# 基于二次规划的参考线拟合

## 1. 背景：

- 当前使用的是3阶贝塞尔曲线进行参考线的拟合，但是曲率的效果不好且求解较慢。
- 使用分段五次多项式曲线（参数曲线）平滑参考线。使其构造成二次规划（QP）形式，使用osqp求解。
    - 参数曲线的优势：能够更快地处理投影，Frenet变换
- QP标准形式如下：

$$\begin{array}{ll} \text{minimize} & \frac{1}{2} \cdot x^T \cdot P \cdot x + q^T \cdot x \\ \text{s.t.} & LB \leq x \leq UB \\ & A_{eq} x = b_{eq} \\ & Ax \geq b \end{array}$$

    - 特殊地，等式约束可以化为两个不等式约束使用osqp求解。
- 在本例中，构建的凸二次优化问题：

$$\begin{array}{ll} \text{minimize} & \frac{1}{2} \cdot x^T \cdot P \cdot x + q^T \cdot x \\ \text{s.t.} & l \leq Ax \leq u \end{array}$$

  其中， P: objective_matrix， q: objective_vector

  A: constraint_matrix，l: lower_bounds, u: upper_bounds

    - 如果P是半正定矩阵，那么目标函数是一个凸函数。如果有至少一个向量x满足约束而且f(x)在可行域有下界，二次规划问题就有一个全局最小值x。
    - 如果P是正定矩阵，那么全局最小值就是唯一的。如果P=0，二次规划问题就变成线性规划问题。
    - 正定矩阵的判定方式：正定矩阵的判定
        - 各阶顺序主子式全都大于0
        - 所有特征值都大于0
    - 半正定矩阵：所有特征值都大于等于0

## 2. 理论说明及名词解释：

- $S : [0, s_{max}] \to \mathbb{R}$
- 将 $S$ 分成等分的区间：

$$\begin{aligned} & [t_i, t_{i+1}], i = 0, \ldots, k-1 \\ & [0, s_{max}] = [t_0, t_1) \cup [t_1, t_2) \cup \cdots \cup [t_{k-2}, t_{k-1}) \cup [t_{k-1}, t_k) \cup [t_k] \\ & 0 = t_0 \leq t_1 \leq \cdots \leq t_{k-1} \leq t_k = s_{max} \end{aligned}$$

- 对于每一段pieces，我们定义一个多项式：

$$P_i : [t_i, t_{i+1}] \to \mathbb{R}$$

- 对于第i个子区间， $S$ 被定义为 $P_i$：

$$\begin{aligned} &S(t) = P_0(t), t_0 \le t < t_1 \\ &S(t) = P_1(t), t_1 \le t < t_2 \\ &\vdots \\ &S(t) = P_{k-1}(t), t_{k-1} \le t \le t_k \end{aligned}$$

- **节点（knot）**指的是将区间划分为一段一段的分段点。**节点向量（knot vector）** $\mathbf{t} = (t_0, \ldots, t_k)$ 是由多个节点组成的向量，代表着对于样条是如何进行分段的。
- 如果多项式最大的次数为n，定义**degree为n，或者order为n+1**。
- 同时定义**连续性参数向量 smoothness vector**, 即：

$$\begin{aligned} &P_{i-1}^{(0)}(t) = P_i^{(0)}(t) \\ &P_{i-1}^{(1)}(t) = P_i^{(1)}(t) \\ &\vdots \\ &P_{i-1}^{(r_i)}(t) = P_i^{(r_i)}(t) \end{aligned}$$

  对于这样的vector， $\mathbf{r} = (r_1, \ldots, r_{k-1})$ 定义为smoothness vector

## 3. 本例数学描述

- 基于五次多项式描述每段参考线

$$\begin{aligned} P(t) : x &= \frac{a_0}{5!}(t-t_i)^5 + \frac{a_1}{4!}(t-t_i)^4 + \frac{a_2}{3!}(t-t_i)^3 + \frac{a_3}{2!}(t-t_i)^2 + a_4(t-t_i) + a_5 \\ &= \sum_{j=0}^{n\_deg} a_j \cdot \frac{(t-t_i)^{n\_deg - j}}{(n\_deg - j)!}, \ t_i \le t < t_{i+1} = A^T S \end{aligned}$$

