曲线曲面基本理论

计算机图形学三大块内容：光栅图形显示、几何造型技术、真实感图形显示。光栅图形学是图形学的基础，有大量的思想和算法

几何造型技术是一项研究在计算机中，如何表达物体模型形状的技术

描述物体的三维模型有三种:

线框模型、曲面模型和实体模型

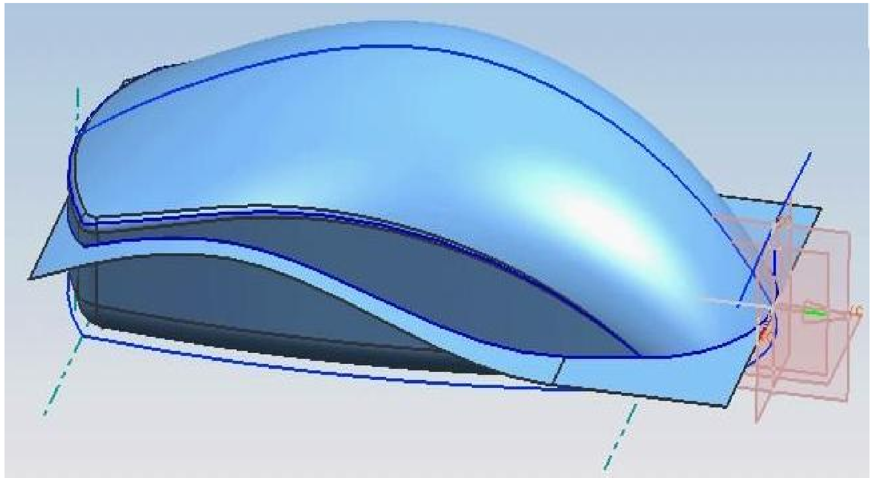
线框模型用顶点和棱边来表示物体

曲面模型只描述物体的表面和表面的连接关系，不描述物体内部的点的属性

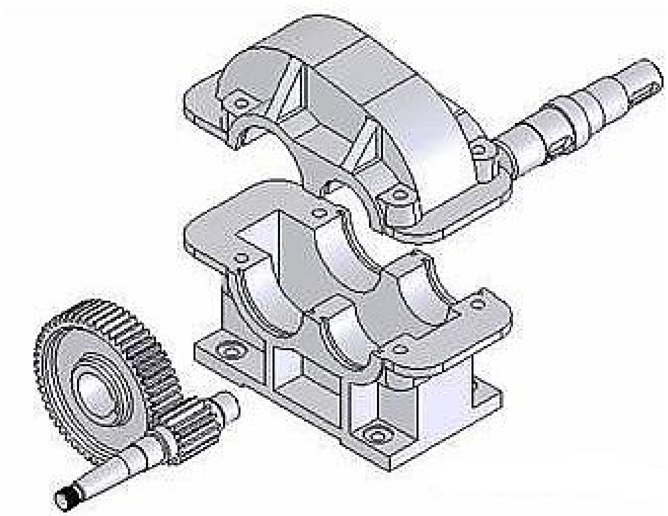
实体模型不但有物体的外观而且也有物体内点的描述

一、几何造型的历史

曲面造型：60年代，法国雷诺汽车公司的Pierre Bézier研发了汽车外形设计的UNISURF系统



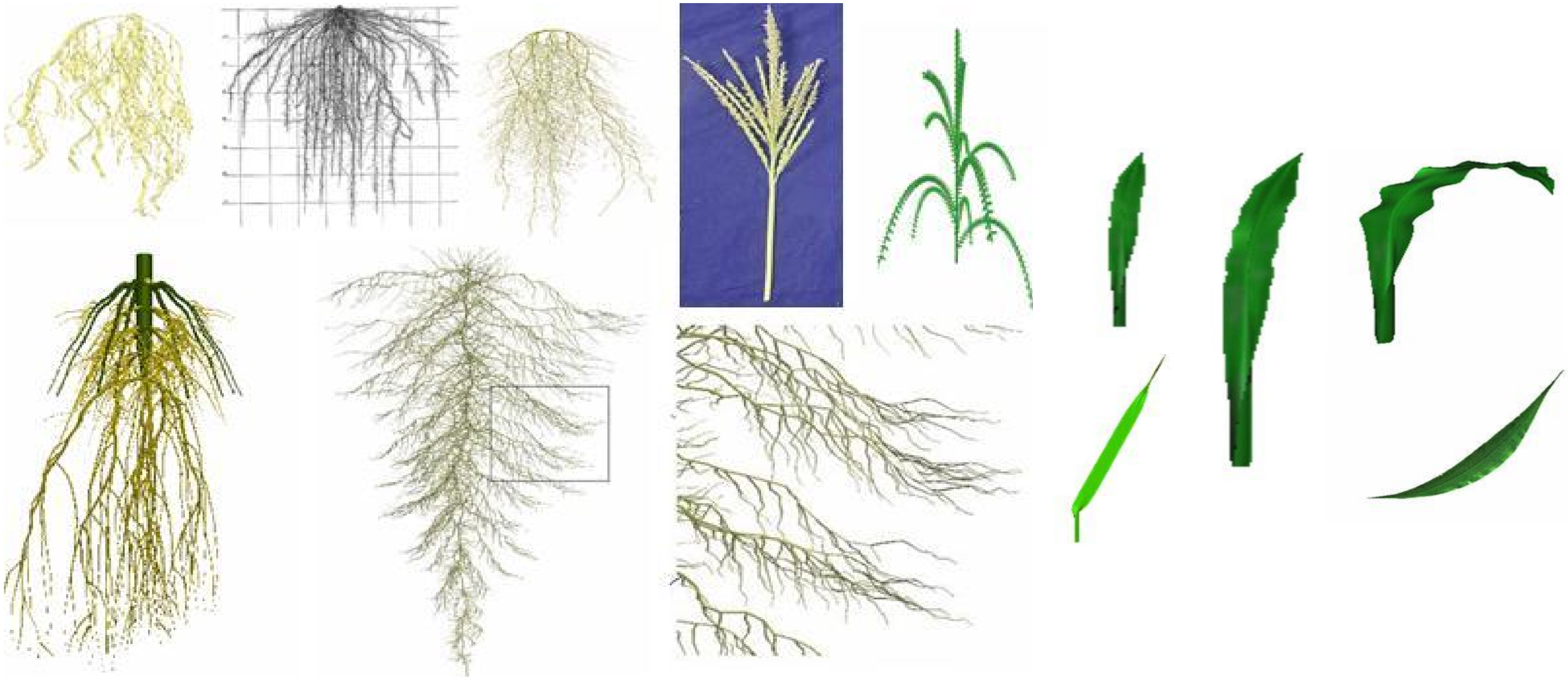
实体造型：1973英国剑桥大学CAD小组的Build系统、美国罗彻斯特大学的PADL-1系统等



– In 1962, Pierre Bézier, an engineer of French Renault Car company, propose a new kind of curve representation, and finally developed a system UNISURF for car surface design in 1972.

- Solid Modeling（实体造型）：
  - In1973, Ian Braid of Cambridge University developed a solid modeling system for designing engineering parts.
  - Ian Braid presented in his dissertation “Designing with volumes”, this work being demonstrated with the BUILD-1 system.


Super-7



二、曲线曲面基础

1、显示、隐式和参数表示

曲线和曲面的表示方程有参数表示和非参数表示之分，非参数表示又分为显式表示和隐式表示。

对于一个平面曲线，显式表示一般形式是：$y = \mathrm{f}(\mathrm{x})$

$$y = f(x)$$

在此方程中，一个x值与一个y值对应，所以显式方程不能表示封闭或多值曲线

如果一个平面曲线方程，表示成 f（x，y）= 0 的形式，称之为隐式表示。隐式表示的优点是易于判断一个点是否在曲线上

2、显式或隐式表示存在的问题

（1）与坐标轴相关

（2）用隐函数表示不直观，作图不方便

（3）用显函数表示存在多值性

（4）会出现斜率为无穷大的情形

3、参数方程

为了克服以上问题，曲线曲面方程通常表示成参数的形式，假定用t表示参数，平面曲线上任一点P可表示为：

$$p(t) = [x(t), y(t)]$$

空间曲线上任一三维点P可表示为：

$$p(t) = [x(t), y(t), z(t)]$$

它等价于笛卡儿分量表示：

$$p(t) = x(t)i + y(t)j + z(t)k$$

这样，给定一个t值，就得到曲线上一点的坐标

假设曲线段对应的参数区间为[a,b]，即a≤t≤b。为方便期间，可以将区间[a,b]规范化成[0,1]，参数变换为：

$$t' = \frac{t - a}{b - a}$$

参数曲线一般可写成：

$$p = p(t) \qquad t \in [0,1]$$

$$p = p(t) \qquad t \in [0,1]$$

该形式把曲线上表示一个点的位置矢量的各个分量合写在一起当成一个整体，考虑的是曲线上点之间的相对位置关系而不是它们与所取坐标系之间的相对位置关系

类似地，可把曲面表示成为双参数u和v的矢量函数

$$p(u,v) = p(x(u,v), y(u,v), z(u,v)) \quad (u,v) \in [0,1] \times [0,1]$$

最简单的参数曲线是直线段，端点为P_1、P_2的直线段参数方程可表示为：

$$p(t) = p_1 + (p_2 - p_1)t \qquad t \in [0,1]$$

4、参数方程的优势

在曲线、曲面的表示上，参数方程比显式、隐式方程有更多的优越性，主要表现在：

（1）可以满足几何不变性的要求

即指形状的数学表示及其所表达的形状不随所取坐标系而改变的性质

（2）有更大的自由度来控制曲线、曲面的形状

$$y = ax^3 + bx^2 + cx + d$$

只有四个系数控制曲线的形状。而二维三次曲线的参数表达式为：

$$p(t) = \begin{bmatrix} a_1 t^3 + a_2 t^2 + a_3 t + a_4 \\ b_1 t^3 + b_2 t^2 + b_3 t + b_4 \end{bmatrix} \qquad t \in (0,1)$$

有8个系数可用来控制此曲线的形状

（3）直接对参数方程进行几何变换

对非参数方程表示的曲线、曲面进行变换，必须对曲线、曲面上的每个型值点进行几何变换；而对参数表示的曲线、曲面可对其参数方程直接进行几何变换

（4）便于处理斜率为无穷大的情形，不会因此而中断计算

（5）界定曲线、曲面的范围十分简单

具有规格化的参数变量$t \in [0,1]$

（6）易于用向量（矢量）和矩阵运算，简化计算

5、参数曲线的基本概念

这部分内容完全来自于微分几何。微分几何是用微分的方法来研究曲线的局部性质，如曲线的弯曲程度等

一条用参数表示的三维曲线是一个有界点集，可写成一个带参数的、连续的、单值的数学函数，其形式：

$$\begin{cases} x = x(t) \\ y = y(t) \\ z = z(t) \end{cases} \qquad 0 \le t \le 1$$

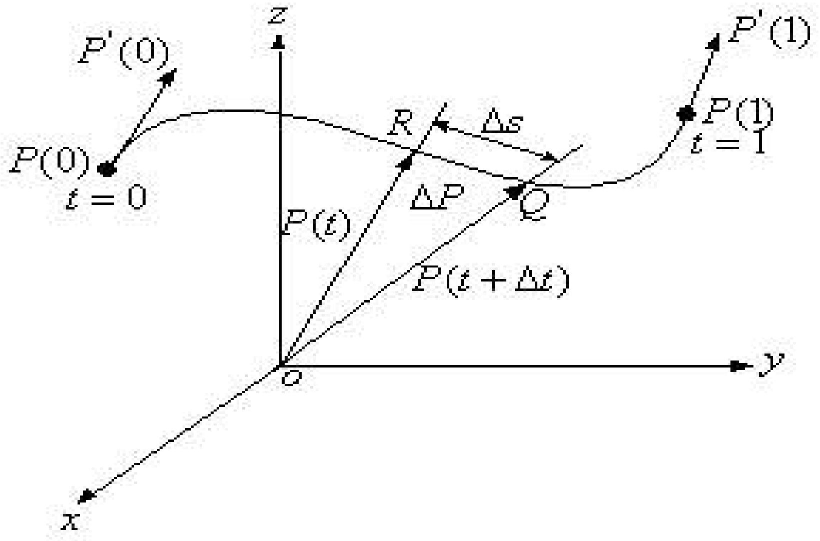
$$p'(t) = \frac{dP}{dt} \qquad p''(t) = \frac{d^2P}{dt^2}$$

（1）位置矢量

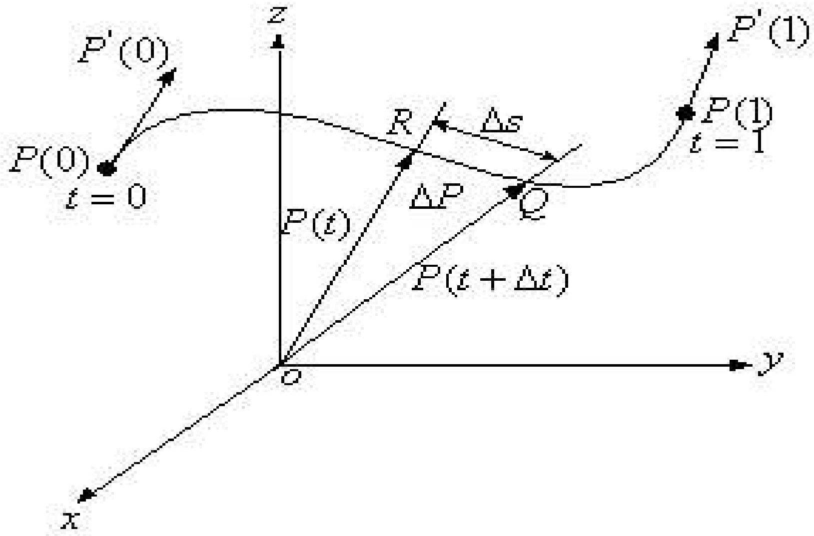
曲线上任一点的位置矢量

可表示为：

$$p(t) = [x(t), y(t), z(t)]$$



（2）切矢量



选择弧长s作为参数，当△t➞0时，弦长△s➞0，但方向不能趋向于0

$$T = \frac{dP}{ds} = \lim_{\Delta s \to 0} \frac{\Delta P}{\Delta s}$$ 单位矢量

根据弧长微分公式有：

$$(ds)^2 = (dx)^2 + (dy)^2 + (dz)^2$$

引入参数t，可改写为：

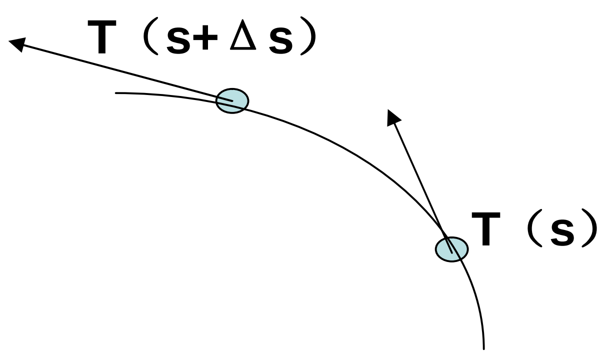
$$\left(ds/dt\right)^2=\left(dx/dt\right)^2+\left(dy/dt\right)^2+\left(dz/dt\right)^2=\left|P'(t)\right|^2$$

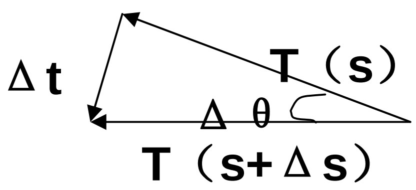
$$T=\frac{dP}{ds}=\frac{dP}{dt}\cdot\frac{dt}{ds}=\frac{P'(t)}{\left|P'(t)\right|}$$

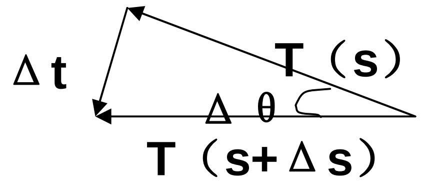
即T是单位切矢量

（3）曲率

切向量求导，求导以后还是一个向量，称为曲率，其几何意义是曲线的单位切向量对弧长的转动率，即刻画这一点的曲线的弯曲程度。







$$k = \left|T'\right| = \lim_{\Delta s \to 0}\left|\frac{\Delta T}{\Delta s}\right| = \lim_{\Delta s \to 0}\left|\frac{T(s+\Delta s)-T(s)}{\Delta s}\right| = \lim_{\Delta s \to 0}\left|\frac{\Delta q}{\Delta s}\right|$$

曲率越大，表示曲线的弯曲程度越大

曲率k的倒数 $r = \dfrac{1}{k}$ 称为曲率半径

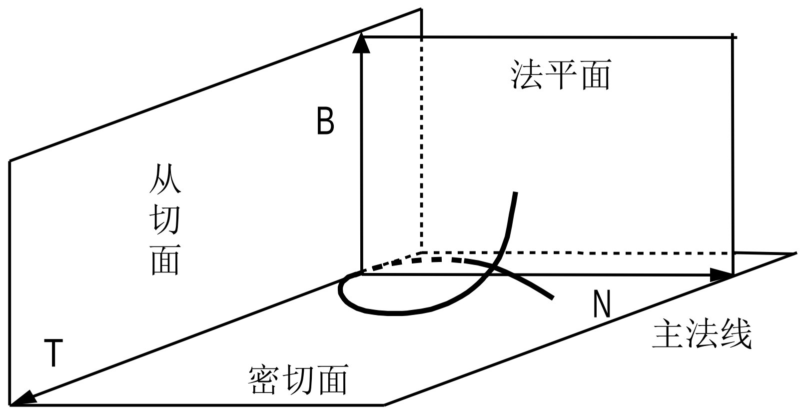
曲率半径越大，圆弧越平缓

曲率半径越小，圆弧越陡

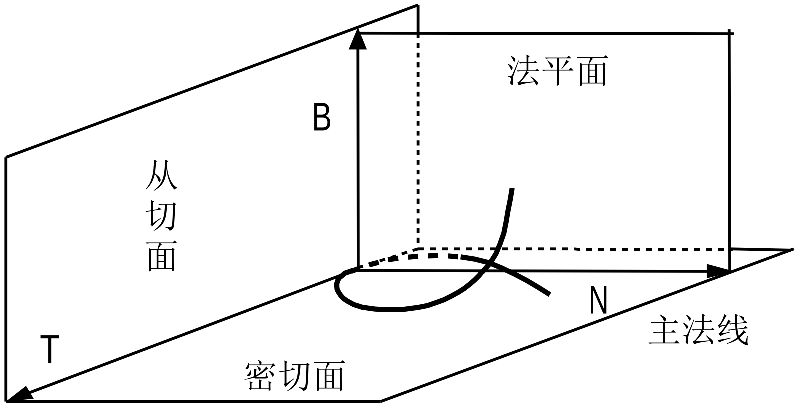
（4）法矢量

法矢量是与切矢量垂直的向量

N、B 构成的平面称为法平面，N、T 构成的平面称为密切平面，B、T 构成的平面称为从切平面



T(切矢)、N(主法矢)和B(副法矢)构成了曲线上的活动坐标架



$$B = T \times N$$

（5）挠率

空间曲线不但要弯曲，而且还要扭曲，即要离开它的密切平面。为了能刻画这一扭曲程度，等价于去研究密切平面的法矢量（即曲线的副法矢量）关于弧长的变化率。

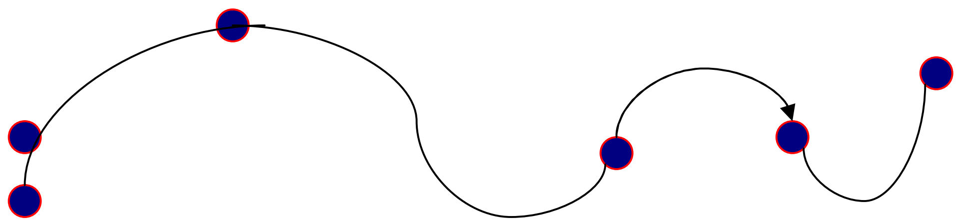
挠率 t 的绝对值等于副法线方向(或密切平面)对于弧长的转动率：

$$|t| = \lim_{\Delta s} \left|\frac{\Delta q}{\Delta s}\right|$$

6、插值

自由曲线和自由曲面一般通过少数分散的点生成，这些点叫做“型值点”、“样本点”或“控制点”

