

2 多项式及Python程序

2.1 线性插值（一阶，恒定速度）

线性插值(Linear Interpolation)，顾名思义，就是使用线性的方法来进行插值。即将给定的数据点依次用线段连起来，点与点之间运动的速度是恒定值。假设我们用 $q(t)$ 来表示插值以后的曲线，则用数学的方式来表示线性插值就是：

$$q(t) = a_0 + a_1(t - t_0), \quad t_0 \leq t \leq t_1 \quad (2.1)$$

其中， a_0, a_1 是待确定的常量参数。 t_0 表示初始时刻， a_0 表示初始时刻的位置， a_1 表示斜率，也就是速度，这里为常量。因此，给定下一个时刻 t_1 处的位置 $q(t_1)$ ，我们就有：

$$\begin{cases} q(t_0) = q_0 = a_0 \\ q(t_1) = q_1 = a_0 + a_1(t_1 - t_0) \end{cases} \quad (2.2)$$

可以计算得到两个常量参数：

$$\begin{cases} a_0 = q_0 \\ a_1 = (q_1 - q_0)/(t_1 - t_0) \end{cases} \quad t_0 \neq t_1 \quad (2.3)$$

2.2 三次多项式插值（三阶，加速度可变）

三次多项式插值方法（Cubic Polynomial Spline）是一种常用的插值方法，其位置和速度曲线是连续的，加速度是可变的，但加速度不一定连续。考虑2个数据点之间插值的情况，其数学表达式为：

$$\begin{cases} q(t) = a_0 + a_1(t - t_0) + a_2(t - t_0)^2 + a_3(t - t_0)^3 \\ \dot{q}(t) = a_1 + 2a_2(t - t_0) + 3a_3(t - t_0)^2 \\ \ddot{q}(t) = 2a_2 + 6a_3(t - t_0) \end{cases}, t_0 \leq t \leq t_f \quad (2.4)$$

其中， t_1, a_0, a_1, a_2 为待确定的参数。

考察给定2个数据点进行插值的情况，如果给定了在初始时刻 t_0 和最终时刻 t_f 处的位置与速度信息 $(q_0, v_0), (q_f, v_f)$ ，设 $h = q_f - q_0, T = t_f - t_0$ ，
初始位置和速度：

$$\begin{cases} q(t_0) = q_0 = a_0 \\ \dot{q}(t_0) = v_0 = a_1 \end{cases} \quad (2.5)$$

末端位置和速度：

$$\begin{cases} q(t_f) = q_f = a_0 + a_1T + a_2T^2 + a_3T^3 \\ \dot{q}(t_f) = v_f = a_1 + 2a_2T + 3a_3T^2 \end{cases} \quad (2.6)$$

结合公式 (2.5), (2.6), 则这些参数可以使用以下公式计算：

$$\begin{cases} a_0 = q_0 \\ a_1 = v_0 \\ a_2 = \frac{3h - (v_f + 2v_0)T}{T^2} = \frac{3}{T^2}(q_f - q_0) - \frac{1}{T}(v_f + 2v_0) \\ a_3 = \frac{-2h + (v_f + v_0)T}{T^3} = \frac{-2}{T^3}(q_f - q_0) + \frac{1}{T^2}(v_f + v_0) \end{cases} \quad (2.7)$$

对于给定 n 个一系列数据点进行插值的情况，只需要对所有相邻的两个数据点使用上述公式即可依次计算得到整条插值曲线，五次多项式会在后面补充。