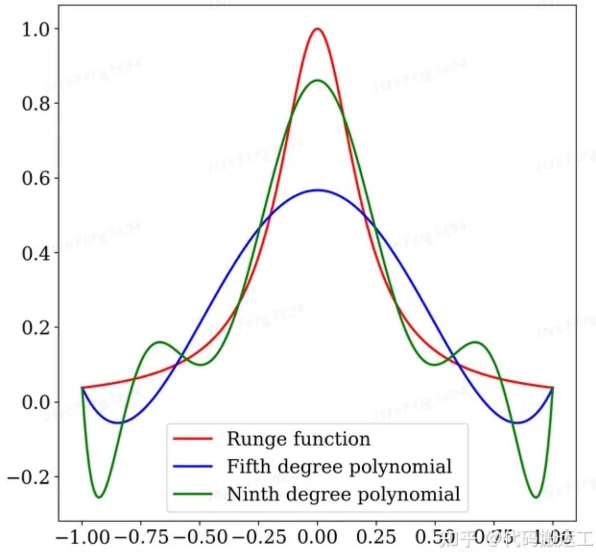
其中： $A = [a0, a1, a2, a3, a4, a5]^T$ ，
$S = [\frac{(t-t_i)^5}{5!}, \frac{(t-t_i)^4}{4!}, \frac{(t-t_i)^3}{3!}, \frac{(t-t_i)^2}{2!}, (t-t_i), 1]^T$

对于y也有同样的描述：

$$P(t) : y = \sum_{j=0}^{n\_deg} b_j \cdot \frac{(t - t_i)^{n\_deg - j}}{(n\_deg - j)!}, \ t_i \le t < t_{i+1} = B^T S$$

其中： $B = [b0, b1, b2, b3, b4, b5]^T$ ， $S = [\frac{(t-t_i)^5}{5!}, \frac{(t-t_i)^4}{4!}, \frac{(t-t_i)^3}{3!}, \frac{(t-t_i)^2}{2!}, (t-t_i), 1]^T$

- 为什么要使用分段五次多项式：




- 在上图中，共有9个点。通过一条九次多项式可以将所有点进行拟合。但是使用高次多项式拟合会出现典型的**龙格现象**（在一组等间插值点上使用具有高次多项式的多项式插值时出现的**区间边缘处的振荡问题，**如图中的绿线）
- 使用分段五次多项式，分段可以较为精确地拟合复杂道路，例如直角弯，U型弯，S弯等。五次多项式也能保证拟合曲线的高阶信息的平滑性

- 思考：根据apollo 参考线平滑的设计，可以对每段曲线的自变量归一化处理 $t \in [0,1]$ , 可能会简化计算。

## 4. 优化变量

需要优化求解的就是多项式系数向量 $A, B$ . 假设共有k段样条，需要优化变量数目是2*6*k

$$X=\left[a_0^0, a_1^0, \ldots a_5^0, a_0^1, a_1^1, \ldots a_5^1, \ldots b_0^0, b_1^0, \ldots b_5^0, b_0^1, b_1^1, \ldots, b_5^1, \ldots\right]^T$$

## 5. cost function的构建

$$\sum_{s=0.0}^{s_{max}}((x-x_i)^2+(y-y_i)^2)+\sum_{i=0}^{k-1}(\sum_{j=0}^{n_{regulator}}(w_j a_j)^2+\sum_{j=0}^{n_{regulator}}(w_j b_j)^2)$$

