathbf{A}(t)\mathbf{p} \leq \mathbf{b}, \forall t \in [t_l, t_r]$)
 - 速度
 - 加速度



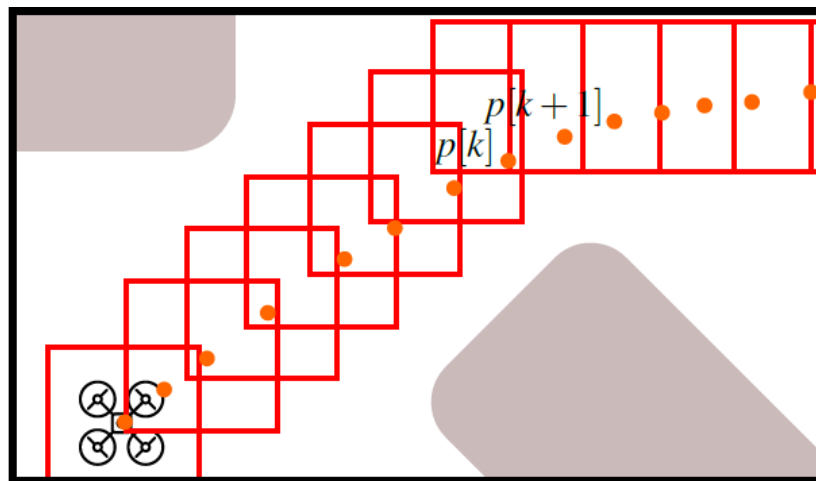
- 方案1: 反复检查极值并添加额外的约束。



Online generation of collision-free trajectories for quadrotor flight in unknown cluttered environments, J. Chen, ICRA 2016

- 迭代求解非常耗时。
- 严格来说, 没有可行的解决方案满足所有约束。
需运行10次迭代来确定解决方案的状态?

- 方案2: 在离散时间点添加大量约束。



A hybrid method for online trajectory planning of mobile robots in cluttered environments, L Campos-Macías, RAL 2017

- 生成轨迹过于保守。
- 约束太多, 计算负担高。

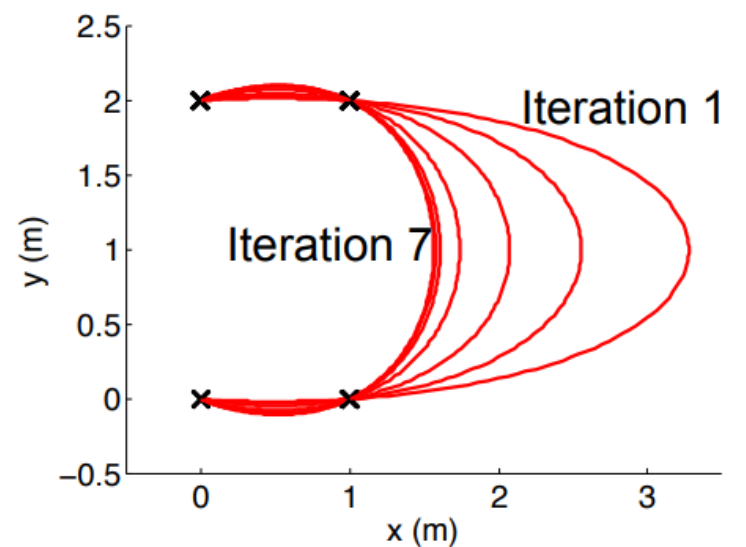
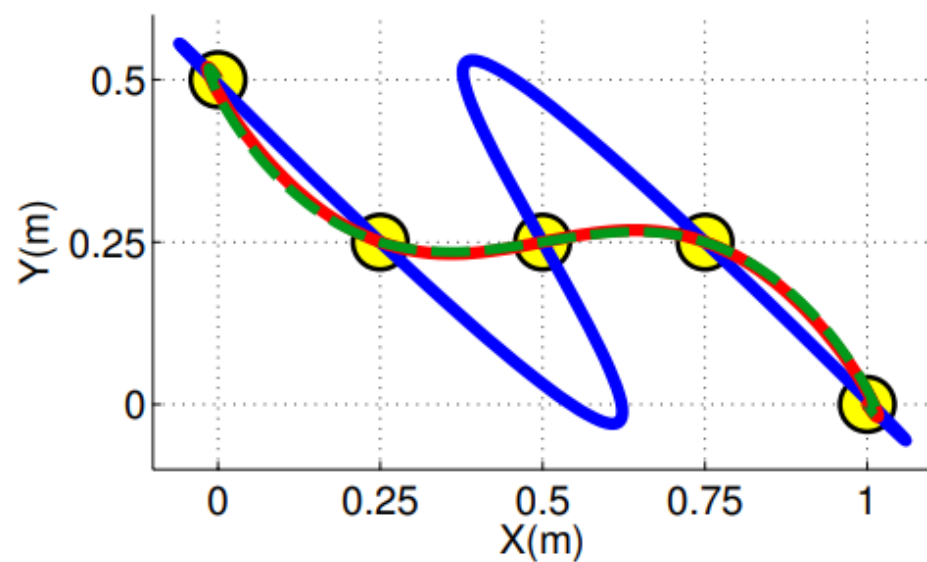


