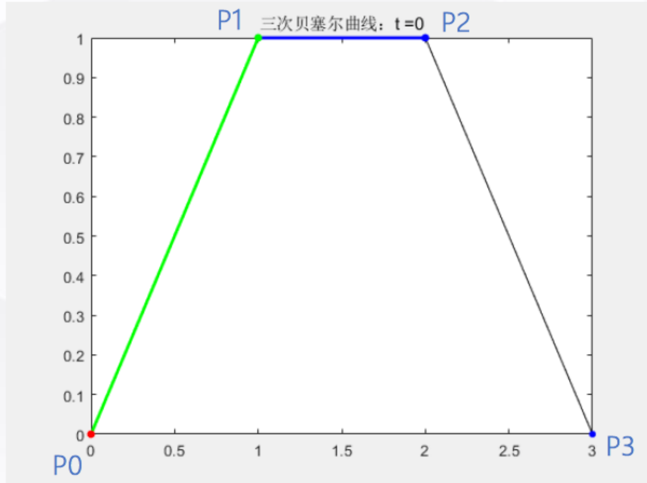
3 局部路径规划算法——贝塞尔曲线法

算法简介

- 贝塞尔曲线于1962年由法国工程师皮埃尔·贝塞尔（Pierre Bézier）所广泛发表，他运用贝塞尔曲线来为汽车的主体进行设计。
- 贝塞尔曲线是应用于二维图形应用程序的数学曲线，由一组称为控制点的向量来确定，给定的控制点按顺序连接构成控制多边形，贝塞尔曲线逼近这个多边形，进而通过调整控制点坐标改变曲线的形状。

算法思想

- 对于车辆系统，规划的轨迹应满足以下准则：轨迹连续；轨迹曲率连续；轨迹容易被车辆跟随，且容易生成；
- 给定n+1个数据点，p0~pn，生成一条曲线，使得该曲线与这些点描述的形状相符。

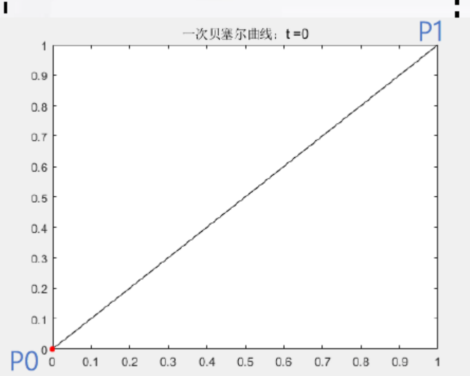


3 局部路径规划算法——贝塞尔曲线法

算法精讲——算法推导

- 设P0，P1两个控制点，t取值范围为[0,1]。则贝塞尔曲线生成点可以表达为：

$$p_1(t)=(1-t)P_0+tP_1$$

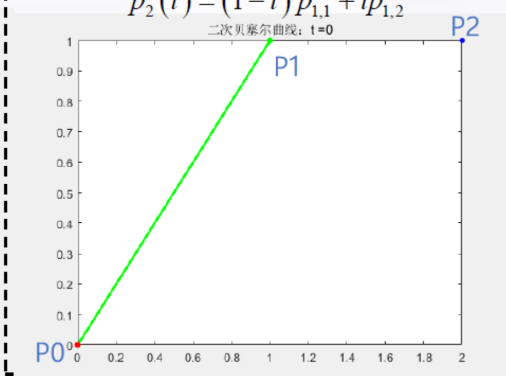


- 设P0，P1，P2三个控制点，t取值范围为[0,1]。
- P0和P1构成一阶，P1和P2也构成一阶，即：

$$\begin{cases} p_{1,1}(t)=(1-t)P_0+tP_1 \\ p_{1,2}(t)=(1-t)P_1+tP_2 \end{cases}$$