对平方展开，这里只列出x坐标，y坐标格式相同：

$$(x-x_i)^2=x^2-2xx_i+x_i^2=A^T S S^T A-2A^T Sxi+x_i^2$$

去掉常数项 $x_i^2$ ，化为标准形式，则：

- objective_matrix

$$P \;=\; SS^T \;=$$

$$\begin{bmatrix} \frac{(t-t_i)^5}{5!}\cdot\frac{(t-t_i)^5}{5!}, & \frac{(t-t_i)^5}{5!}\cdot\frac{(t-t_i)^4}{4!}, & \frac{(t-t_i)^5}{5!}\cdot\frac{(t-t_i)^3}{3!}, & \frac{(t-t_i)^5}{5!}\cdot\frac{(t-t_i)^2}{2!}, & \frac{(t-t_i)^5}{5!}\cdot(t-t_i), & \frac{(t-t_i)^5}{5!} \\ \frac{(t-t_i)^4}{4!}\cdot\frac{(t-t_i)^5}{5!}, & \frac{(t-t_i)^4}{4!}\cdot\frac{(t-t_i)^4}{4!}, & \frac{(t-t_i)^4}{4!}\cdot\frac{(t-t_i)^3}{3!}, & \frac{(t-t_i)^4}{4!}\cdot\frac{(t-t_i)^2}{2!}, & \frac{(t-t_i)^4}{4!}\cdot(t-t_i), & \frac{(t-t_i)^4}{4!} \\ \frac{(t-t_i)^3}{3!}\cdot\frac{(t-t_i)^5}{5!}, & \frac{(t-t_i)^3}{3!}\cdot\frac{(t-t_i)^4}{4!}, & \frac{(t-t_i)^3}{3!}\cdot\frac{(t-t_i)^3}{3!}, & \frac{(t-t_i)^3}{3!}\cdot\frac{(t-t_i)^2}{2!}, & \frac{(t-t_i)^3}{3!}\cdot(t-t_i), & \frac{(t-t_i)^3}{3!} \\ \frac{(t-t_i)^2}{2!}\cdot\frac{(t-t_i)^5}{5!}, & \frac{(t-t_i)^2}{2!}\cdot\frac{(t-t_i)^4}{4!}, & \frac{(t-t_i)^2}{2!}\cdot\frac{(t-t_i)^3}{3!}, & \frac{(t-t_i)^2}{2!}\cdot\frac{(t-t_i)^2}{2!}, & \frac{(t-t_i)^2}{2!}\cdot(t-t_i), & \frac{(t-t_i)^2}{2!} \\ (t-t_i)\cdot\frac{(t-t_i)^5}{5!}, & (t-t_i)\cdot\frac{(t-t_i)^4}{4!}, & (t-t_i)\cdot\frac{(t-t_i)^3}{3!}, & (t-t_i)\cdot\frac{(t-t_i)^2}{2!}, & (t-t_i)\cdot(t-t_i), & (t-t_i) \\ \frac{(t-t_i)^5}{5!}, & \frac{(t-t_i)^4}{4!}, & \frac{(t-t_i)^3}{3!}, & \frac{(t-t_i)^2}{2!}, & (t-t_i), & 1 \end{bmatrix}$$

- objective_vector：

$$q = S\cdot x_i = [\frac{(t-t_i)^5}{5!}\cdot x_i, \frac{(t-t_i)^4}{4!}\cdot x_i, \frac{(t-t_i)^3}{3!}\cdot x_i, \frac{(t-t_i)^2}{2!}\cdot x_i, (t-t_i)\cdot x_i, 1\cdot x_i]^T$$

- 正则化：
  - regulator是为了尽量让参数方程的高阶项减小。简单的在高阶项上增加了一个系数，防止过拟合。

## 6. 约束矩阵的构建

### 6.1 连续矩阵的构建

$$P_j^{(i)}(t)|_{t=t_{j\_end}} \;=\; P_{j+1}^{(i)}(t)|_{t=t_{j\_start}}\;(i\;<=\;num\_deg, j=0...k-2)$$