Time Allocation

时间分配

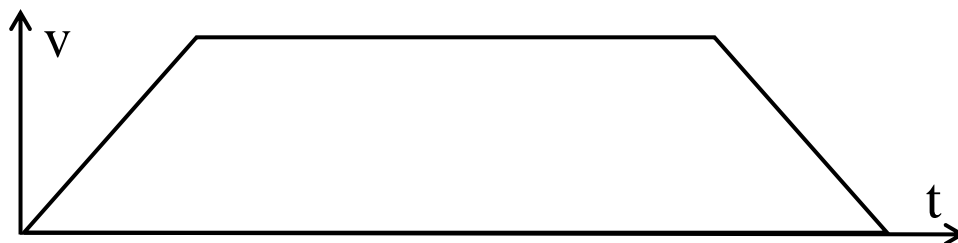


- 分段轨迹取决于分段时间分配。
- 时间分配显著影响最终轨迹。
- 如何合理分配时间？

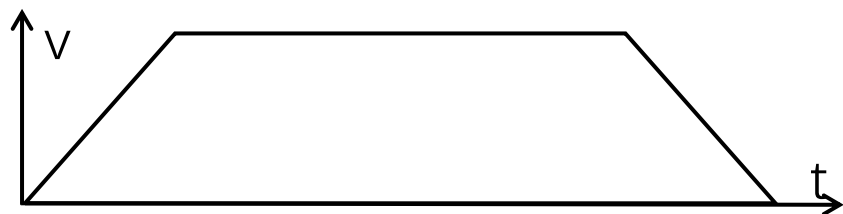




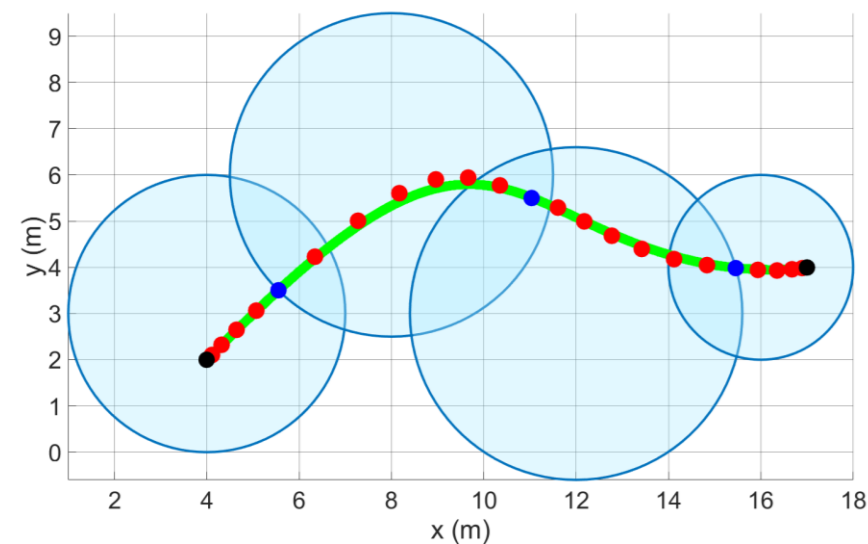
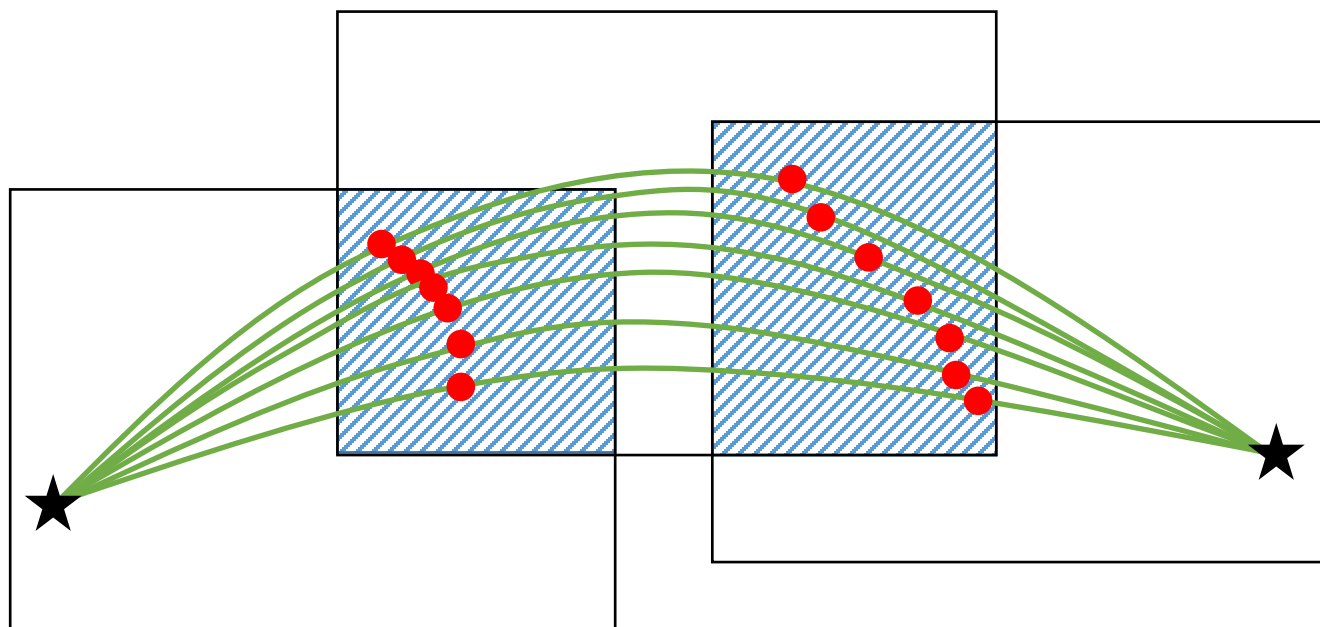
- 使用“T形速度时间”曲线来获得段轨迹的持续时间。
 - 假设每一段轨迹：匀加速->最大速度->匀减速->速度至0.
- 使用预期平均速度来获得每段轨迹的持续时间。