- 在生成的两个一阶点基础上，可以生成二阶贝塞尔点：

$$p_2(t)=(1-t)p_{1,1}+tp_{1,2}$$



- 设P0，P1，P2，P3个控制点，P0和P1、P1和P2、P2和P3都构成一阶，即：

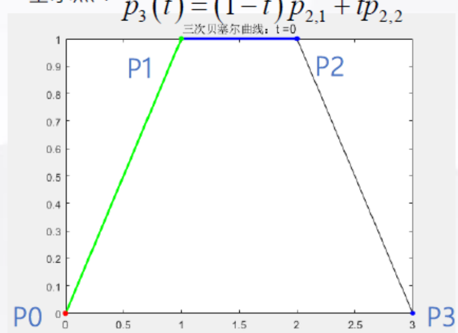
$$\begin{cases} p_{1,1}(t)=(1-t)P_0+tP_1 \\ p_{1,2}(t)=(1-t)P_1+tP_2 \\ p_{1,3}(t)=(1-t)P_2+tP_3 \end{cases}$$

- 在生成的三个一阶点基础上，可以生成两个二阶贝塞尔点：

$$\begin{cases} p_{2,1}(t)=(1-t)p_{1,1}+tp_{1,2} \\ p_{2,2}(t)=(1-t)p_{1,2}+tp_{1,3} \end{cases}$$

- 在生成的两个二阶点基础上，可以生成三阶贝塞尔点：

$$p_3(t)=(1-t)p_{2,1}+tp_{2,2}$$



3 局部路径规划算法——贝塞尔曲线法

算法精讲——算法推导

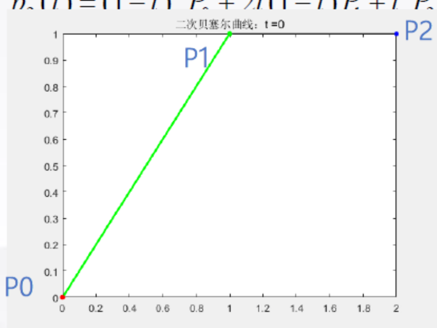
- 针对P0，P1，P2三个控制点而言，由以下三个递推式得到贝塞尔点

$$\begin{cases} p_{1,1}(t)=(1-t)P_0+tP_1 \\ p_{1,2}(t)=(1-t)P_1+tP_2 \end{cases}$$

$$p_2(t)=(1-t)p_{1,1}+tp_{1,2}$$

- 则贝塞尔点与3个控制点的关系为：

$$p_2(t)=(1-t)^2P_0+2t(1-t)P_1+t^2P_2$$



- 最多……阶导数不为常数

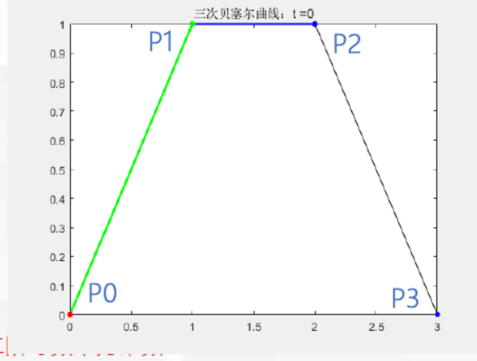
- 针对P0，P1，P2，P3四个控制点而言，
- 由以下三个递推式得到贝塞尔点

$$\begin{cases} p_{1,1}(t)=(1-t)P_0+tP_1 \\ p_{1,2}(t)=(1-t)P_1+tP_2 \\ p_{1,3}(t)=(1-t)P_2+tP_3 \end{cases} \quad \begin{cases} p_{2,1}(t)=(1-t)p_{1,1}+tp_{1,2} \\ p_{2,2}(t)=(1-t)p_{1,2}+tp_{1,3} \end{cases}$$

$$p_3(t)=(1-t)p_{2,1}+tp_{2,2}$$

- 则贝塞尔点与4个控制点的关系为：

$$p_3(t)=(1-t)^3P_0+3t(1-t)^2P_1+3t^2(1-t)P_2+t^3P_3$$



- 最多二……

3 局部路径规划算法——贝塞尔曲线法

算法精讲——算法推导

对于P0，P1，P2，…，Pn共n+1个控制点而言，贝塞尔点定义为：

$$p_n(t)=\sum_{i=0}^{n}C_n^i\cdot(1-t)^{n-i}\cdot t^i\cdot P_i=\sum_{i=0}^{n}\boxed{B_{i,n}(t)}\cdot P_i \qquad (i=0\text{时}, B_{i,n}=0)$$