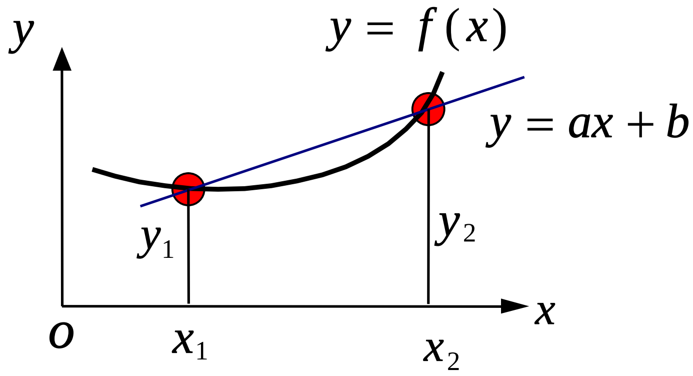
给定一组有序的数据点P_i(i=0,1,2,…n)，要求构造一条曲线顺序通过这些数据点，称为对这些数据点进行插值（interpolation），所构造的曲线称为插值曲线



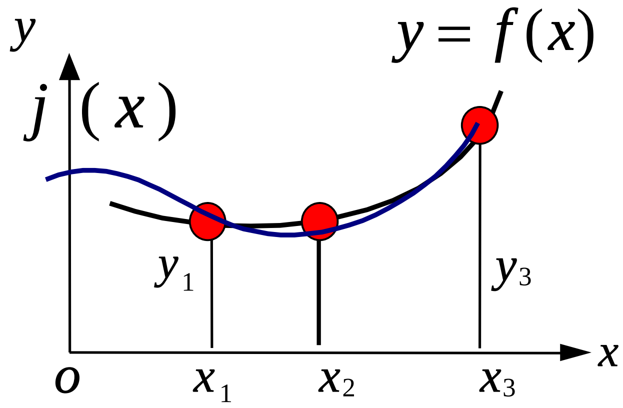
把曲线插值推广到曲面，类似地就有插值曲面

构造插值曲线曲面所采用的数学方法称为曲线曲面插值法

(1) 线性插值：假设给定函数f(x)在两个不同点x_1和x_2的值，用一个线形函数：y=ax+b，近似代替，称为的线性插值函数



（2）抛物线插值：已知三个点的坐标，要求构造一个抛物线函数



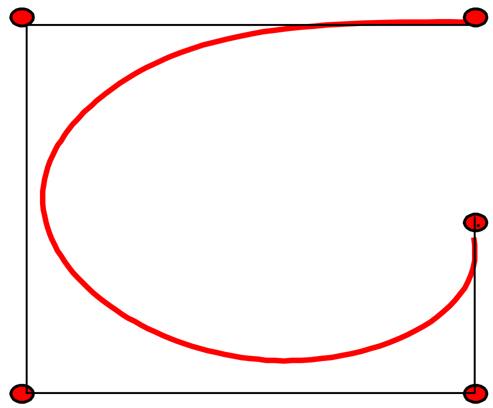
$$j\ (x) = ax^{2} + bx + c$$

7、拟合

构造一条曲线使之在某种意义下最接近给定的数据点(但未必通过这些点)，所构造的曲线为拟合曲线

在计算数学中，逼近通常指用一些性质较好的函数近似表示一些性质不好的函数。在计算机图形学中，逼近继承了这方面的含义，因此插值和拟合都可以视为逼近

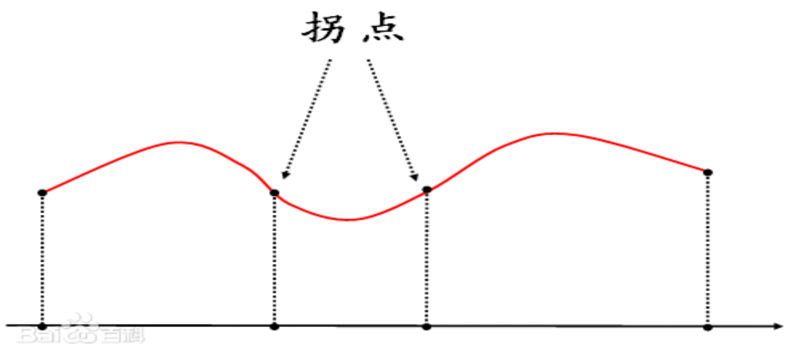
对于逼近样条，连接控制点序列的折线通常被显示出来，以提醒设计者控制点的次序



一般将连接有一定次序控制点的直线序列称为**控制多边形**或**特征多边形**

7、光顺

指曲线的拐点不能太多（有一、二阶导数等）



在数学领域是指：凸曲线与凹曲线的连接点

对平面曲线而言，相对光顺的条件是:

a．具有二阶几何连续性(G^2)

b．不存在多余拐点和奇异点

c．曲率变化较小

8、连续性

当许多参数曲线段首尾相连构成一条曲线时，如何保证各曲线段在连接处具有合乎要求的连续性是一个重要问题。假定参数曲线段p_i以参数形式进行描述：

$$p_i = p_i(t) \qquad t \in [t_{i0}, t_{i1}]$$

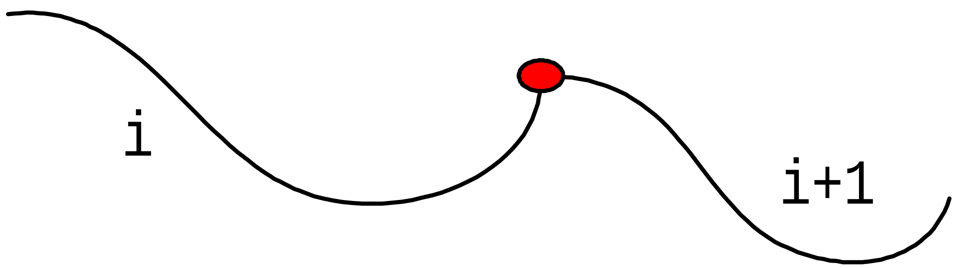
这里讨论参数曲线两种意义上的连续性：即参数连续性和几何连续性

（1）参数连续性

0阶参数连续性：

记作C^0连续性，是指曲线的几何位置连接，即第一个曲线段在t_{i1}处的x,y,z值与第二个曲线段在$t_{(i+1)0}$处的x,y,z值相等：

$$p_i(t_{i1}) = p_{(i+1)}(t_{(i+1)0})$$



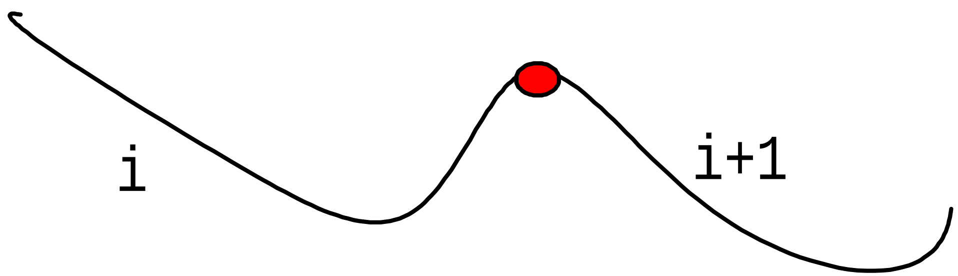
1阶参数连续性：

记作C^1连续性，指代表两个相邻曲线段的方程在相交点处有相同的一阶导数（切线）：

$$p_i(t_{i1}) = p_{(i+1)}(t_{(i+1)0})$$

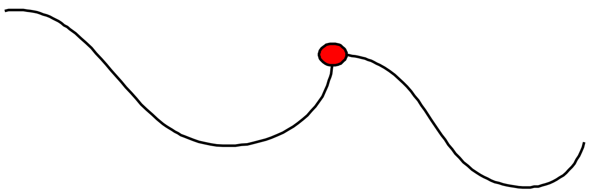
$$\text{且}\quad p_i'(t_{i1}) = p_{(i+1)}'(t_{(i+1)0})$$

一阶连续性对数字化绘画及一些设计应用已经足够

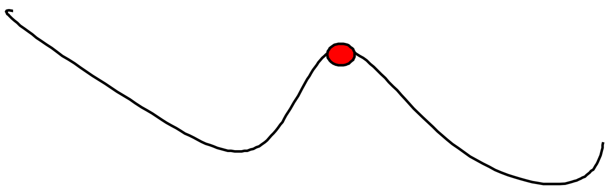


2阶参数连续性：

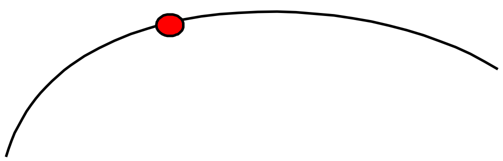
记作C^2连续性，指两个相邻曲线段的方程在相交点处具有相同的一阶和二阶导数。类似地，还可定义高阶参数连续性



(a)0阶连续性



(b)1阶连续性



(c)2阶连续性

对于C^2连续性，交点处的切向量变化率相等，即切线从一个曲线段平滑地变化到另一个曲线段

二阶连续性对电影中的动画途径和很多精密CAD需求有用

经典的参数连续性在图形学里是不适合的，因为太苛刻，所以引进了几何连续性的概念

汽车曲面的设计美观要求很高，但有时候车身的一条曲线并不是参数连连续的，但人眼看上去已经是很光滑的了，因此需要一种更弱的连续性

$$\Phi(t)=\begin{cases} V_0+\dfrac{V_1-V_0}{3}t, & 0\le t\le 1 \\ V_0+\dfrac{V_1-V_0}{3}+(t-1)\dfrac{2(V_1-V_0)}{3}, & 1\le t\le 2 \end{cases}$$

$$\Phi'(1^-)=\frac{1}{3}(V_1-V_0) \qquad \Phi'(1^+)=\frac{2}{3}(V_1-V_0)$$

$$\Phi'(1^-) = \frac{1}{3}(V_1 - V_0) \qquad \Phi'(1^+) = \frac{2}{3}(V_1 - V_0)$$

上面的函数实际上是直线方程。但可以发现，在t=1这一点，直线的左右导数不相等

在微积分里，如果一个函数的在一点处它的左导数和右导数都存在并且相等，就说明在这一点是连续的

（2）几何连续性

曲线段相连的另一个连续性条件是几何连续性。与参数连续性不同的是，它只需曲线段在相交处的参数导数成比例即可

0阶几何连续性：记作G^0连续性。与0阶参数连续性的定义相同，满足：

$$p_i(t_{i1}) = p_{(i+1)}(t_{(i+1)0})$$

1阶几何连续性，记作G^1连续性。若要求在结合处达到G^1连续，就是说两条曲线在结合处在满足G0连续的条件下，并有公共的切矢

$$Q'(0) = aP'(1) \qquad (a > 0)$$

2阶几何连续性，记作G^2连续性。就是说两条曲线在结合处在满足G^1 连续的条件下，并有公共的曲率

一阶导数相等和有公共切向量这两个概念差别是什么？导数相等是大小方向都相等，而公共切矢意味着方向相同但大小不等

所谓参数连续意味着导数相等，导数相等意味着两个切向量不但方向相等而且长度也相等

如果是几何连续的话，只是要求切向量一样，方向一样，长度可以不同。条件减弱了

9、参数化

过三点P_0、P_1和P_2构造参数表示的插值多项式是唯一的还是有多个呢？（设置一个课间提问）

插值多项式可以有无数条，这是因为对应地参数t在[0，1]中可以有无数种取法

$$t_0 = 0, t_1 = \frac{1}{2}, t_2 = 1 \qquad t_0 = 0, t_1 = \frac{1}{3}, t_2 = 1$$

参数方程：
$$x(t) = a_1 t^2 + a_2 t + a_3$$
$$y(t) = b_1 t^2 + b_2 t + b_3$$

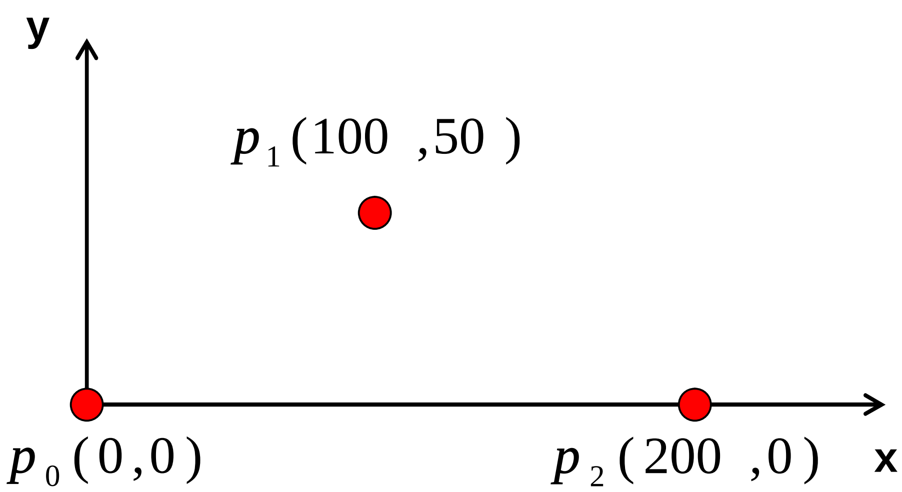
插值问题实际上就是解方程组的问题。但如果参数取的不一样的话，结果是不一样的

每个参数值称为节点（knot）。对于一条插值曲线，p_0、p_1、p_2这些点称为型值点

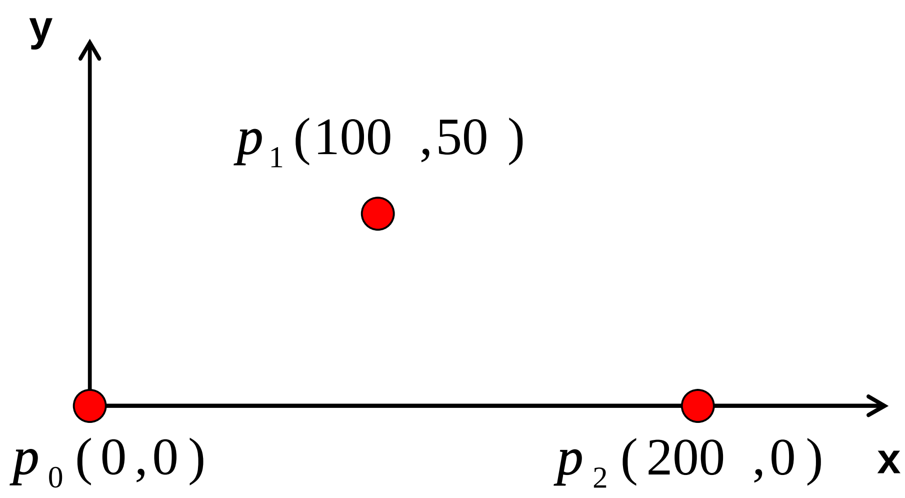
对于一条插值曲线，型值点p_0，p_1，… ，p_n 与其参数域 $t\in[t0, t1]$ 内的节点之间有一种对应关系。对于一组有序的型值点，所确定一种参数分割，称之这组型值点的参数化

现在给定3个点p_0、p_1、p_2，坐标是（0，0）、（100，50）、（200，0），求一条2次的多项式曲线来插值这三个点

假设第一个点参数t = 0，
第二个点参数取t = 1/2，
第三个点的参数取t = 1。
如何列这个方程？



$$t_1 = 0 \quad t_2 = \frac{1}{2} \quad t_3 = 1$$



$$x(t) = a_1 t^2 + a_2 t + a_3$$

$$y(t) = b_1 t^2 + b_2 t + b_3$$

$$0 = x(0) = a_3$$

$$0 = y(0) = b_3$$

$$100 = x(\frac{1}{2}) = \frac{a_1}{4} + \frac{a_2}{2} + a_3$$

$$50 = y(\frac{1}{2}) = \frac{b_1}{4} + \frac{b_2}{2} + b_3$$

$$200 = x(1) = a_1 + a_2 + a_3$$

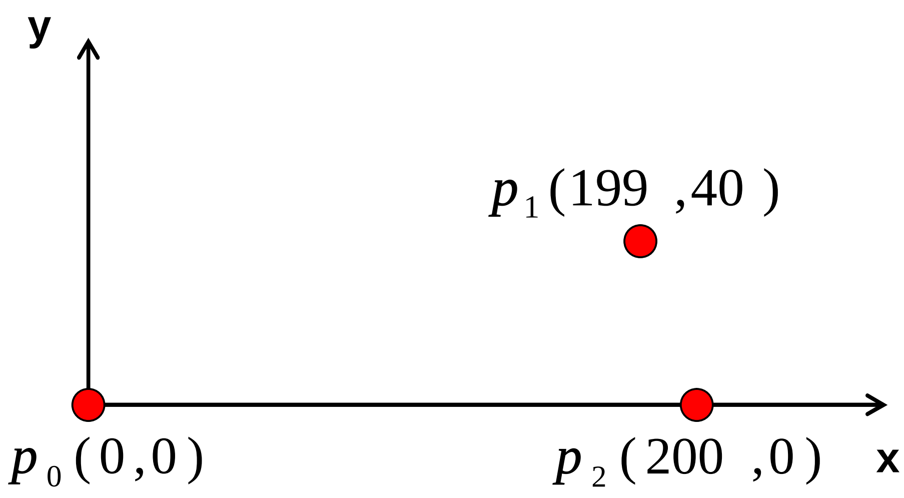
$$0 = y(1) = b_1 + b_2 + b_3$$

6个方程6个未知数，插值问题的本质是方程的个数和未知数的个数是一致的

现在的问题是凭什么取：t=0，t=1/2，t=1?

为什么t不可以取别的值,如t=0，t=1/3，t=1? 哪种取法更科学?