在本例中，考虑到的是到三阶连续

$$\begin{cases} x_j = \frac{a_{j-1}^0}{5!}(t_j-t_{j-1})^5 + \frac{a_{j-1}^1}{4!}(t_j-t_{j-1})^4 + \frac{a_{j-1}^2}{3!}(t_j-t_{j-1})^3 + \frac{a_{j-1}^3}{2!}(t_j-t_{j-1})^2 + a_{j-1}^4(t_j-t_{j-1}) + a \\ \qquad = \frac{a_j^0}{5!}(t_j-t_j)^5 + \frac{a_j^1}{4!}(t_j-t_j)^4 + \frac{a_j^2}{3!}(t_j-t_j)^3 + \frac{a_j^3}{2!}(t_j-t_j)^2 + a_j^4(t_j-t_j) + a_j^5, \\ x_j^{'} = \frac{a_{j-1}^0}{4!}(t_j-t_{j-1})^4 + \frac{a_{j-1}^1}{3!}(t_j-t_{j-1})^3 + \frac{a_{j-1}^2}{2!}(t_j-t_{j-1})^2 + a_{j-1}^3(t_j-t_{j-1}) + a_{j-1}^4 \\ \qquad = \frac{a_j^0}{4!}(t_j-t_j)^4 + \frac{a_j^1}{3!}(t_j-t_j)^3 + \frac{a_j^2}{2!}(t_j-t_j)^2 + a_j^3(t_j-t_j) + a_j^4, \\ x_j^{''} = \frac{a_{j-1}^0}{3!}(t_j-t_{j-1})^3 + \frac{a_{j-1}^1}{2!}(t_j-t_{j-1})^2 + a_{j-1}^2(t_j-t_{j-1}) + a_{j-1}^3 \\ \qquad = \frac{a_j^0}{3!}(t_j-t_j)^3 + \frac{a_j^1}{2!}(t_j-t_j)^2 + a_j^2(t_j-t_j) + a_j^3, \\ x_j^{'''} = \frac{a_{j-1}^0}{2!}(t_j-t_{j-1})^2 + a_{j-1}^1(t_j-t_{j-1}) + a_{j-1}^2 = \frac{a_j^0}{2!}(t_j-t_j)^2 + a_j^1(t_j-t_j) + a_j^2 \end{cases}$$

化为矩阵标准形式:

$$
\begin{bmatrix}
\frac{(t_j-t_{j-1})^5}{5!}, & \frac{(t_j-t_{j-1})^4}{4!}, & \frac{(t_j-t_{j-1})^3}{3!}, & \frac{(t_j-t_{j-1})^2}{2!}, & (t_j-t_{j-1}), & 1, & 0, & 0, & 0, & 0, & 0, & -1 \\
\frac{(t_j-t_{j-1})^4}{4!}, & \frac{(t_j-t_{j-1})^3}{3!}, & \frac{(t_j-t_{j-1})^2}{2!}, & (t_j-t_{j-1}), & 1, & 0, & 0, & 0, & 0, & 0, & -1, & 0 \\
\frac{(t_j-t_{j-1})^3}{3!}, & \frac{(t_j-t_{j-1})^2}{2!}, & (t_j-t_{j-1}), & 1, & 0, & 0, & 0, & 0, & 0, & -1, & 0, & 0 \\
\frac{(t_j-t_{j-1})^2}{2!}, & (t_j-t_{j-1}), & 1, & 0, & 0, & 0, & 0, & 0, & -1, & 0, & 0, & 0 \\
(t_j-t_{j-1}), & 1, & 0, & 0, & 0, & 0, & 0, & -1, & 0, & 0, & 0, & 0
\end{bmatrix}
\begin{bmatrix}
a_{j-1}^0 \\ a_{j-1}^1 \\ a_{j-1}^2 \\ a_{j-1}^3 \\ a_{j-1}^4 \\ a_{j-1}^5 \\ a_j^0 \\ a_j^1 \\ a_j^2 \\ a_j^3 \\ a_j^4 \\ a_j^5
\end{bmatrix}
=
$$