伯恩斯坦基函数

伯恩斯坦基函数的一阶导数为：

$$B'_{i,n}(t)=\left[C_n^i\cdot(1-t)^{n-i}\cdot t^i\right]'$$

$$=\frac{n!}{i!\cdot(n-i)!}\cdot\left[-(n-i)\cdot(1-t)^{n-i-1}\cdot t^i+i\cdot t^{i-1}\cdot(1-t)^{n-i}\right]$$

$$=\frac{n!}{i!\cdot(n-i)!}\cdot i\cdot t^{i-1}\cdot(1-t)^{n-i}-\frac{n!}{i!\cdot(n-i)!}\cdot(n-i)\cdot(1-t)^{n-i-1}\cdot t^i$$

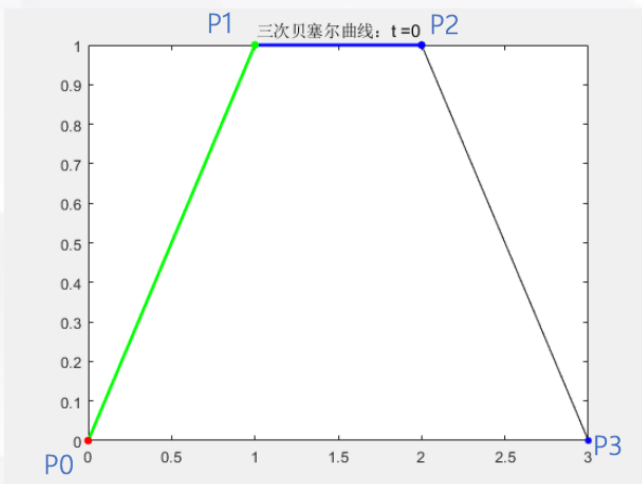
$$=n\frac{(n-1)!}{(i-1)!\cdot(n-i)!}t^{i-1}\cdot(1-t)^{n-i}-n\frac{(n-1)!}{i!\cdot(n-i-1)!}\cdot(1-t)^{n-i-1}\cdot t^i$$

$$=nC_{n-1}^{i-1}t^{i-1}\cdot(1-t)^{n-i}-nC_{n-1}^{i}(1-t)^{n-i-1}\cdot t^i=n\left[B_{i-1,n-1}(t)-B_{i,n-1}(t)\right]$$

数学描述

t=0时，p_n (t) =P0； t=1时，p_n (t) =Pn

性质1：P0和Pn分别位于贝塞尔曲线的起点和终点。
性质2：几何特性不随坐标系的变换而变化。



$p_n(t)=\sum_{i=0}^{n}C_n^i(1-t)^{n-i}t^iP_i \qquad B_{i,n}(t)=C_n^i(1-t)^{n-i}t^i$

当t=0时 $B_{0,n}(t)=C_n^0(1-t)^n0^0=1 \quad B_{i,n}=C_n^i(1-t)^n0^i=0$，($i\neq0$时)