这样一条曲线参数t应该如何取比较好？如果再取成t=0，t=1/2，t=1好不好？



参数化的本质就是找一组恰当的参数t来匹配这一组不同的型值点。给定一组不同的型值点，就要给出不同的参数化即不同的t值，这样才使得这条曲线美观、合理

参数化常用方法:

（1）均匀参数化

节点在参数轴上呈等距分布。如0、1/10、2/10。。。

（2）累加弦长参数化（根据长度的比例关系来确定t）

$$\begin{cases} t_0 = 0 \\ t_i = t_{i-1} + \left|\Delta P_{i-1}\right|, i = 1,2,\cdots,n \end{cases} \qquad \Delta P_i = P_{i+1} - P_i$$

这种参数法如实反映了型值点按弦长的分布情况，能够克服型值点按弦长分布不均匀的情况下采用均匀参数化所出现的问题

3、向心参数化法

$$t_0 = 0$$

$$t_i = t_{i-1} + \left|\Delta P_{i-1}\right|^{1/2}, i = 1,2,\cdots,n$$

向心参数化法假设在一段曲线弧上的向心力与曲线切矢从该弧段始端至末端的转角成正比，加上一些简化假设，得到向心参数化法。此法尤其适用于非均匀型值点分布。

10、参数曲线的代数和几何形式

以三次参数曲线为例，讨论参数曲线的代数和几何形式

(1)代数形式：

$$\begin{cases} x(t) = a_{3x}t^3 + a_{2x}t^2 + a_{1x}t + a_{0x} \\ y(t) = a_{3y}t^3 + a_{2y}t^2 + a_{1y}t + a_{0y} \\ z(t) = a_{3z}t^3 + a_{2z}t^2 + a_{1z}t + a_{0z} \end{cases} \qquad t \in [0,1]$$

上述代数式写成矢量式是：

$$P(t)=a_3t^3+a_2t^2+a_1t+a_0 \qquad t\in[0,1]$$

注意：a_3，a_2，a_1，a_0是参数曲线的系数，但记住不是常数而是向量。a_3对应刚才的a_{3x}、a_{3y}、a_{3z}。改变系数曲线如何变化是不清楚的，这是代数形式的缺点

几何形式:

几何形式是利用一条曲线端点的几何性质来刻画一条曲线。所谓端点的几何性质，就是指曲线的端点位置、切向量、各阶导数等端点的信息。

对三次参数曲线，若用其端点位矢P(0)、P(1)和切矢P’(0)、P’(1)描述。需要这四个量来刻画三次参数曲线

将P(0)、P(1)、P’(0)和 P’(1) 简记为:

$$P_0 、 P_1 、 p_0' 、 p_1'$$

$$P(t) = a_3 t^3 + a_2 t^2 + a_1 t + a_0 \qquad t \in [0,1]$$

$$P_0 = a_0$$

$$P_1 = a_3 + a_2 + a_1 + a_0$$

$$P_0' = a_1$$

$$P_1' = 3a_3 + 2a_2 + a_1$$

$$\Rightarrow \begin{cases} a_0 = P_0 \\ a_1 = P_0' \\ a_2 = -3P_0 + 3P_1 - 2P_0' - P_1' \\ a_3 = 2P_0 - 2P_1 + P_0' + P_1' \end{cases}$$

$$\begin{cases} a_0 = P_0 \\ a_1 = P_0' \\ a_2 = -3P_0 + 3P_1 - 2P'_0 - P_1' \\ a_3 = 2P_0 - 2P_1 + P_0' + P_1' \end{cases}$$

$$P(t) = a_3 t^3 + a_2 t^2 + a_1 t + a_0$$

$$P(t) = (2\mathrm{t}^3 - 3\mathrm{t}^2 + 1)\mathrm{p}_0 + (-2\mathrm{t}^3 + 3\mathrm{t}^2)\mathrm{p}_1 + (\mathrm{t}^3 - 2\mathrm{t}^2 - \mathrm{t})\mathrm{p}_0' + (\mathrm{t}^3 - \mathrm{t}^2)\mathrm{p}_1'$$

得到：

$$P(t) = (2t^3 - 3t^2 + 1)p_0 + (-2t^3 + 3t^2)p_1 + (t^3 - 2t^2 - t)p_0' + (t^3 - t^2)p_1'$$

令：

$$F_0(t) = 2t^3 - 3t^2 + 1 \quad F_1(t) = -2t^3 + 3t^2$$

$$G_0(t) = t^3 - 2t^2 + t \quad G_1(t) = t^3 - t^2$$

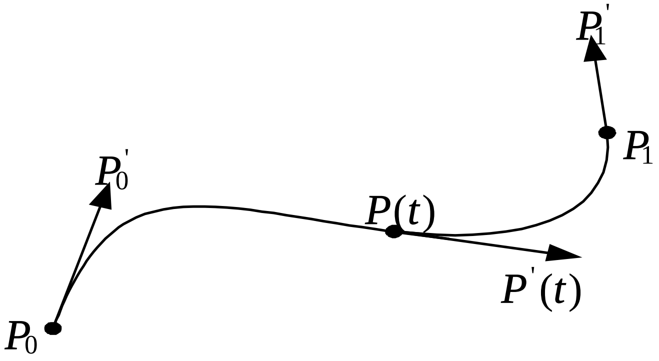
$$P(t) = F_0 P_0 + F_1 P_1 + G_0 P_0^{'} + G_1 P_1^{'} \qquad t \in [0,1]$$

$$P(t) = F_0 P_0 + F_1 P_1 + G_0 P_0' + G_1 P_1' \qquad t \in [0,1]$$

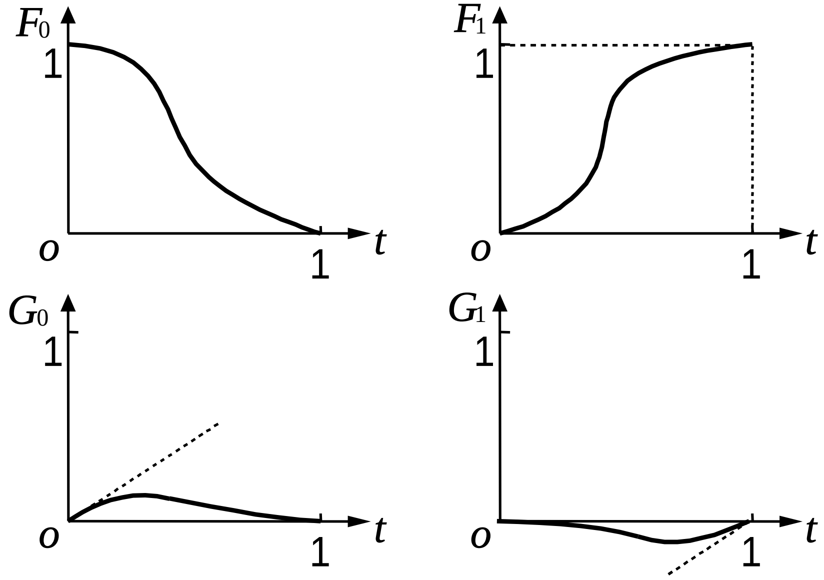
上式是三次Hermite曲线（三次哈密特曲线）的几何形式，几何系数是：

$$P_0、P_1、p_0'、p_1'$$

F0、F1、G0、G1称为调和函数（或混合函数）



曲线端点位矢和切矢



三次调和函数

Bezier曲线与曲面

一、Bezier曲线的背景和定义

1、Bezier曲线的背景

给定n+1个数据点，$p_0(x_0,y_0),\cdots p_n(x_n,y_n)$，生成一条曲线，使得该曲线与这些点所描述的形状相符

如果要求曲线通过所有的数据点，则属于插值问题；如果只要求曲线逼近这些数据点，则属于逼近问题

逼近在计算机图形学中主要用来设计美观的或符合某些美学标准的曲线。为了解决这个问题，有必要找到一种用小的部分即曲线段构建曲线的方法，来满足设计标准

当用曲线段拟合曲线f(x)时，可以把曲线表示为许多小线段$\phi_i(x)$之和，其中$\phi_i(x)$称为基（混合）函数。

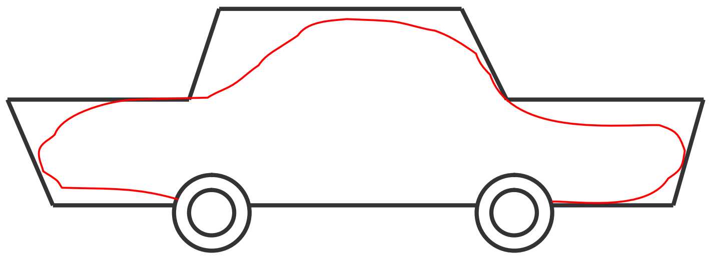
$$f(x) = \sum_{i=0}^{n} a_i f_i(x)$$

这些基（混合）函数是要用于计算和显示的。因此，经常选择多项式作为基（混合）函数

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$$

1962年，贝塞尔（P.E.Bezier）构造了一种以逼近为基础的参数曲线和曲面的设计方法，并用这种方法完成了一种称为UNISURF 的曲线和曲面设计系统。

想法基点是在进行汽车外形设计时，先用折线段勾画出汽车的外形大致轮廓，然后用光滑的参数曲线去逼近这个折线多边形



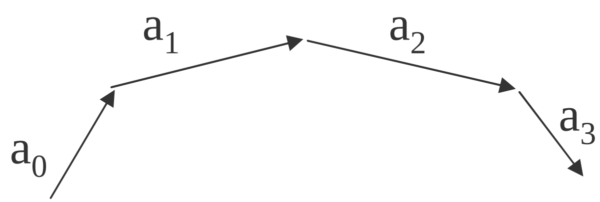
这个折线多边形被称为特征多边形。逼近该特征多边形的曲线被称为Bezier曲线

Bezier方法将函数逼近同几何表示结合起来，使得设计师在计算机上就象使用作图工具一样得心应手

贝塞尔曲线广泛地应用于很多图形图像软件中，例如Flash、Illstrator、CoralDRAW 和 Photoshop 等等

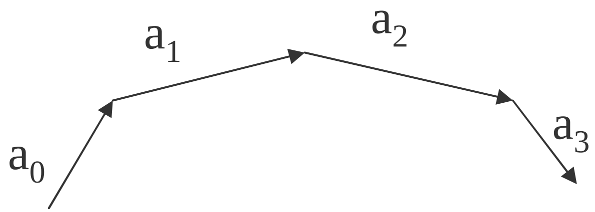
贝塞尔把参数n次曲线表示为：

$$p(t)=\sum_{i=0}^{n} a_i f_{i,n}(t) \qquad 0 \le t \le 1$$



其中系数矢量a_i(i=0,1,…,n)顺序首尾相接

从a_0的末端到a_n的末端所形成的折线称为控制多边形或贝塞尔多边形



$$p(t) = \sum_{i=0}^{n} a_i f_{i,n}(t) \qquad 0 \le t \le 1$$

$$f_{i,n}(t) = \begin{cases} 1, & i = 0, \\ \dfrac{(-t)^i}{(i-1)!} \dfrac{d^{i-1}}{dt^{i-1}} \left(\dfrac{(1-t)^{n-1} - 1}{t} \right) \end{cases}$$ 称为贝塞尔基函数

航空学报
1980年第一期

Bézier 基函数的导出

北京航空学院　施法中　韩道康

摘　要

Bézier 教授在《数值控制》一书中，对 Bézier 曲线曲面作了优美而详尽的说明。Bézier 基函数的表达式表于本文（2）和（3）。

Bézier 曲线有着很多重要的和有趣的几何性质。这些性质是 Bézier 基函数性质的直接推论。到目前为止还没有见到关于导出这些基函数的文献。

本文证明了从 Bézier 曲线的三条简单的、合理的几何要求出发，用两种不同的计算方法可以完全确定并导出这些基函数。

1972年，剑桥大学的博士生Forrest 在《Computer Aided Design》发表了他一生中最著名的论文。Forest证明了Bezier曲线的基函数可以简化成伯恩斯坦基函数！

一个连续函数 y=f（x），任给一个ξ＞0，总能找到一个多项式和这个函数足够逼近。伯恩斯坦有一套逼近的理论，逼近的形式是：

$$B_{i,n}(t) = C_n^i t^i (1-t)^{n-i} = \frac{n!}{i!(n-i)!} t^i (1-t)^{n-i} \qquad (i = 0,1,\ldots.\ n)$$

C_n^i 从n个不同元素中，任取i(i≤n)个元素并成一组，叫做从n个不同元素中取出i个元素的一个组合

Forest证明了Bezier曲线的基函数可以简化成伯恩斯坦基函数

2、Bezier曲线的定义

针对Bezier曲线，给定空间n+1个点的位置矢量P_i（i=0，1，2，…，n），则Bezier曲线段的参数方程表示如下：

$$p(t)=\sum_{i=0}^{n} P_i B_{i,n}(t) \qquad t\in[0,1]$$

$$p(t)=\sum_{i=0}^{n} P_i B_{i,n}(t) \qquad t\in[0,1]$$

其中p_i（x_i,y_i,z_i），i=0,1,2⋯n 是控制多边形的n+1个顶点，即构成该曲线的特征多边形；$B_{i,n}$（t）是Bernstein基函数，有如下形式：

$$B_{i,n}(t)=\frac{n!}{i!(n-i)!}t^i(1-t)^{n-i}=C_n^i t^i(1-t)^{n-i} \qquad (i=0,1,\ldots.\ n)$$

二项式定理，又称牛顿二项式定理。该定理给出两个数之和的整数次幂的恒等式

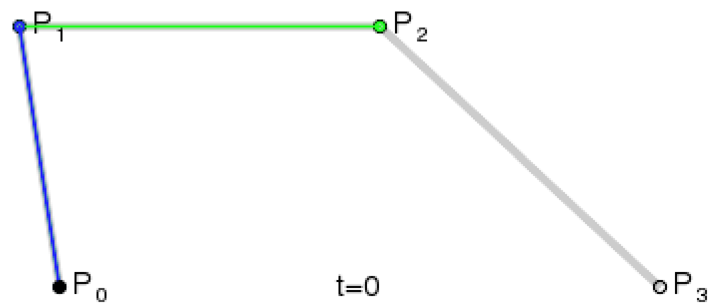
$$B_{i,n}(t) = \frac{n!}{i!(n-i)!} t^i (1-t)^{n-i} = C_n^i t^i (1-t)^{n-i} \qquad (i = 0,1,\ldots n)$$

$\sum_{i=0}^{n} B_{i,n}(t)$ 恰好是二项式 $[t+(1-t)]^n$ 的展开式！

注意：当i=0，t=0时，t^i=1，i!=1。即： 0^0=1，0!=1

P_i代表空间的很多点，t在0到1之间，把t代进去可以算出一个数--即平面或空间一个点

随着t值的变化，点也在变化。当t从0变到1时，就得到空间的一个图形，这个图形就是bezier曲线



（1）一次Bezier曲线

$$p(t) = \sum_{i=0}^{n} P_i B_{i,n}(t) \qquad t \in [0,1]$$

当n=1时，有两个控制点p_0和p_1，Bezier多项式是一次多项式：

$$p(t) = \sum_{i=0}^{1} P_i B_{i,1}(t) = P_0 B_{0,1}(t) + P_1 B_{1,1}(t)$$

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

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

$$B_{0,1}(t) = \frac{n!}{i!(n-i)!} t^{i} (1-t)^{n-i} = \frac{1!}{0!(1-0)!} t^{0} (1-t)^{1-0}$$

$$= (1-t)$$

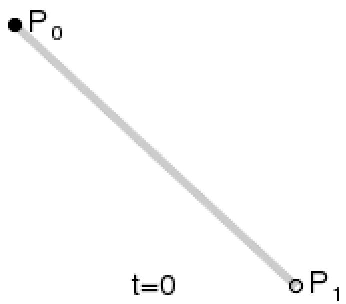
$$B_{1,1}(t) = \frac{n!}{i!(n-i)!} t^{i} (1-t)^{n-i} = \frac{1!}{1!(1-1)!} t^{1} (1-1)^{1-1}$$

$$= t$$

$$p(t) = \sum_{i=0}^{1} P_i B_{i,1}(t) = P_0 B_{0,1}(t) + P_1 B_{1,1}(t)$$

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

这恰好是连接起点p_0和终点p_1的直线段！



（2）二次Bezier曲线

$$p(t)=\sum_{i=0}^{n} P_i B_{i,n}(t) \qquad t\in[0,1]$$

当n=2时，有3个控制点p_0、p_1和p_2，Bezier多项式是二次多项式：

$$p(t)=\sum_{i=0}^{2} P_i B_{i,2}(t)=P_0B_{0,2}(t)+P_1B_{1,2}(t)+P_2B_{2,2}(t)$$

$$B_{0,2}(t)=\frac{n!}{i!(n-i)!}t^i(1-t)^{n-i}=\frac{2!}{0!(2-0)!}t^0(1-t)^{2-0}$$

$$=(1-t)^2$$

$$B_{0,2}(t) = (1 - t)^2$$

$$B_{1,2}(t) = \frac{n!}{i!(n-i)!} t^i (1-t)^{n-i} = \frac{2!}{1!(2-1)!} t^1 (1-t)^{2-1}$$

$$= 2t(1-t)$$

$$B_{2,2}(t) = \frac{n!}{i!(n-i)!} t^i (1-t)^{n-i} = \frac{2!}{2!(2-2)!} t^2 (1-t)^{2-2}$$

$$= t^2$$

$$p(t) = \sum_{i=0}^{2} P_i B_{i,2}(t) = P_0 B_{0,2}(t) + P_1 B_{1,2}(t) + P_2 B_{2,2}(t)$$

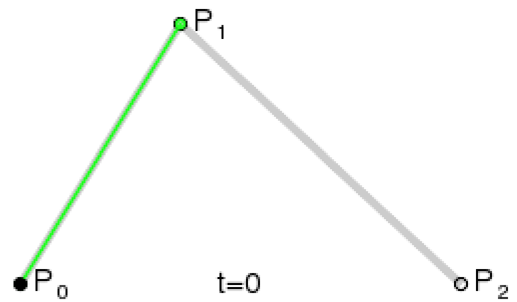
$$= (1-t)^2 P_0 + 2t(1-t) P_1 + t^2 P_2$$

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

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

二次Bezier曲线为抛物线，其矩阵形式为：

$$p(t) = \begin{bmatrix} t^2 & t & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & -2 & 1 \\ -2 & 2 & 0 \\ 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} P_0 \\ P_1 \\ P_2 \end{bmatrix}$$



（3）三次Bezier曲线

三次Bezier曲线由4个控制点生成，这时n=3，有4个控制点p_0、p_1、p_2和p_3，Bezier多项式是三次多项式：

$$p(t)=\sum_{i=0}^{3} P_i B_{i,3}(t)=P_0 B_{0,3}(t)+P_1 B_{1,3}(t)+P_2 B_{2,3}(t)+P_3 B_{3,3}(t)$$

$$B_{0,3}(t)=\frac{n!}{i!(n-i)!}t^i(1-t)^{n-i}=\frac{3!}{0!(3-0)!}t^0(1-t)^{3-0}=(1-t)^3$$

$$B_{1,3}(t)=\frac{n!}{i!(n-i)!}t^i(1-t)^{n-i}=\frac{3!}{1!(3-1)!}t^1(1-t)^{3-1}=3t(1-t)^2$$

$$B_{2,3}(t) = \frac{n!}{i!(n-i)!} t^i (1-t)^{n-i} = \frac{3!}{2!(3-2)!} t^2 (1-t)^{3-2} = 3t^2(1-t)$$

$$B_{3,3}(t) = \frac{n!}{i!(n-i)!} t^i (1-t)^{n-i} = \frac{3!}{3!(3-3)!} t^3 (1-t)^{3-3} = t^3$$

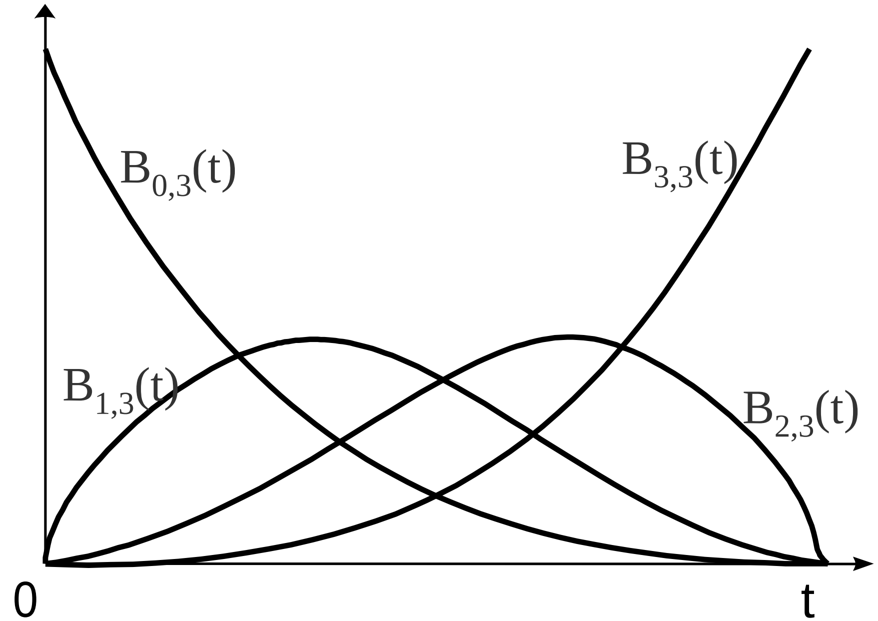
$$p(t) = \sum_{i=0}^{3} P_i B_{i,3}(t) = (1-t^3) P_0 + 3t(1-t)^2 P_1 + 3t^2(1-t) P_2 + t^3 P_3$$