- 远非最优时间分配
- 只能获得保守的时间分配
- 对环境无法变通



- 允许轨迹点在飞行走廊的重合区域内进行调节，提供一定解空间的自由度。
- 只给定了段与段之间的时间 T ，但一定程度上缓解了时间分配不好带来的问题。

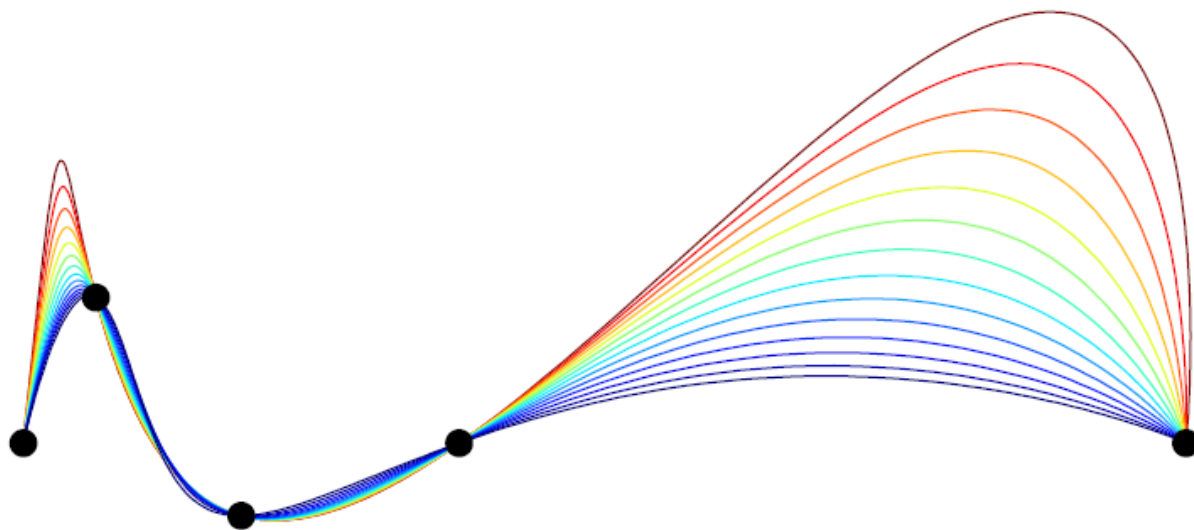


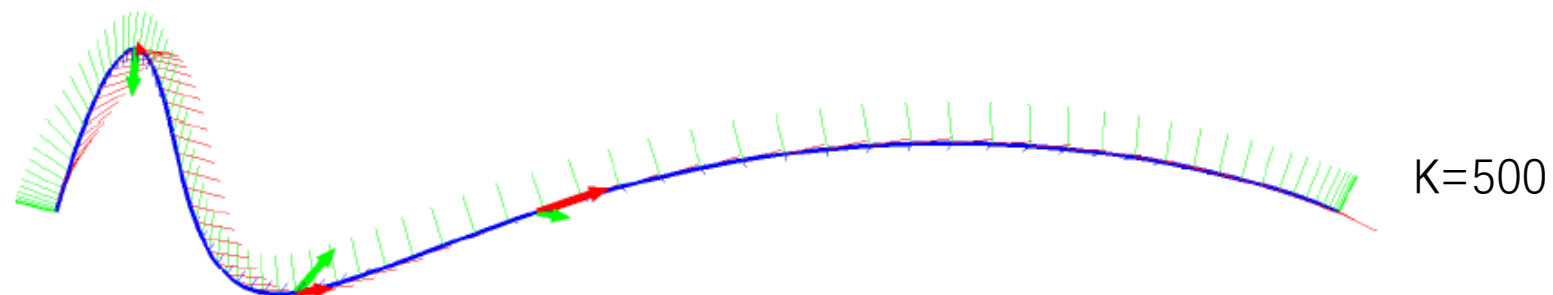


$$J_T = \begin{bmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_M \end{bmatrix}^T \begin{bmatrix} Q_1(T_1) & & \\ & \ddots & \\ & & Q_M(T_M) \end{bmatrix} \begin{bmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_M \end{bmatrix}^T + k_T \sum_{i=1}^M T_i$$

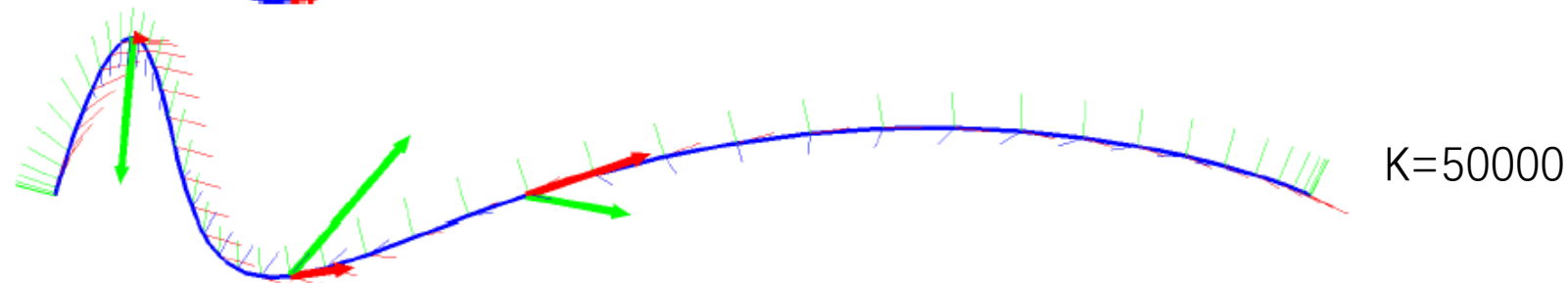
惩罚Cost中的时间

- 最小化目标函数 J
- 从数值上得到 T 的梯度



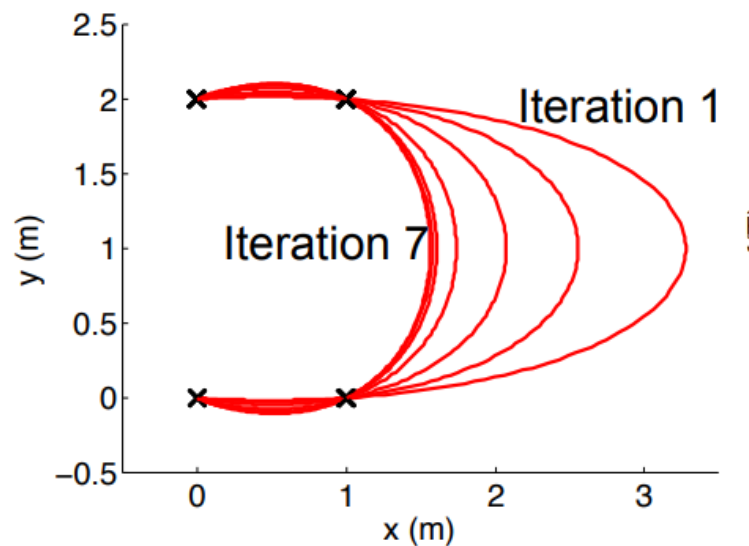


- 调整轨迹段间的时间比例
- 调整总时间



动力学可行性?

- 在没有约束的情况下，寻找最佳的时间分配，固定比例。
- 通过保持最佳比例来调整总轨迹时间。
- 直到出现约束为止。



- 调整轨迹段间的时间比例
- 固定总时间