## 6.2 斜率，曲率约束

这里对s = 0.0初始点的斜率（1阶导）和曲率（2阶导）进行约束。

### 6.2.1 斜率

$$
P_0^{(1)}(t)|_{t=0.0} : \dot{x} = (x_1 - x_0)/s_1 = a_0^4
$$
$$
P_0^{(1)}(t)|_{t=0.0} : \dot{y} = (y_1 - y_0)/s_1 = b_0^4
$$

### 6.2.2 曲率

a. 三点确定曲率公式：已知三点分别是A,B, C 则借助于叉乘

$$
\kappa = \frac{\sin B}{0.5 * |AC|} = \frac{2 * |AB||BC| \sin B}{|AB||BC||AC|} = \frac{2 * \vec{AB} \times \vec{BC}}{|AB||BC||AC|}
$$

b. 初始点曲率计算

$$
\kappa = \frac{\dot{x}\ddot{y} - \ddot{x}\dot{y}}{\sqrt[1.5]{\dot{x}^2 + \dot{y}^2}}
$$

其中：$\dot{x}$，$\dot{y}$ 前面已求得，

$$
P_0^{(2)}(t)|_{t=0.0} : \ddot{x} = a_0^3
$$
$$
P_0^{(2)}(t)|_{t=0.0} : \ddot{y} = b_0^3
$$

但是代码里面 $\frac{1}{\sqrt[1.5]{\dot{x}^2 + \dot{y}^2}}$ 代码里面，这一块儿需要补充

以上，起始点的斜率，曲率约束已计算完毕，共三个等式，共同带入到约束矩阵里面即可。

在本例中，都是等式约束，因此 $l = Ax = u$

## 7. osqp求解器参数输入

参考：OSQP使用说明

## 补充：CSC矩阵

Compressed Sparse Column Format (CSC)的目的是为了压缩矩阵，减少矩阵存储所占用的空间。通过增加一些"元信息"来描述矩阵中的非零元素存储的位置(基于列)，然后结合非零元素的值来表示矩阵。这样在一些场景下可以减少矩阵存储的空间。

- 举例1：对于如下的3x3稀疏矩阵,可由3个数组表示：

$$\begin{bmatrix} 1 & 0 & 4 \\ 0 & 3 & 5 \\ 2 & 0 & 6 \end{bmatrix} => \begin{Bmatrix} p = [0,2,3,6] \\ i = [0,2,1,0,1,2] \\ x = [1,2,3,4,5,6] \end{Bmatrix}$$

    - i 数组表示每一列中，非0元素所在的行号，行号索引从0开始。
    - x 数组表示按照列输出依次记录的矩阵中的非0值。
    - p 数组的元素数目为列数+1， p[0]一定为0。p[i] 表示累计到第i-1列的非0值的数目。
        - p[1] = 2:表示第0列有2个非0值。对应的行号为i矩阵的前2个
        - p[2] = 3: 表示到第1列新增了1个非0值。
        - p的最后一个元素 = i数组和x数组的元素数目。即数组中非0元素的数目
- 举例2：对于如下的2x4稀疏矩阵

$$\begin{bmatrix} 9 & 0 & 0 & 0 \\ 0 & 8 & 6 & 5 \end{bmatrix} => \begin{Bmatrix} p = [0,1,2,3,4] \\ i = [0,1,1,1] \\ x = [9,8,6,5] \end{Bmatrix}$$

## 参考资料：


1. @HNP @CP @参考线拟合 @笛卡尔Frenet转换
2. Apollo ReferenceLine Smooth--多项式拟合（splines)曲线平滑原理
3. 理解Compressed Sparse Column Format (CSC)_weixin_34216036的博客-CSDN博客