其中

$$B_{0,3}(t) = (1-t)^3$$
$$B_{1,3}(t) = 3t(1-t)^2$$
$$B_{2,3}(t) = 3t^2(1-t)$$
$$B_{3,3}(t) = t^3$$

为三次Bezier曲线的基函数。

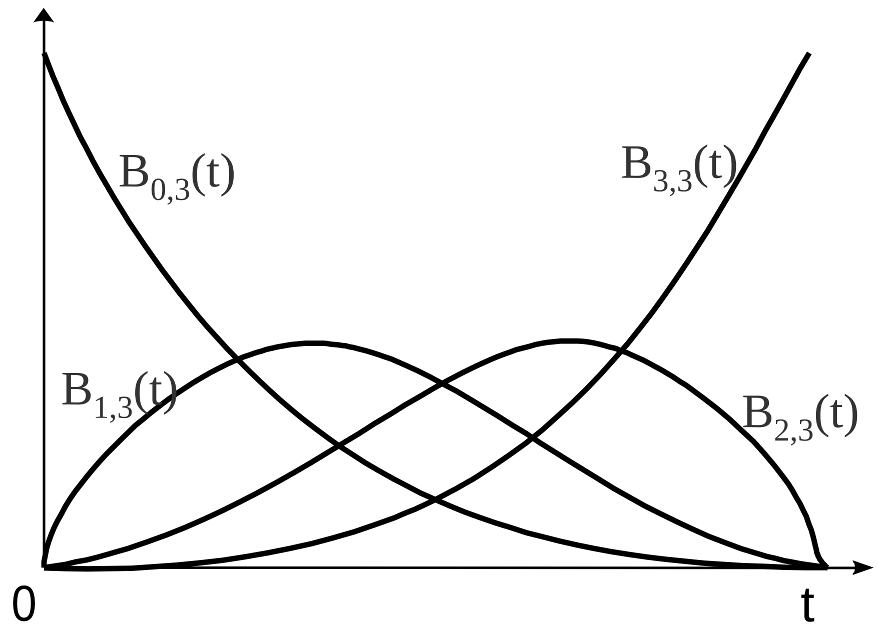
这四条曲线均是三次曲线，任何三次Bezier曲线都是这四条曲线的线形组合



三次Bezier曲线四个Bezier基函数

注意图中每个基函数在参数t的整个（0，1）的开区间范围内不为0

Bezier曲线不可能对曲线形状进行局部控制，如果改变任一控制点位置，整个曲线会受到影响



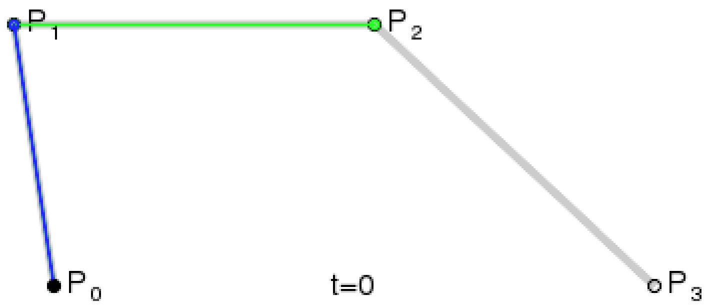
三次Bezier曲线四个Bezier基函数

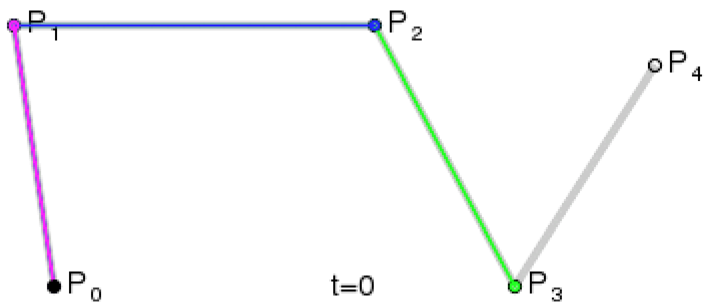
把Bezier三次曲线多项式写成矩阵形式：

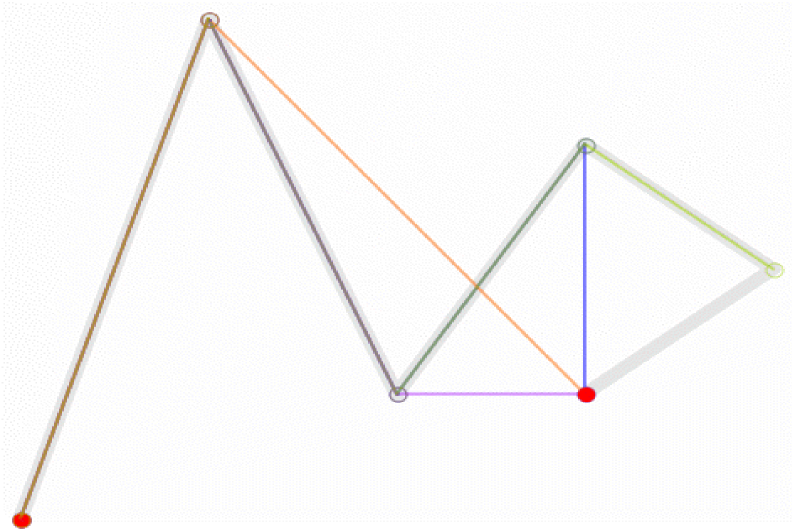
$$p(t)=\begin{bmatrix}t^3 & t^2 & t & 1\end{bmatrix}\begin{bmatrix}-1 & 3 & -3 & 1\\ 3 & -6 & 3 & 0\\ -3 & 3 & 0 & 0\\ 1 & 0 & 0 & 0\end{bmatrix}\begin{bmatrix}P_0\\ P_1\\ P_2\\ P_3\end{bmatrix} \qquad t\in[0,1]$$

$$=T\cdot M_{be}\cdot G_{be}$$

其中，M_{be}是三次Bezier曲线系数矩阵，为常数；G_{be}是4个控制点位置矢量。


P_1
P_2
P_0
t=0
P_3


P_1
P_2
P_4
P_0
t=0
P_3



二、Bernstein基函数的性质

1、正性（非负性）

$$B_{i,n}(t) = \begin{cases} = 0 & t = 0,1 \\ > 0 & t \in (0,1), i = 1,2,\cdots,n-1; \end{cases}$$

2、权性

基函数有n+1项，n+1个基函数的和加起来正好等于1

$$\sum_{i=0}^{n} B_{i,n}(t) \equiv 1 \qquad t \in (0,1)$$

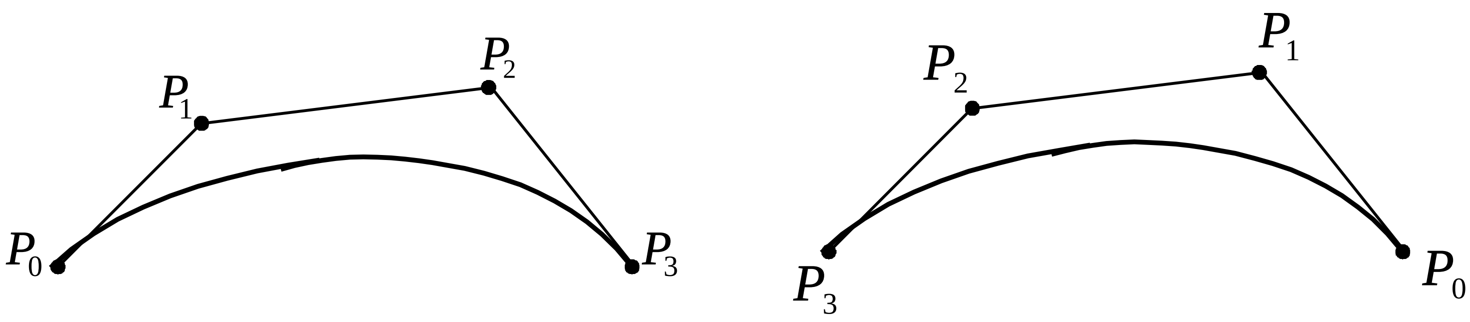
3、端点性质

$$B_{i,n}(0)=\begin{cases}1 & (i=0)\\ 0 & otherwise\end{cases}$$

$$B_{i,n}(1)=\begin{cases}1 & (i=n)\\ 0 & otherwise\end{cases}$$

4、对称性

可以证明，假如保持n次Bezier曲线控制多边形的顶点位置不变，而把次序颠倒过来，则此时曲线仍不变，只不过曲线的走向相反而已



$$B_{i,n}(t) = B_{n-i,n}(1-t)$$

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

$$= C_n^i t^i (1-t)^{n-i} = B_{i,n}(t)$$

5、递推性

$$B_{i,n}(t)=(1-t)B_{i,n-1}(t)+tB_{i-1,n-1}(t) \qquad (i=0,1,\mathbf{L}\ n)$$

即n次的Bernstein基函数可由两个n-1次的Bernstein基函数线性组合而成。因为:

$$C_n^i=(C_{n-1}^i+C_{n-1}^{i-1})$$

$$\begin{aligned}
B_{i,n}(t) &= C_n^i t^i (1-t)^{n-i} = (C_{n-1}^i + C_{n-1}^{i-1}) t^i (1-t)^{n-i} \\
&= (1-t) C_{n-1}^i t^i (1-t)^{(n-1)-i} + t C_{n-1}^{i-1} t^{i-1} (1-t)^{(n-1)-(i-1)} \\
&= (1-t) B_{i,n-1}(t) + t B_{i-1,n-1}(t)
\end{aligned}$$

6、导函数

$$B'_{i,n}(t) = n[B_{i-1,n-1}(t) - B_{i,n-1}(t)] \qquad i = 0,1,\cdots,n;$$

7、最大值

$B_{i,n}(t)$在 $t = \dfrac{i}{n}$ 处达到最大值

8、积分

$$\int_0^1 B_{i,n}(t)\,dt = \frac{1}{n+1}$$

9、降阶公式

$$B_{i,n}(u) = (1-u)B_{i,n-l}(u) + uB_{i-l,n-l}(u)$$

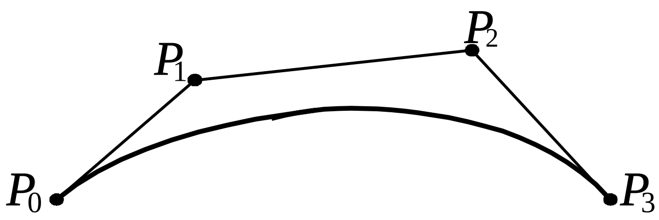
一个n次Bernstein基函数能表示成两个n-1次基函数的线性和

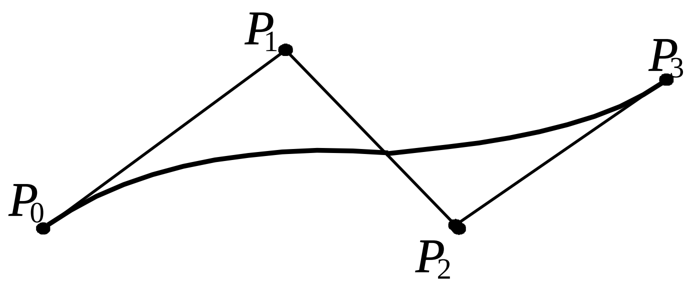
10、升阶公式

$$B_{i,n}(n) = \frac{i+1}{n+1}B_{i+1,n+1}(t) + \frac{n+1-i}{n+1}B_{i-1,n-1}(t)$$

三、Bezier曲线的性质

1、端点性质





顶点p_0和p_n分别位于实际曲线段的起点和终点上

Bezier曲线段的参数方程表示如下:

$$p(t)=\sum_{i=0}^{n} P_i B_{i,n}(t) = P_0 B_{0,n}(t) + P_1 B_{1,n}(t) + \mathbf{K} + P_n B_{n,n}(t)$$

$$p(0)=\sum_{i=0}^{n} P_i B_{i,n}(0) = P_0 B_{0,n}(0) + P_1 B_{1,n}(0) + \mathbf{K} + P_n B_{n,n}(0) = P_0$$

$$p(1)=\sum_{i=0}^{n} P_i B_{i,n}(1) = P_0 B_{0,n}(1) + P_1 B_{1,n}(1) + \mathbf{K} + P_n B_{n,n}(1) = P_n$$

2、一阶导数

Bernstein基函数的一阶导数为：

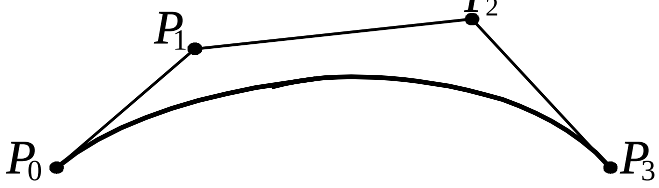
$$B'_{i,n}(t) = n[B_{i-1,n-1}(t) - B_{i,n-1}(t)] \qquad i = 0,1,\cdots,n;$$

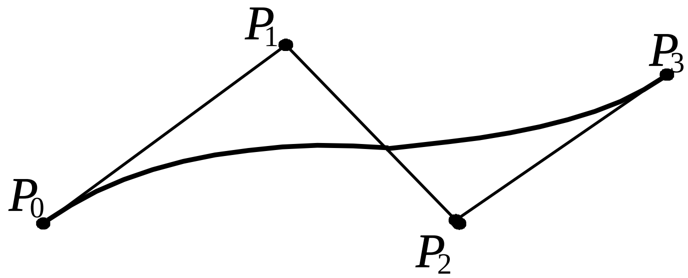
$$p'(t) = n\sum_{i=1}^{n}(p_i - p_{i-1})B_{i-1,n-1}(t)$$

当 t=0：　　　　当 t=1：

$$p'(0) = n(p_1 - p_0) \qquad p'(1) = n(p_n - p_{n-1})$$

这说明Bezier曲线的起点和终点处的切线方向和特征多边形的第一条边及最后一条边的走向一致





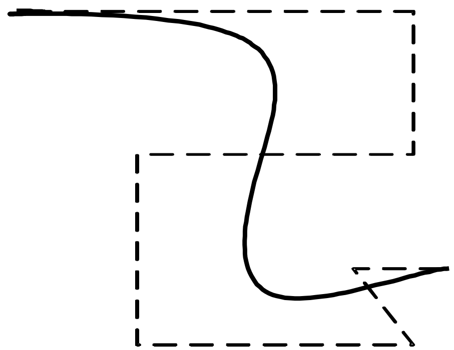
3、几何不变性

指某些几何特性不随坐标变换而变化的特性。Bezier曲线的形状仅与控制多边形各顶点的相对位置有关，而与坐标系的的选择无关

4、变差缩减性

若Bezier曲线的特征多边形是一个平面图形，则平面内任意直线与p(t)的交点个数不多于该直线与其特征多边形的交点个数，这一性质叫变差缩减性质

此性质反映了Bezier曲线比其特征多边形的波动还小，也就是说Bezier曲线比特征多边形的折线更光顺



三、Bezier曲线的生成

生成一条Bezier曲线实际上就是要求出曲线上的点。下面介绍两种曲线生成的方法：

1、根据定义直接生成Bezier曲线

绘制Bezier曲线主要有以下步骤：

$$p(t)=\sum_{i=0}^{n} P_i B_{i,n}(t) \qquad t\in[0,1]$$

$$B_{i,n}(t) = \frac{n!}{i!(n-i)!}t^i(1-t)^{n-i} = C_n^i t^i (1-t)^{n-i} \qquad (i = 0, 1, \ldots, n)$$

① 首先给出 C_n^i 的递归计算式：

$$C_n^i = \frac{n!}{i!\left(n-i\right)!} = \frac{n-i+1}{i} C_n^{i-1} \qquad \mathrm{n} \geq \mathrm{i}$$

② 将 $p(t) = \sum_{i=0}^{n} P_i B_{i,n}(t)$ 表示成分量坐标形式：

$$x(t) = \sum_{i=0}^{n} x_i B_{i,n}(t)$$

$$y(t) = \sum_{i=0}^{n} y_i B_{i,n}(t) \qquad t \in [0,1]$$

$$z(t) = \sum_{i=0}^{n} z_i B_{i,n}(t)$$

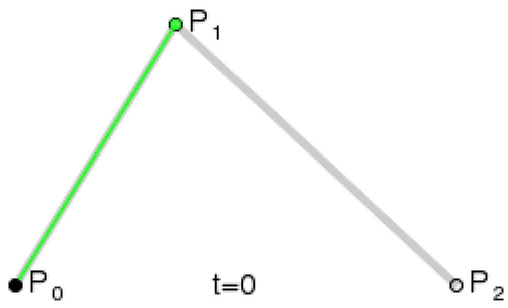
根据以上的公式可以直接写出绘制Bezier曲线的程序

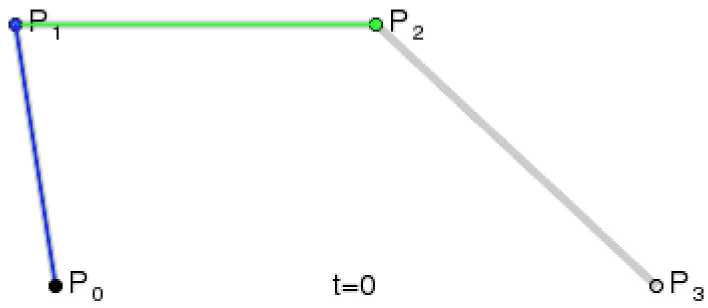
$$C_n^i = \frac{n!}{i!(n-i)!} = \frac{n-i+1}{i} C_n^{i-1} \qquad \mathrm{n} \ge \mathrm{i}$$

2、Bezier曲线的递推(de Casteljau)算法

根据Bezier曲线的定义确定的参数方程绘制Bezier曲线,因其计算量过大,不太适合在工程上使用

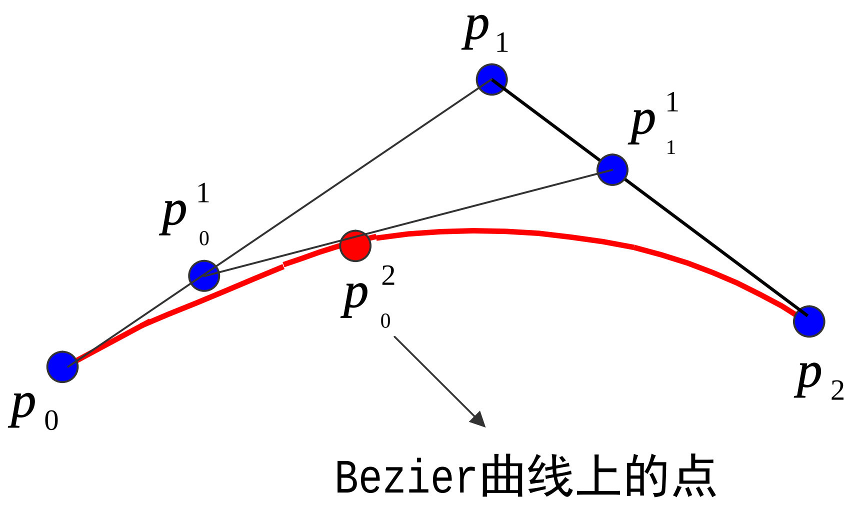
de Casteljau提出的递推算法则要简单得多



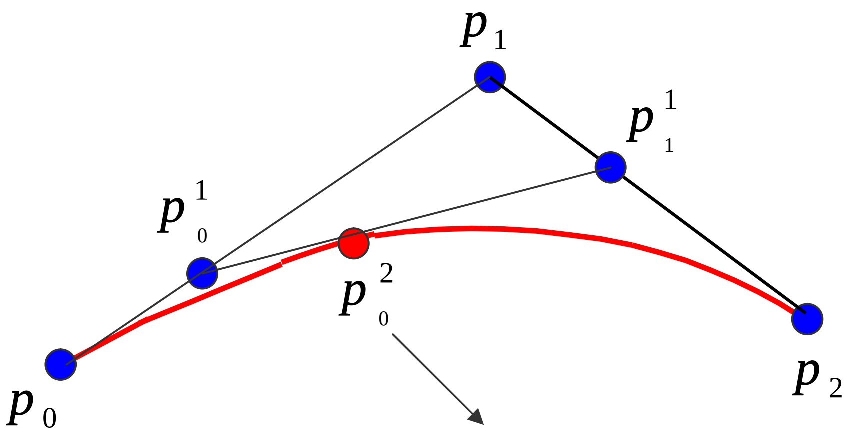


Bezier曲线上的任一个点(t)，都是其它相邻线段的同等比例(t)点处的连线，再取同等比例(t)的点再连线，一直取到最后那条线段的同等比例(t)处，该点就是Beizer曲线上的点(t)