$p_n(0)=p_0$ 同理 $p_n(1)=p_n$，性质1得证

3 局部路径规划算法——贝塞尔曲线法

Ally

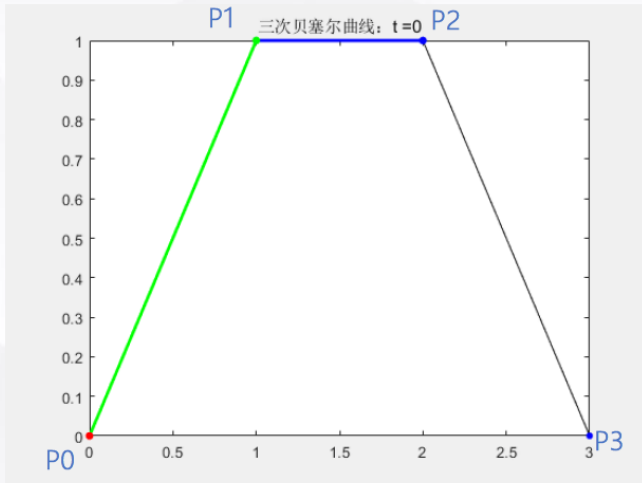
算法精讲——算法推导

$$p_n(t)=\sum_{i=0}^{n} C_n^i\cdot(1-t)^{n-i}\cdot t^i\cdot P_i=\sum_{i=0}^{n} B_{i,n}(t)\cdot P_i$$

$$B'_{i,n}(t)=n\left[B_{i-1,n-1}(t)-B_{i,n-1}(t)\right]$$

则贝塞尔点求导为：

$$\begin{aligned}
p'_n(t)&=\left[\sum_{i=0}^{n} C_n^i\cdot(1-t)^{n-i}\cdot t^i\cdot P_i\right]'\\
&=B'_{0,n}P_0+B'_{1,n}P_1+\cdots+B'_{n,n}\cdot P_n\\
&=0+n\left(B_{0,n-1}-B_{1,n-1}\right)P_1+n\left(B_{1,n-1}-B_{2,n-1}\right)P_2\cdots+n\left(B_{n-1,n-1}-B_{n,n-1}\right)P_n\\
&=n\left[B_{0,n-1}(P_1-P_0)+B_{1,n-1}(P_2-P_1)+\cdots+B_{n-1,n-1}(P_n-P_{n-1})\right]\\
&=n\sum_{i=1}^{n} B_{i-1,n-1}(t)\cdot(P_i-P_{i-1})
\end{aligned}$$



性质3：起点和终点处的切线方向与和特征多边形的第一条边及最后一条边分别相切

$p'_n(t) = n\sum_{r=1}^{n} B_{r-1,n-1}(t)\,(p_r - p_{r-1})$

t=0时 $p'_n(0) = nB_{0,n-1}(0)(p_1-p_0) + nB_{1,n-1}(0)(p_2-p_1) + \cdots$

$\downarrow \quad C_{n-1}^0(1-0)^{n-1}0^0 = 1$ $\qquad \downarrow \quad C_{n-1}^1(1-0)^{n-2}0^1 = 0$

$\Rightarrow p'_n(0) = n\cdot(p_1-p_0)$

Bezier为一个向量值函数，即自变量为标量，值为向量

→ 曲线导数方向即为切线方向

↓为

p_1-p_0 因此和第一条边相切

由Bezier定义可知，n+1个控制点的曲线阶数为n阶，点数越多阶数越大，不利于拟合，因此有分段Bezier曲线。分段Bezier如何保证接点处一阶连续，那一阶连续的条件是？下面给出推导。

设第一段Bezier曲线的倒数第二个点为p_{n-1}，倒数第一个点为p_n

第二段Bezier曲线的起始点为Q_0，第二个点为Q_1

$p_n'(t) = n\sum_{i=1}^{n} B_{i-1,n-1}(t)\,(p_i - p_{i-1})$，则$t=1$时有

$p_n'(1) = n(p_n - p_{n-1})$

$t=0$时有 $p_n'(0) = n(p_1 - p_0)$

设一段Bezier有n个控制点，二段有m个，则

一段：$p_n'(1) = n(p_n - p_{n-1})$ 　　二段 $Q_m'(0) = m(Q_1 - Q_0)$

要求 $n(p_n - p_{n-1}) = m(Q_1 - Q_0)$

又∵ $Q_0 = p_n$ $\Rightarrow$ $n(Q_0 - p_{n-1}) = m(Q_1 - Q_0)$

接缝满足位置连续和切向的一致即可满足视觉上平滑.