以二次Bezier曲线为例，求曲线上t=1/3的点：



$$P_0^1 = (1-t)P_0 + tP_1$$
$$P_1^1 = (1-t)P_1 + tP_2$$
$$P_0^2 = (1-t)P_0^1 + tP_1^1$$



t从0变到1，第一、二式就分别表示控制二边形的第一、二条边，它们是两条一次Bezier曲线。将一、二式代入第三式得：

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

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

当t从0变到1时，它表示了由三顶点P_0、P_1、P_2三点定义的一条二次Bezier曲线

二次Bezier曲线P_0^2可以定义为分别由前两个顶点(P_0,P_1)和后两个顶点(P_1,P_2)决定的一次Bezier曲线的线性组合

由(n+1)个控制点P_i(i=0,1,...,n)定义的n次Bezier曲线P_0^n可被定义为分别由前、后n个控制点定义的两条(n-1)次Bezier曲线P_0^{n-1}与P_1^{n-1}的线性组合:

$$P_0^n = (1-t)P_0^{n-1} + tP_1^{n-1} \qquad t \in [0,1]$$

由此得到Bezier曲线的递推计算公式：

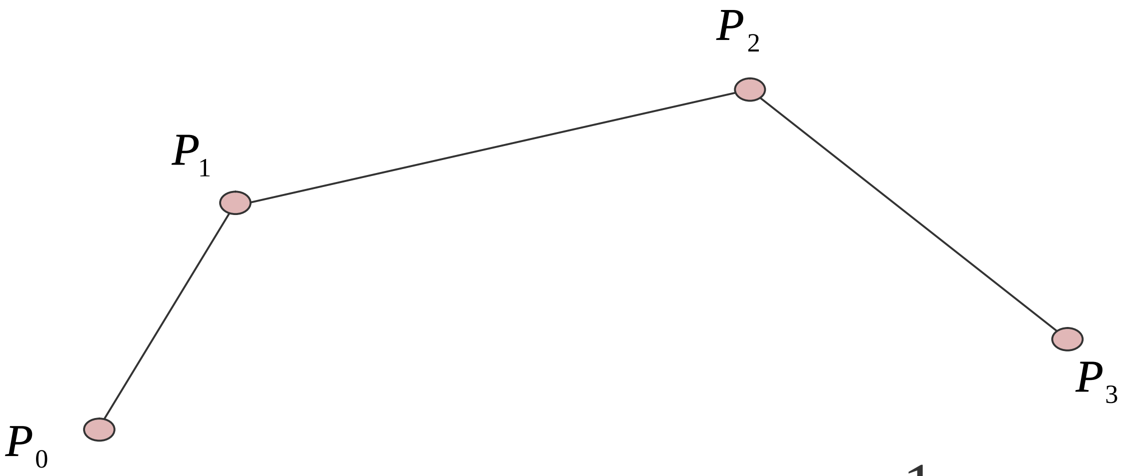
$$P_i^k = \begin{cases} P_i & k = 0 \\ (1-t)P_i^{k-1} + tP_{i+1}^{k-1} & k = 1,2,\ldots,n, i = 0,1,\ldots,n-k \end{cases}$$

这便是著名的de Casteljau算法。用这一递推公式，在给定参数下，求Bezier曲线上一点P(t)非常有效

de Casteljau算法稳定可靠，直观简便，可以编出十分简捷的程序，是计算Bezier曲线的基本算法和标准算法。

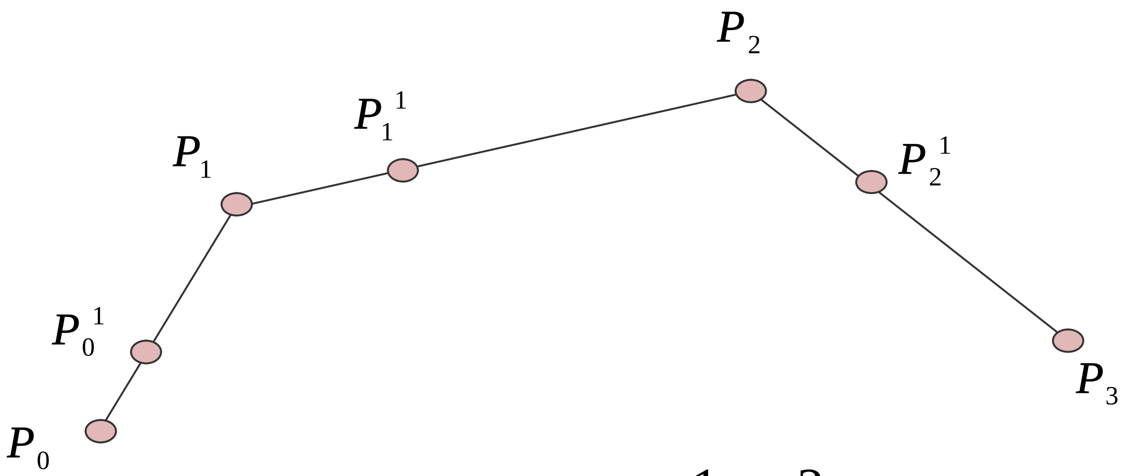
3、de Casteljau算法几何作图

这一算法可用简单的几何作图来实现



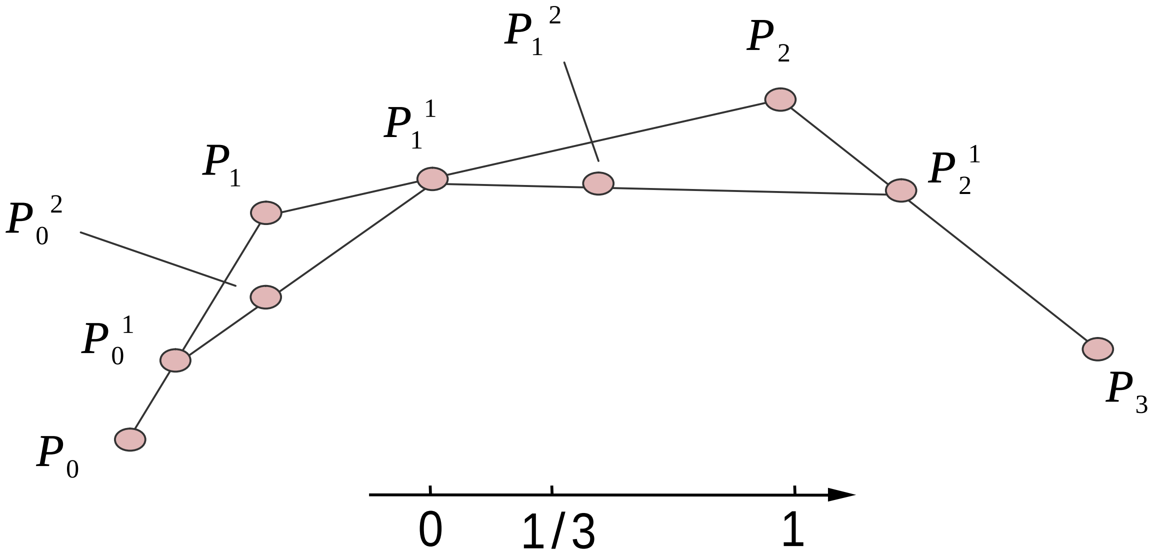
$t \in [0,1]$ 假设 $t = \dfrac{1}{3}$

（1）依次对原始控制多边形每一边执行相同的定比分割，所得分点就是由第一级递推生成的中间顶点 $P_i^1(i=0,1,\cdots,n-1)$

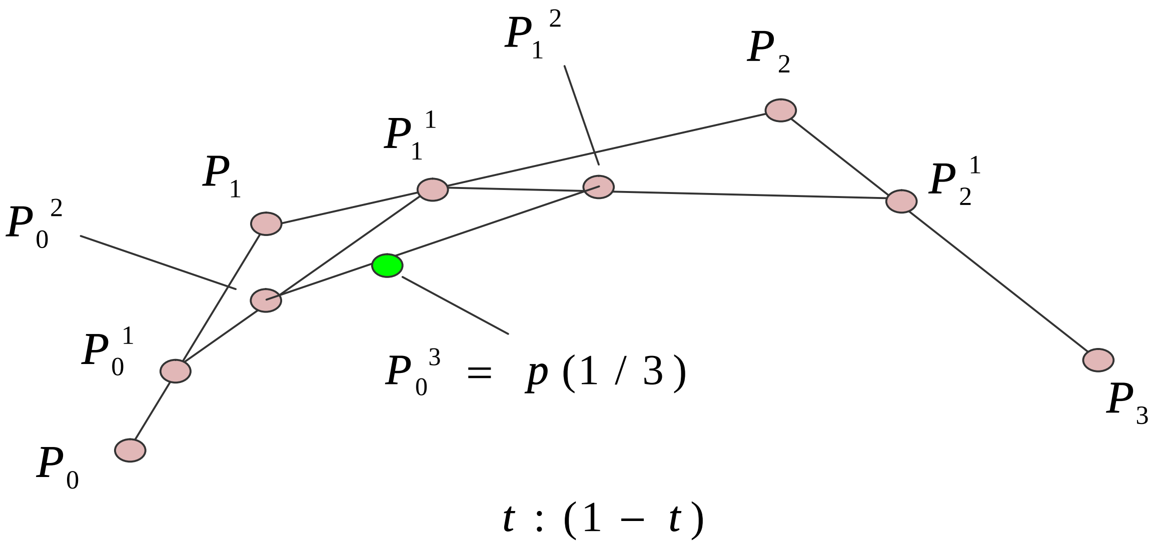


$$t:(1-t) \qquad \frac{1}{3}:(\frac{2}{3})$$

（2）对这些中间顶点构成的控制多边形再执行同样的定比分割，得第二级中间顶点：$P_i^2(i=0,1,\mathbf{L},n-2)$

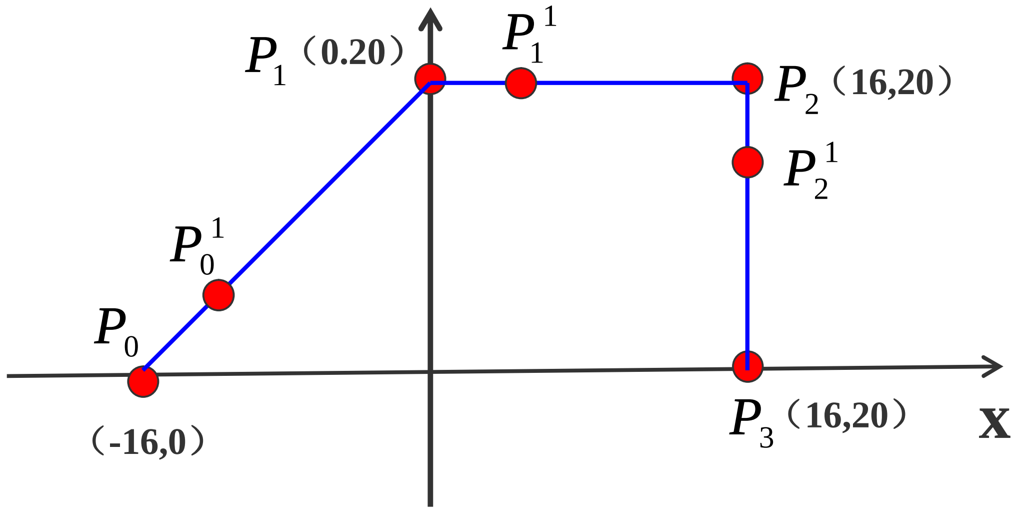


（3）重复进行下去，直到n级递推得到一个中间顶点 $P(t)$ ，即为所求曲线上的点



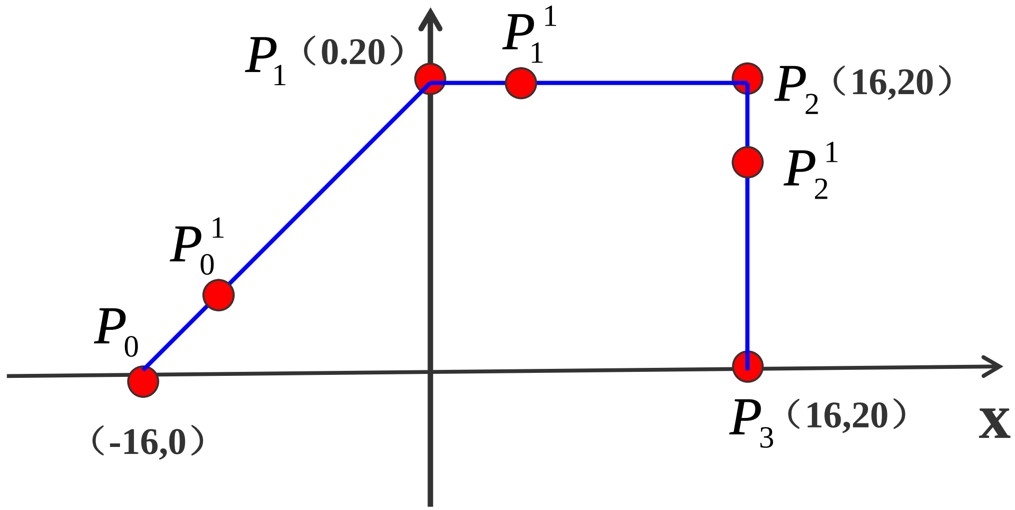
Bezier曲线计算举例

计算参数为 t=1/4 的P值



$$P_i^k = \begin{cases} P_i & k = 0 \\ (1-t)P_i^{k-1} + tP_{i+1}^{k-1} & k = 1,2,\dots, n, i = 0,1,\dots, n-k \end{cases}$$

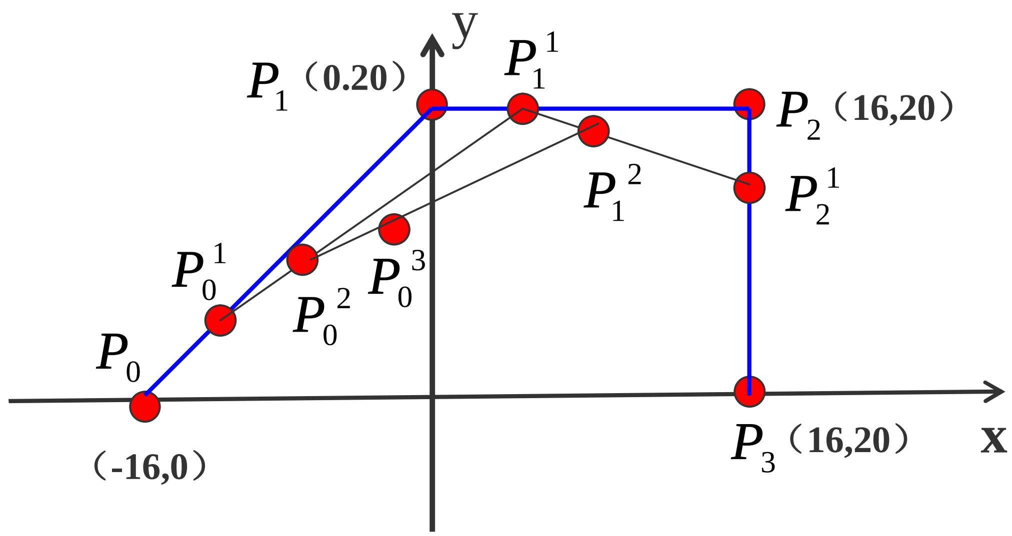
$$P_0^1 = (1-t)P_0 + tP_1 = (1-\frac{1}{4})p_0 + \frac{1}{4}p_1 = \frac{3}{4}[-16,0] + \frac{1}{4}[0,20] = [-12,5]$$



$$P_0^1 = (1-t)P_0 + tP_1 = (1-\frac{1}{4})p_0 + \frac{1}{4}p_1 = \frac{3}{4}[-16,0] + \frac{1}{4}[0,20] = [-12,5]$$

$$P_1^1 = (1-t)P_1 + tP_2 = (1-\frac{1}{4})p_1 + \frac{1}{4}p_2 = \frac{3}{4}[0,20] + \frac{1}{4}[16,20] = [4,20]$$

$$P_2^1 = (1-t)P_2 + tP_3 = (1-\frac{1}{4})p_2 + \frac{1}{4}p_3 = \frac{3}{4}[16,20] + \frac{1}{4}[16,0] = [16,15]$$



$$P_0^2 = (1-t)P_0^1 + tP_1^1 = (1-\frac{1}{4})P_0^1 + \frac{1}{4}P_1^1 = \frac{3}{4}[-12,5] + \frac{1}{4}[4,20] = [-8,8.75]$$

$$P_1^2 = (1-t)P_1^1 + tP_2^1 = (1-\frac{1}{4})P_0^1 + \frac{1}{4}P_1^1 = \frac{3}{4}[-12,5] + \frac{1}{4}[4,20] = [-8,8.75]$$

$$P_0^3 = (1-t)P_0^2 + tP_1^2 = (1-\frac{1}{4})P_0^2 + \frac{1}{4}P_1^2 = \frac{3}{4}[-8,8.75] + \frac{1}{4}[7,18.75]$$

$$P_0^1 = (1-t)P_0 + tP_1 = (1-\frac{1}{4})p_0 + \frac{1}{4}p_1 = \frac{3}{4}[-16,0] + \frac{1}{4}[0,20] = [-12,5]$$

$$P_1^1 = (1-t)P_1 + tP_2 = (1-\frac{1}{4})p_1 + \frac{1}{4}p_2 = \frac{3}{4}[0,20] + \frac{1}{4}[16,20] = [4,20]$$

$$P_2^1 = (1-t)P_2 + tP_3 = (1-\frac{1}{4})p_2 + \frac{1}{4}p_3 = \frac{3}{4}[16,20] + \frac{1}{4}[16,0] = [16,15]$$

$$P_0^2 = (1-t)P_0^1 + tP_1^1 = (1-\frac{1}{4})P_0^1 + \frac{1}{4}P_1^1 = \frac{3}{4}[-12,5] + \frac{1}{4}[4,20] = [-8,8.75]$$

$$P_0^3 = (1-t)P_0^2 + tP_1^2 = (1-\frac{1}{4})P_0^2 + \frac{1}{4}P_1^2 = \frac{3}{4}[-8,8.75] + \frac{1}{4}[7,18.75]$$

$$= [-4.25, 11.25] = P(1/4)$$

四、Bezier曲线的拼接

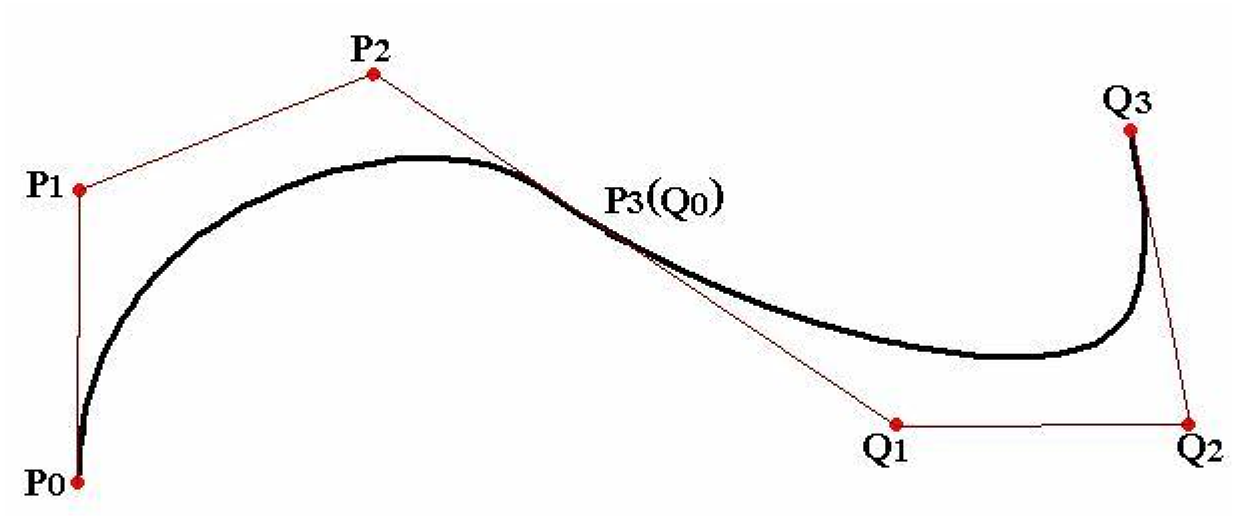
几何设计中，一条Bezier曲线往往难以描述复杂的曲线形状。这是由于增加特征多边形的顶点数，会引起Bezier曲线次数的提高，而高次多项式又会带来计算上的困难

$$p(t) = \sum_{i=0}^{n} P_i B_{i,n}(t) \qquad t \in [0,1]$$

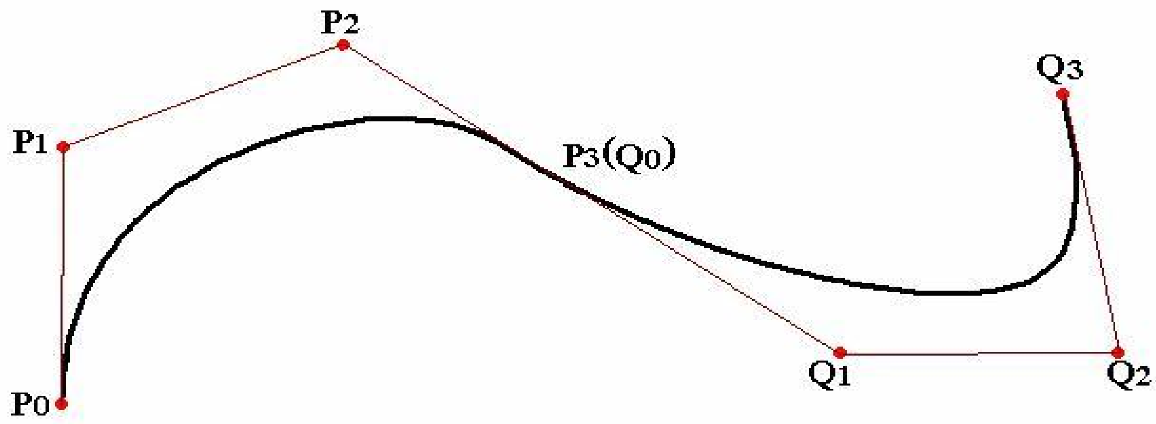
$$B_{i,n}(t) = C_n^i t^i (1-t)^{n-i} \qquad (i = 0,1,\ldots, n)$$

采用分段设计，然后将各段曲线相互连接起来，
并在接合处保持一定的连续条件

给定两条Bezier曲线 P(t)和 Q(t)，相应控制点为 $P_i(i=0,1,...,n)$和 $Q_i(i=0,1,...,m)$



（1）要使它们达到 G^0连续，则：$P_n = Q_0$



（2）要使它们达到G^1连续，只要保证P_{n-1}，$P_n=Q$，Q_1三点共线就行了

Bezier曲线的起点和终点处的切线方向和特征多边形的第一条边及最后一条边的走向一致

五、Bezier曲线的升阶与降阶

假设有一个二次多项式：

$$a_0 + a_1 t + a_2 t = 0$$

能否找一个三次多项式近似逼近这个二次多项式，或精确地转成一个三次多项式？

$$a_0 + a_1 t + a_2 t^2 = 0$$

$$a_0 + a_1 t + a_2 t^2 + 0 * t^3 = 0$$

所谓升阶就是保证曲线的形状和定向保持不变，但是要增加顶点个数

伯恩斯坦基函数不是简单的t^2、t^3，2次Bezier基函数是$B_{i,2}$，3次是$B_{i,3}$，如何从$B_{i,2}$转化成$B_{i,3}$的形式？

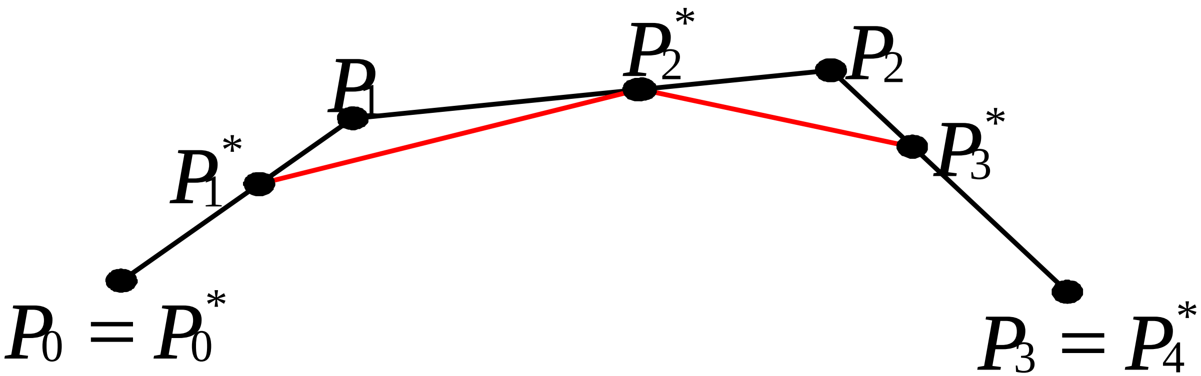
1、Bezier曲线的升阶

所谓升阶是指保持Bezier曲线的形状与方向不变，增加定义它的控制顶点数，即提高该Bezier曲线的次数

设给定原始控制顶点$p_0,p_1..p_n$，定义一条n次Bezier曲线

$$P(t)=\sum_{i=0}^{n} P_i B_{i,n}(t) \qquad t\in[0,1]$$

增加一个顶点后，仍定义同一条曲线的新控制顶点为 $P_0^*, P_1^*, ..., P_{n+1}^*$，则有：

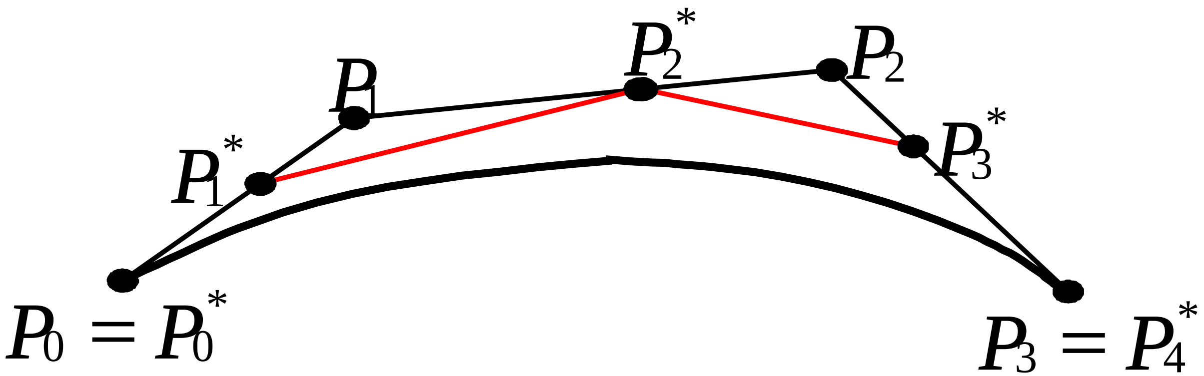


$$\sum_{i=0}^{n} C_n^i P_i t^i (1-t)^{n-i} = \sum_{i=0}^{n+1} C_{n+1}^i P_i^* t^i (1-t)^{n+1-i}$$

$$P_i^* C_{n+1}^i = P_i C_n^i + P_{i-1} C_n^{i-1}$$

化简即得：$P_i^* = \dfrac{i}{n+1} P_{i-1} + \left(1 - \dfrac{i}{n+1}\right) P_i \qquad (i = 0, 1, \cdots, n+1)$

三次Bezier曲线的升阶实例如下图所示：



新的多边形更加靠近曲线。在80年代初，有人证明如果一直升阶升下去的话，控制多边形收敛于这条曲线

2、Bezier曲线的降阶

降阶是升阶的逆过程

$$a_0 + a_1 t + a_2 t^2 + a_3 t^3 = b_0 + b_1 t + b_2 t^2$$

如果a_3不等于0，想找一个二次多项式精确等于它是不可能的。如果一定要降下来，那只能近似逼近

如果把a_3t^3扔掉，也不是不可以，但误差太大。必须想办法找到一个二次多项式，尽量逼近这个三次多项式

假定P_i是由P_i^*升阶得到，则由升阶公式有：

$$P_i = \frac{n-i}{n} P_i^* + \frac{i}{n} P_{i-1}^*$$

从这个方程可以导出两个递推公式：

$$P_i^* = \frac{nP_i - iP_{i-1}^*}{n-i} \qquad i = 0,1,\ldots,n-1$$

$$P_{i-1}^* = \frac{nP_i - (n-i)P_i^*}{i} \qquad i = n, n-1, \ldots, 1$$

可以综合利用上面两个式子进行降解，但仍然不精确

Bezier曲线曲面升降阶的重要性

第一 它是CAD系统之间数据传递与交换的需要

第二 它是系统中分段(片)线性逼近的需要. 通过逐次降阶，把曲面化为直线平面，便于求交和曲面绘制

第三 它是外形信息压缩的需要。降阶处理以后可以减少存储的信息量

六、Bezier曲面

基于Bezier曲线的讨论，可以给出Bezier曲面的定义和性质，Bezier曲线的一些算法也可以很容易扩展到Bezier曲面的情况

1、Bezier曲面的定义

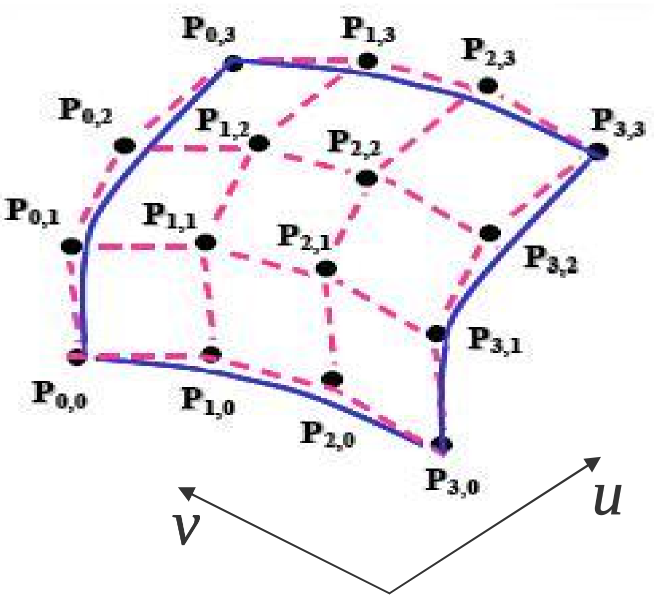
设 $p_{i,j}(0,1,\ldots,n;\ j=0,1,\ldots,m)$ 为 $(n+1)\times(m+1)$ 个空间点，则 $m\times n$ Bezier曲面定义为：

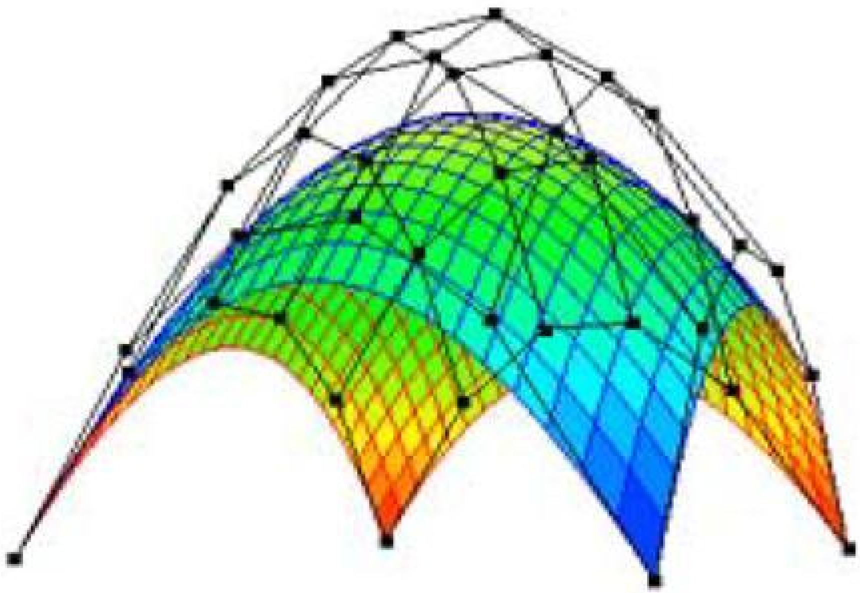
$$P(u,v)=\sum_{i=0}^{m}\sum_{j=0}^{n}P_{ij}B_{i,m}(u)B_{j,n}(v) \qquad u,v\in[0,1]$$

$$B_{i,m}(u)=C_m^i u^i(1-u)^{m-i}$$

$$B_{i,n}(v)=C_n^j v^j(1-v)^{n-j}$$

依次用线段连接点列中相邻两点所形成的空间网格，称之为特征网格

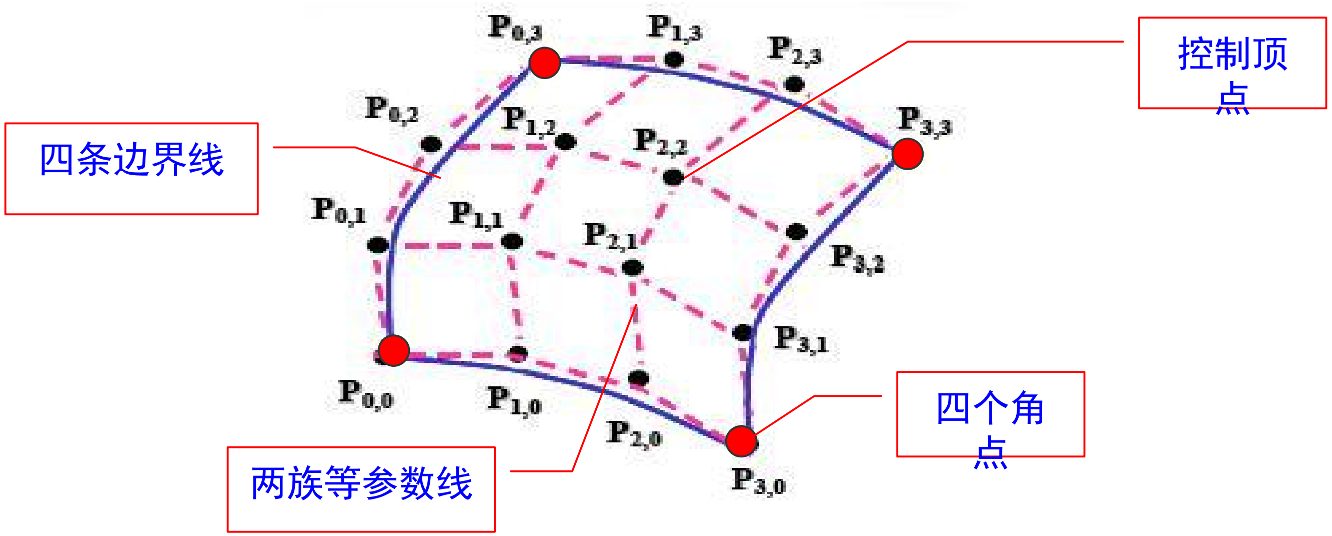




Bezier曲面的矩阵表示式是:

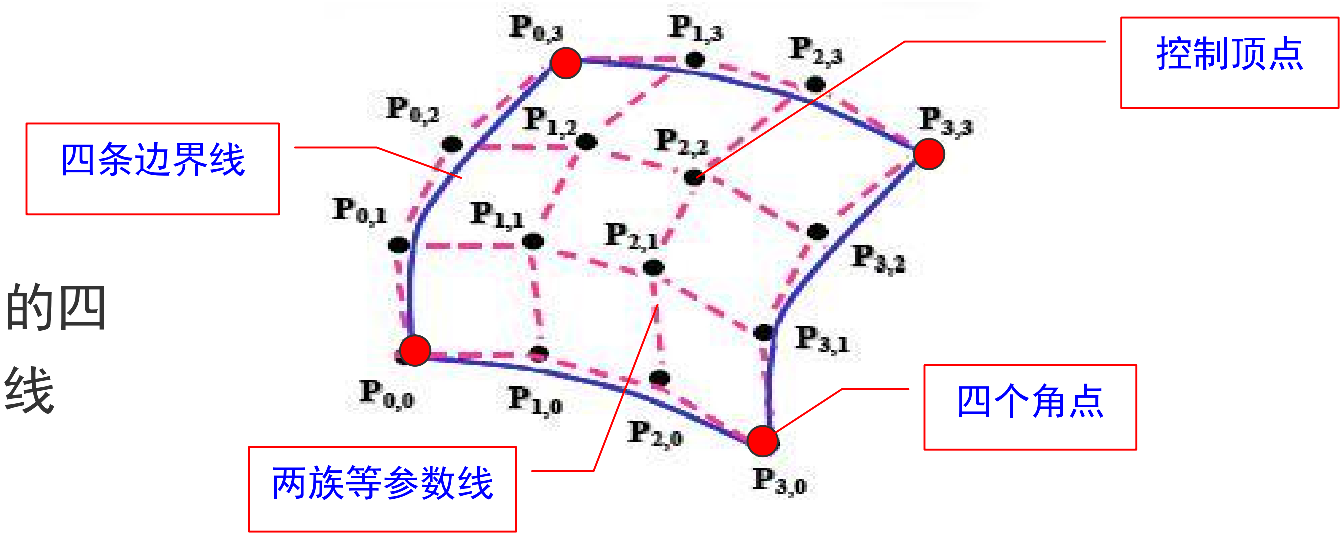
$$P(u,v)=\left[B_{0,n}(u),B_{1,n}(u),\ldots,B_{m,n}(u)\right]\begin{bmatrix} P_{00} & P_{01} & \ldots & P_{0m} \\ P_{10} & P_{11} & \ldots & P_{1m} \\ \ldots & \ldots & \ldots & \ldots \\ P_{n0} & P_{n1} & \ldots & P_{nm} \end{bmatrix}\begin{bmatrix} B_{0,m}(v) \\ B_{1,m}(v) \\ \ldots \\ B_{n,m}(v) \end{bmatrix}$$

上式的曲面称为m×n次的，在一般应用中，n、m不大于4



角点位置：四个角点分别是其控制网格的四个点

$$P(0,0) = P_{0,0} \quad P(0,1) = P_{0,n}$$
$$P(1,0) = P_{m,0} \quad P(1,1) = P_{m,n}$$



边界线：Bezier曲面的四条边界线是Bezier曲线

$$P(u,0) = \sum_{j=0}^{m} P_{j,0} BEZ_{j,m}(u) \qquad P(v,0) = \sum_{k=0}^{n} P_{k,0} BEZ_{k,n}(v)$$

$$P(u,1) = \sum_{j=0}^{m} P_{j,0} BEZ_{j,m}(u) \qquad P(v,1) = \sum_{k=0}^{n} P_{k,0} BEZ_{k,n}(v)$$

$$0 \le u \le 1 \qquad 0 \le v \le 1$$

2、Bezier曲面性质

Bezier曲线的很多性质可推广到Bezier曲面

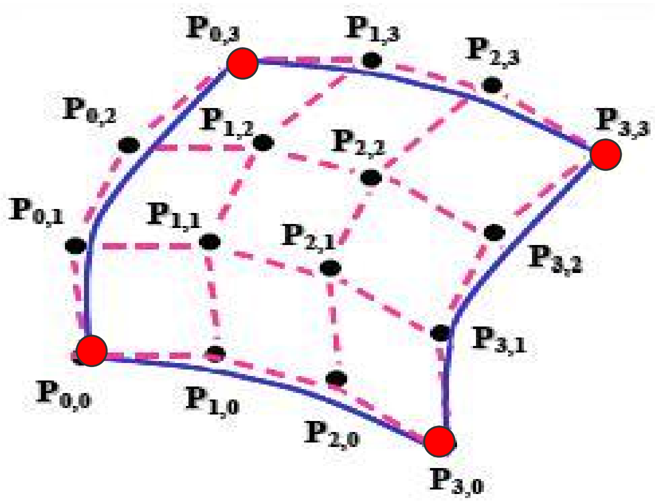
（1）Bezier曲面特征网格的四个角点正好是Bezier曲面的四个角点，即：

$$P(0,0)=P_{00}$$

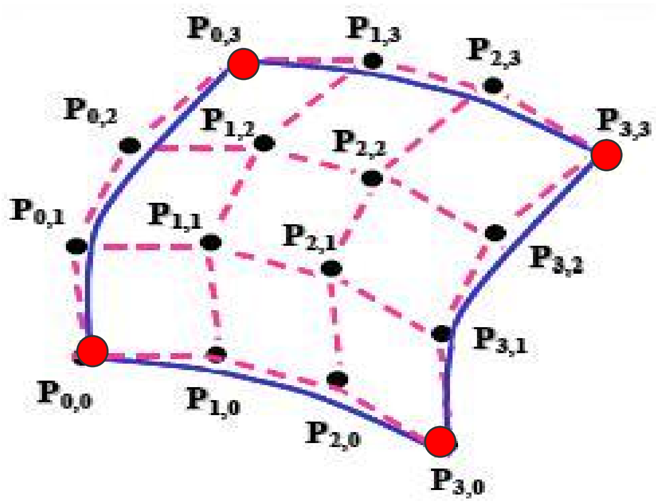
$$P(1,0)=P_{m0}$$

$$P(0,1)=P_{0n}$$

$$P(1,1)=P_{mn}$$



（2）Bezier曲面特征网格最外一圈顶点定义Bezier曲面的四条边界；



（3）几何不变性

（4）对称性

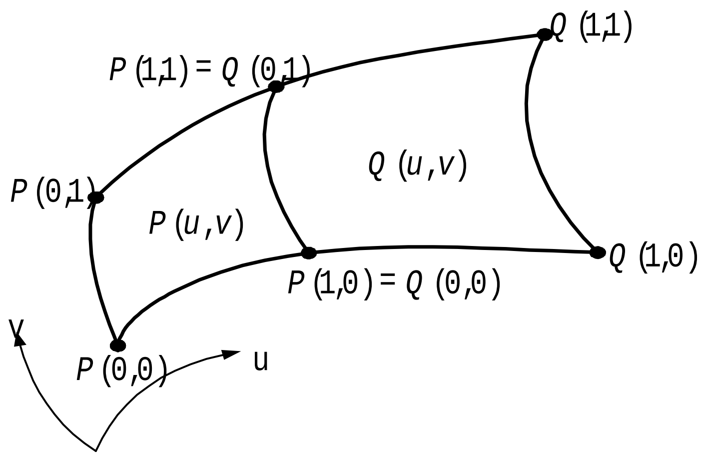
（5）凸包性

3、Bezier曲面片的拼接

设两张m×n次Bezier曲面片：

$$P(u,v)=\sum_{i=0}^{m}\sum_{j=0}^{n}P_{ij}B_{i,m}(u)B_{j,n}(v)$$

$$Q(u,v)=\sum_{i=0}^{m}\sum_{j=0}^{n}Q_{ij}B_{i,m}(u)B_{j,n}(v)\qquad u,v\in[0,1]$$



分别由控制顶点P_{ij}和Q_{ij}定义。

Bezier曲面片的拼接

如果要求两曲面片达到G^0连续，则它们有公共的边界，即：

$$P(1,v)=Q(0,v)$$

于是有： $P_{ni}=Q_{0i}$， $(i=0,1,\ldots,m)$

如果又要求沿该公共边界达到G^1连续，则两曲面片在该边界上有公共的切平面，因此曲面的法向应当是跨界连续的，即：

$$Q_u(0,v)\times Q_v(0,v)=a(v)P_u(1,v)\times P_v(1,v)$$

4、递推(de Casteljau)算法（曲面的求值）

Bezier曲线的递推(de Casteljau)算法，可以推广到Bezier曲面的情形

$$P_{ij}(i=0,1,\mathbf{L},m;\ j=0,1,\mathbf{L},n)$$

$$P(u,v)=\sum_{i=0}^{m-k}\sum_{j=0}^{n-l}P_{i,j}^{k,l}B_{i,m}(u)B_{j,n}(v)=\mathbf{L}=P_{00}^{m.n}\qquad u,v\in[0,1]$$

一条曲线可以表示成两条低一次曲线的组合，一张曲面可以表示成低一次的四张曲面的线性组合

其中：

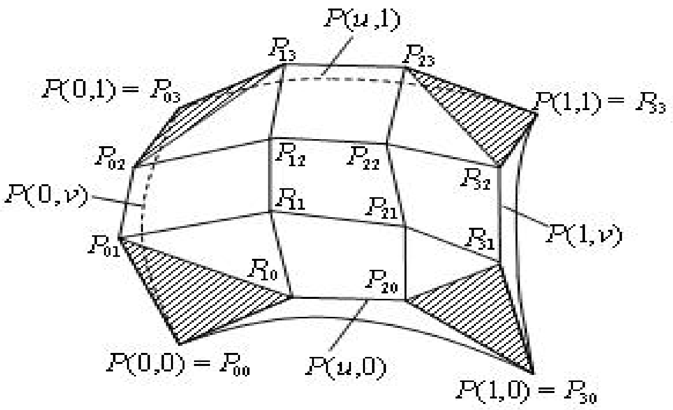
$$P_{i,j}^{k,l}=\begin{cases} P_{ij} & (k=l=0) \\ (1-u)P_{ij}^{k-1,0}+uP_{i+1,j}^{k-1,0} & (k=1,2,\mathbf{L}\ ,m;l=0) \\ (1-v)P_{0,j}^{m,l-1}+vP_{0,j+1}^{m,l-1} & (k=m,l=1,2,\mathbf{L}\ ,n) \end{cases} \tag{1}$$

或

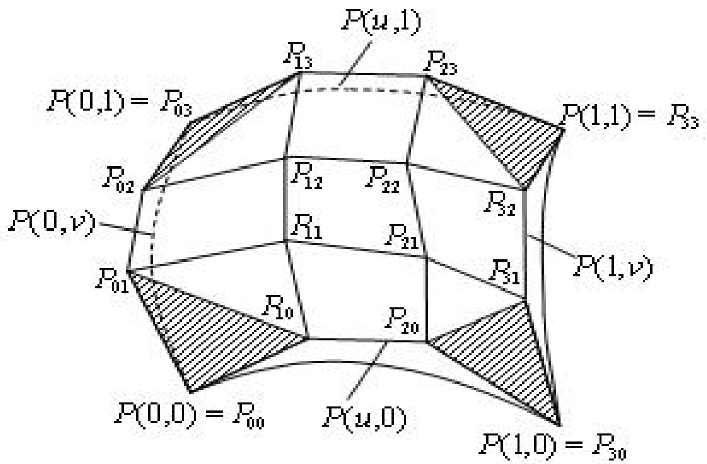
$$P_{ij}^{k,l}=\begin{cases} P_{ij} & (k=l=0) \\ (1-v)P_{ij}^{0,l-1}+vP_{i,j+1}^{0,l-1} & (k=0;l=1,2,\mathbf{L}\ ,n) \\ (1-u)P_{i0}^{k-1,n}+uP_{i+1,0}^{k-1,n} & (k=1,2,\mathbf{L}\ ,m;l=n) \end{cases} \tag{2}$$

$$P_{i,j}^{k,l}=\begin{cases} P_{ij} & (k=l=0) \\ (1-u)P_{ij}^{k-1,0}+uP_{i+1,j}^{k-1,0} & (k=1,2,\cdots,m;l=0) \\ (1-v)P_{0,j}^{m,l-1}+vP_{0,j+1}^{m,l-1} & (k=m,l=1,2,\cdots,n) \end{cases} \quad (1)$$

当按(1)式方案执行时，先以u参数值对控制网格u向的n+1个多边形执行曲线de Casteljau算法，m级递推后，得到沿v向由n+1个顶点 P_{0j}^{m0} 构成的多边形

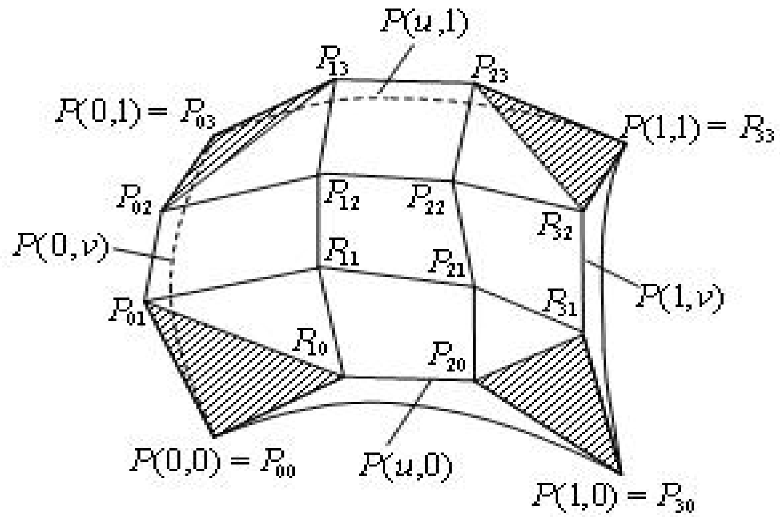


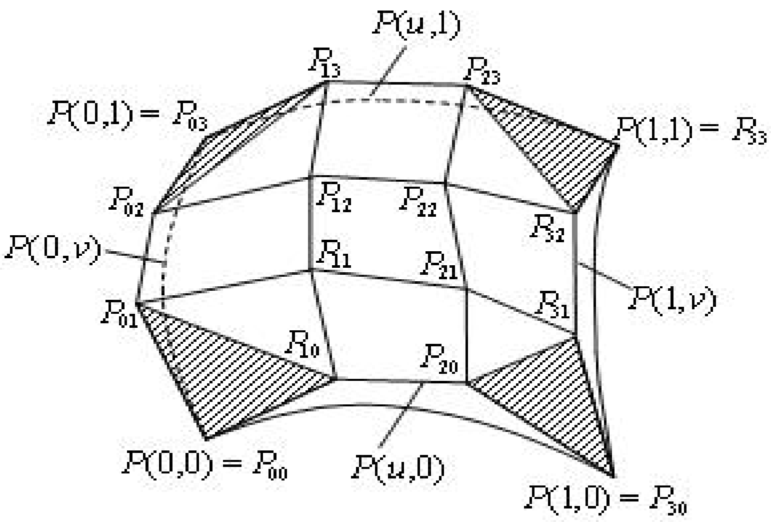
再以v参数值对它执行曲线的de Casteljau算法，n级递推以后，得到一个 P_{00}^{mn}，即所求曲面上的点。



一条曲线可以表示成2条低一次的曲线的线性组合，曲面可以表示成低一次的4张曲面的线性组合。

这张曲面有16个顶点，现在要算曲面上一点 P（1/2，1/2）的值，如何计算？





首先拿u＝1/2来计算，得到4个点。再拿这四个点做为新的Bezier曲线的控制顶点，按v＝1/2来计算，算出的这个点就是所要求的值

之所以有这样的算法，是因为：

$$P(u,v)=\sum_{i=0}^{m}\sum_{j=0}^{n}P_{ij}B_{i,m}(u)B_{j,n}(v)$$

$$=\sum_{i=0}^{m}\left(\sum_{j=0}^{n}P_{ij}B_{j,n}(v)\right)B_{i,m}(u)$$

B样条曲线与曲面

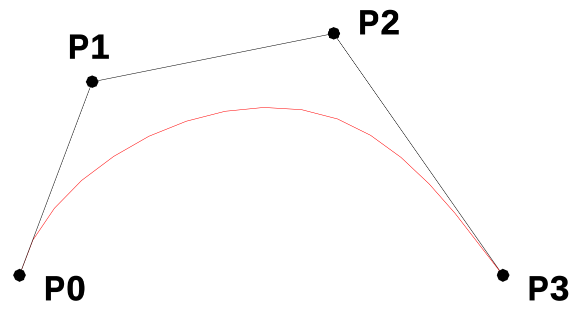
一、B样条产生的背景

Bezier曲线曲面有很多优点，可以用鼠标拖动控制顶点以改变曲线的形状，非常直观，给设计人员很大的自由度

Bezier曲线曲面是几何造型的主要方法和工具

但是Bezier曲线有几点不足:

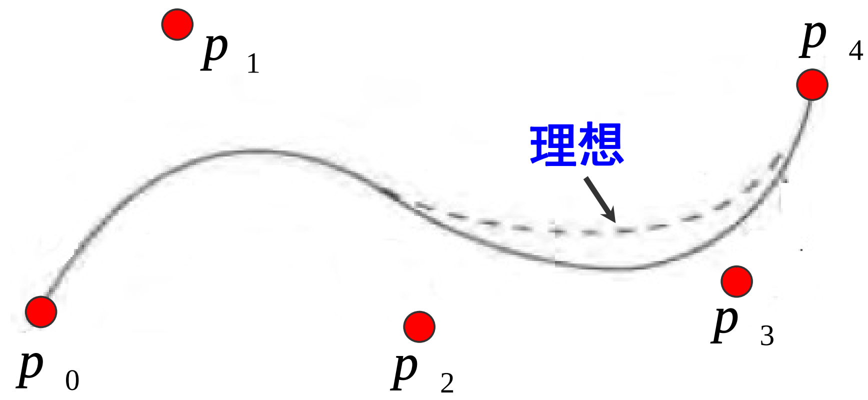
（1）一旦确定了特征多边形的顶点数(n+1个)，也就决定了曲线的阶次(n次)



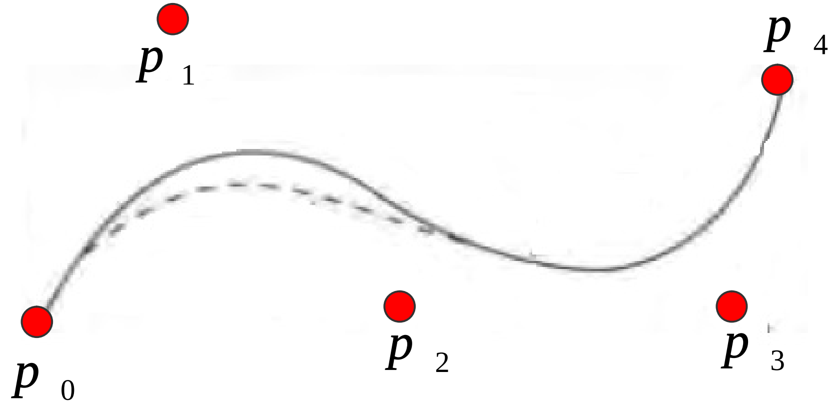
（2）Bezier曲线或曲面的拼接比较复杂

(3）Bezier曲线或曲面不能作局部修改

为了纠正偏离，需要向上移动p_2和p_3，以使Bezier曲线更接近所期望的曲线

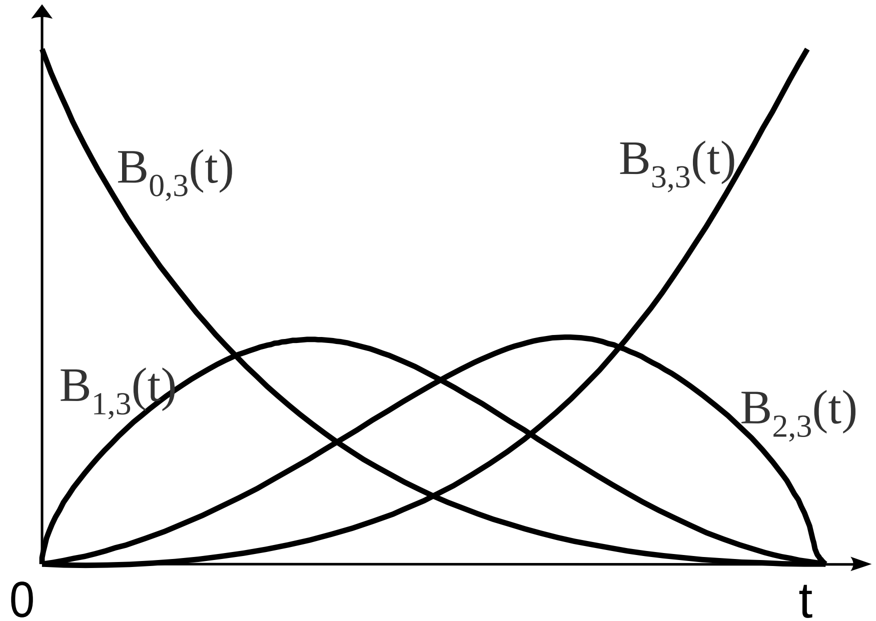


但是，这也影响了曲线的前半部分，迫使它偏离所期望的曲线



函数值不为0的区间通常叫做它的[支撑区间]

因为每个Bernstein多项式在整个区间[0，1]上都有支撑，且曲线是这些函数的混合，所以每个控制顶点对0到1之间的t值处的曲线都有影响

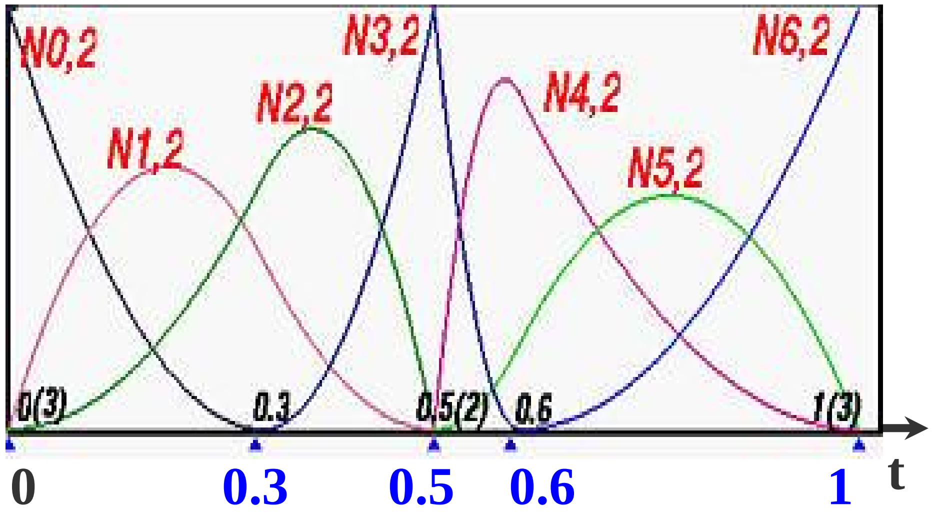


三次Bezier曲线四个Bezier基函数

图中显示了7个混合函数

$N_{0,2}$ $N_{1,2}$ $N_{2,2}$ $N_{3,2}$

$N_{4,2}$ $N_{5,2}$ $N_{6,2}$



每个函数只在区间[0,1]上的一部分有支撑，如：

$N_{1,2}$的支撑是[0,0.5]　　$N_{4,2}$的支撑是[0.5,1]

1972年，Gordon、Riesenfeld等人提出了B样条方法，在保留Bezier方法全部优点的同时，克服了Bezier方法的弱点

样条（spline）—— 分段连续多项式！

整条曲线用一个完整的表达形式，但内在的量是一段一段的，比如一堆的3次曲线拼过去，两条之间满足2次连续

这样既克服了波动现象，曲线又是低次的。既有统一的表达时，又有统一的算法

如何进行分段呢？

现在有n+1个点，每两点之间构造一条多项式，n+1个点有n个小区间

每个小区间构造一条三次多项式，变成了n段的三次多项式拼接在一起，段与段之间要两次连续，这就是三次样条

如有5个点，构造一个多项式，应该是个四次多项式。现在采用样条方式构造四段曲线，每一段都是三次的，且段与段之间要C^2连续。

二、B样条的递推定义和性质

B样条曲线的数学表达式为:

$$P(u) = \sum_{i=0}^{n} P_i B_{i,k}(u) \qquad u \in [u_{k-1}, u_{n+1}]$$

$P_i (i = 0,1,\mathbf{L}\ ,n)$ 是控制多边形的顶点

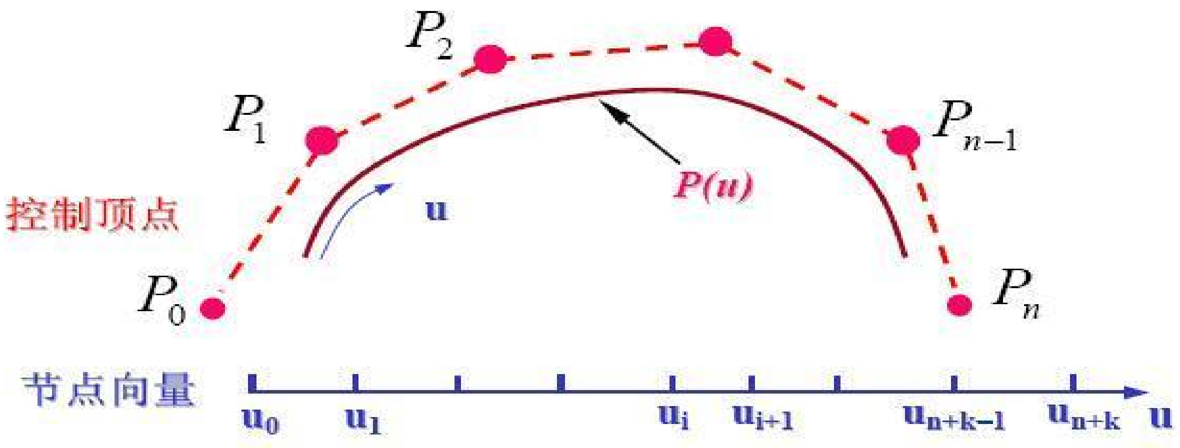
Bezier曲线 $$p(u) = \sum_{i=0}^{n} P_i B_{i,n}(u) \qquad u \in [0,1]$$

$B_{i,k}(u)$ 称为k阶（k-1次）B样条基函数，k是刻画次数的。其中k可以是2到控制点个数n+1之间的任意整数

对Bezier曲线来说，阶数和次数是一样的；但对B样条，阶数是次数加1

$$P(u) = \sum_{i=0}^{n} P_i B_{i,k}(u) \qquad u \in [u_{k-1}, u_{n+1}]$$

B样条基函数是一个称为节点矢量的非递减的参数u的序列所决定的k阶分段多项式，这个序列称为节点向量



$$u_0, u_1, \mathbf{L}, u_{k-1}, u_k, \mathbf{L}, u_n, u_{n+1}, \mathbf{L}, u_{n+k-1}, u_{n+k}$$

B样条基函数实际上就是一个多项式，一个比较复杂的、有特点的多项式而已。如何得到这个B样条基函数？

de Boor-Cox递推定义

B样条基函数可以有各种各样的定义方式，但是公认的最容易理解的是de Boor-Cox递推定义

它的原理是，只要是k阶（k-1次）的B样条基函数，构造一种递推的公式，由0次构造1次，1次构造2次，2次构造3次...依次类推

$$B_{i,1}(u)=\begin{cases}1 & u_i < u < u_{i+1}\\ 0 & Otherwise\end{cases}$$

并约定：$\frac{0}{0}=0$

$$B_{i,k}(u)=\frac{u-u_i}{u_{i+k-1}-u_i}B_{i,k-1}(u)+\frac{u_{i+k}-u}{u_{i+k}-u_{i+1}}B_{i+1,k-1}(u)$$

该递推公式表明：若确定第i个k阶B样条$B_{i,k}(u)$，需要用到$u_i,u_{i+1},...,u_{i+k}$共k+1个节点，称区间$[u_i,u_{i+k}]$为$B_{i,k}(u)$的支承区间

$$B_{i,1}(u)=\begin{cases}1 & u_i < u < u_{i+1}\\ 0 & Otherwise\end{cases}$$

并约定：$\frac{0}{0}=0$

$$B_{i,k}(u)=\frac{u-u_i}{u_{i+k-1}-u_i}B_{i,k-1}(u)+\frac{u_{i+k}-u}{u_{i+k}-u_{i+1}}B_{i+1,k-1}(u)$$

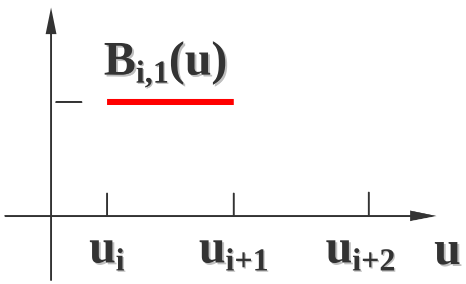
曲线方程中，n+1个控制顶点P_i(i=0,1,...,n)，要用到n+1个k阶B样条$B_{i,k}(u)$。它们支撑区间的并集定义了这一组B样条基的节点矢量 $U=[u_0,u_1,...,u_{n+k}]$

$$u_0,u_1,\cdots,u_{k-1},u_k,\cdots,u_n,u_{n+1},\cdots,u_{n+k-1},u_{n+k}$$

$$B_{i,1}(u)=\begin{cases}1 & u_i < u < u_{i+1} \\ 0 & Otherwise\end{cases}$$

$$B_{i,k}(u)=\frac{u-u_i}{u_{i+k-1}-u_i}B_{i,k-1}(u)+\frac{u_{i+k}-u}{u_{i+k}-u_{i+1}}B_{i+1,k-1}(u)$$

$B_{i,1}(u)$是0次多项式



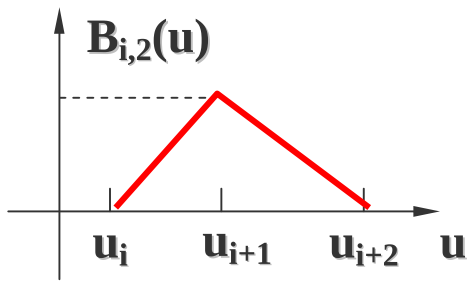
（1阶B样条）

2阶（一次）B样条$B_{i,2}(u)$?

$$B_{i,2}(u)=\frac{u-u_i}{u_{i+1}-u_i}B_{i,1}(u)+\frac{u_{i+2}-u}{u_{i+2}-u_{i+1}}B_{i+1,1}(u) \qquad B_{i,1}(u)=\begin{cases}1 & u_i<u<u_{i+1}\\ 0 & Otherwise\end{cases}$$

$$B_{i+1,1}(u)=\begin{cases}1 & u_{i+1}<u<u_{i+2}\\ 0 & Otherwise\end{cases}$$

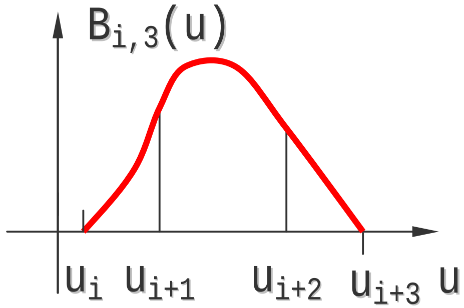
$$B_{i,2}(u)=\begin{cases}\dfrac{u-u_i}{u_{i+1}-u_i} & u_i\le u\le u_{i+1}\\ \dfrac{u_{i+2}-u}{u_{i+2}-u_{i+1}} & u_{i+1}\le u\le u_{i+2}\\ 0 & \text{其它}\end{cases}$$



（2阶B样条）

一次B样条$B_{i,2}(u)$可有两个0次B样条$B_{i,1}(u)$和$B_{i+1,1}(u)$递推得到，是它们的凸线性组合

再有两个一次B样条$B_{i,2}(u)$ 和$B_{i+1,2}(u)$递推得到二次B样条$B_{i,3}(u)$



$B_{i,3}(u)$**的图像**

每个p_i都有一个$B_{i,k}(u)$与之匹配，有n+1个$B_{i,k}(u)$，曲线的次数是k-1次，问题是这条曲线的定义区间是什么？

Bezier曲线的定义区间是[0，1]？

第二个问题是对n+1个顶点，k阶的B样条曲线需要多少个节点向量（u_i）与之匹配？

三、B样条基函数定义区间及节点向量

1、B样条曲线定义区间是什么?

Bezier曲线的定义区间是[0, 1]

2、第二个问题是对n+1个顶点，k阶的B样条曲线需要多少个节点向量（u_i）与之匹配?

1、K阶B样条对应节点向量数

$$B_{i,1}(u)=\begin{cases}1 & u_i < u < u_{i+1}\\ 0 & Otherwise\end{cases}\qquad B_{i,k}(u)=\frac{u-u_i}{u_{i+k-1}-u_i}B_{i,k-1}(u)+\frac{u_{i+k}-u}{u_{i+k}-u_{i+1}}B_{i+1,k-1}(u)$$

对于$B_{i,1}$（1阶0次基函数）来说，涉及u_i到u_{i+1}一个区间，即一阶的多项式涉及一个区间两个节点

$B_{i,2}$是由$B_{i,1}$和$B_{i+1,1}$组成，因此$B_{i,2}$涉及2个区间3个节点；$B_{i,3}$涉及3个区间4个节点...，$B_{i,k}$涉及k个区间k+1个节点

$$u_0, u_1, \ldots, u_{k-1}, u_k, \ldots, u_n, u_{n+1}, \ldots, u_{n+k-1}, u_{n+k}$$

2、B样条函数定义区间

$$P(u)=\sum_{i=0}^{n} P_i B_{i,k}(u) \qquad B_{i,1}(u)=\begin{cases}1 & u_i < x < u_{i+1} \\ 0 & Otherwise\end{cases}$$

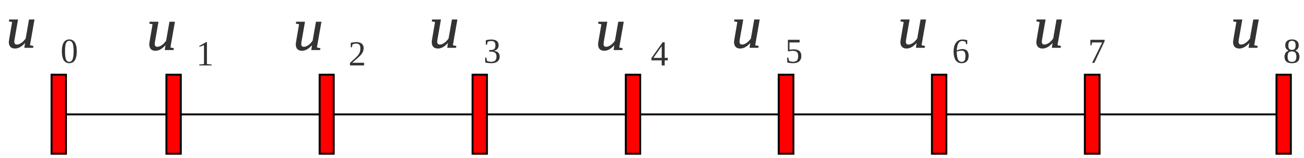
$$u \in [u_{k-1}, u_{n+1}] \qquad B_{i,k}(u)=\frac{u-u_i}{u_{i+k-1}-u_i}B_{i,k-1}(u)+\frac{u_{i+k}-u}{u_{i+k}-u_{i+1}}B_{i+1,k-1}(u)$$

$$u_0, u_1, \ldots, u_{k-1}, u_k, \ldots, u_n, u_{n+1}, \ldots, u_{n+k-1}, u_{n+k}$$

以k=4，n=4为例　　节点矢量为：

$$U=\{u_0, u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8\}$$

$$U = \{u_0, u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8\}$$



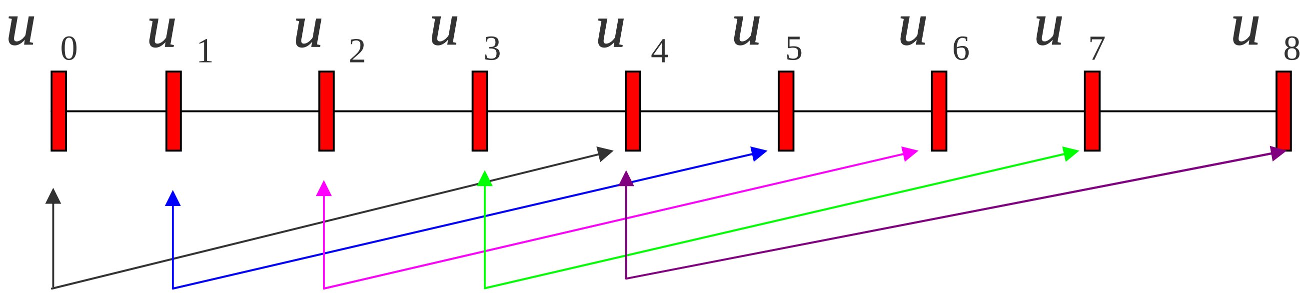
$n=4 \quad k=4$

$$B_{i,k}(u)=\frac{u-u_i}{u_{i+k-1}-u_i}B_{i,k-1}(u)+\frac{u_{i+k}-u}{u_{i+k}-u_{i+1}}B_{i+1,k-1}(u)$$

第一项是$p_0B_{0,4}(u)$，涉及哪些节点？ u_0到u_4

第二项是$p_1B_{1,4}(u)$，涉及哪些节点？ u_1到u_5

第五项是$p_4B_{4,4}(u)$，涉及哪些节点？ u_4到u_8

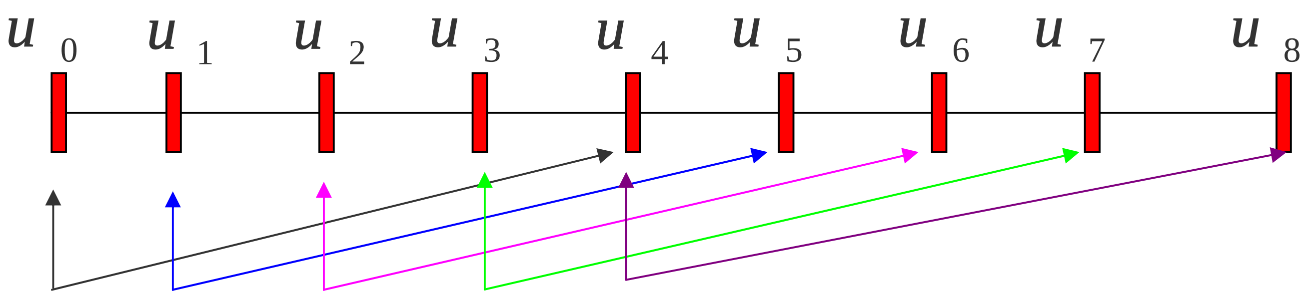


$P_0B_{0,4}(u)$，涉及u_0到u_4个节点　$P_3B_{3,4}(u)$，涉及u_3到u_7个节点

$P_1B_{1,4}(u)$，涉及u_1到u_5个节点　$P_4B_{4,4}(u)$，涉及u_4到u_8个节点

$P_2B_{2,4}(u)$，涉及u_2到u_6个节点

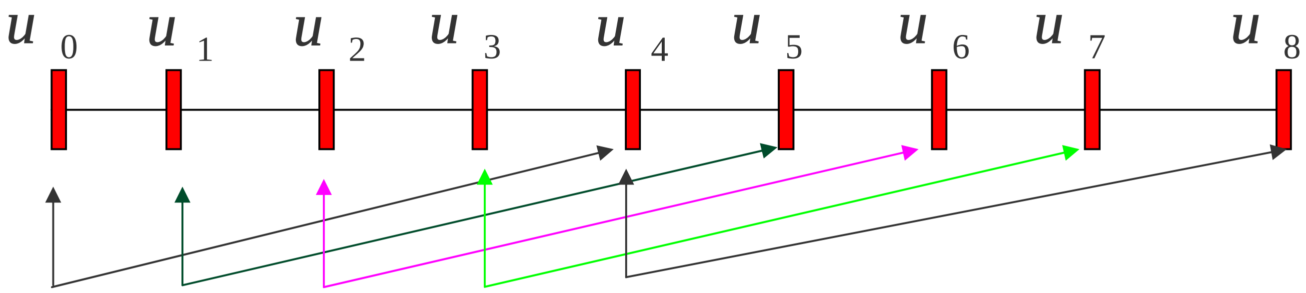
“阶数+顶点”等于节点向量的个数。问函数的定义区间？



区间要合法，区间里必须要有足够的基函数与顶点配对

哪个区间是第一个开始有意义的区间呢？

上图区间是u_3到u_5（从u_{k-1}到u_{n+1}），B样条基函数严重依赖于节点向量的分布



上面的曲线被分成两段：u_3u_4，u_4u_5。如有5个顶点P_0、P_1、P_2、P_3、P_4，B样条是一段段过渡过去

哪些基函数是在u_3u_4区间里有定义？　　正好是$P_0P_1P_2P_3$

$P_1P_2P_3P_4$在u_4u_5区间里有定义，两端之间有三个顶点是一样的，这样就保证了两段拼接的效果非常好

B样条曲线定义：

$$P(u)=\sum_{i=0}^{n} P_i B_{i,k}(u)$$

$$B_{i,1}(u)=\begin{cases}1 & u_i < x < u_{i+1} \\ 0 & Otherwise\end{cases}$$

$$u \in [u_{k-1}, u_{n+1}]$$

$$B_{i,k}(u)=\frac{u-u_i}{u_{i+k-1}-u_i}B_{i,k-1}(u)+\frac{u_{i+k}-u}{u_{i+k}-u_{i+1}}B_{i+1,k-1}(u)$$

u_i是节点值，U=（u_0,u_1,…u_{n+k}）构成了k阶（k-1次）B样条函数的节点矢量

B样条曲线所对应的节点向量区间： $u \in [u_{k-1}, u_{n+1}]$

四、B样条基函数的主要性质

1、局部支承性：

$$B_{i,k}(u)\begin{cases} \geq 0 & u \in [u_i, u_{i+k}] \\ = 0 & otherwise \end{cases}$$

而Bezier在整个区间非0

反过来，对每一个区间(u_i, u_{i+k})，至多只有k个基函数在其上非零

2、权性：

$$\sum_{i=0}^{n} B_{i,k}(u) \equiv 1 \qquad\qquad u \in [u_{k-1}, u_{n+1}]$$

3、连续性

$B_{i,k}(u)$在r重节点处的连续阶不低于 k-1-r

4、分段参数多项式:

$B_{i,k}(u)$在每个长度非零的区间$[u_i,u_{i+1})$上都是次数不高于k-1的多项式，它在整个参数轴上是分段多项式

五、B样条函数的主要性质

1、局部性:

k阶B样条曲线上的一点至多与k个控制顶点有关，与其它控制顶点无关

移动曲线的第i个控制顶点P_i，至多影响到定义在区间上那部分曲线的形状，对曲线其余部分不发生影响

2、变差缩减性：

设平面内 n+1 个控制顶点 构成B样条曲线 P(t) 的特征多边形。在该平面内的任意一条直线与 P(t) 的交点个数不多于该直线和特征多边形的交点个数

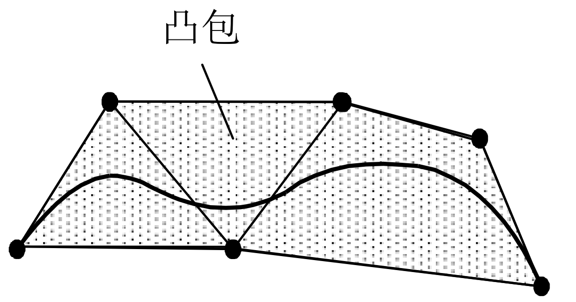
3、几何不变性：

B样条曲线的形状和位置与坐标系的选择无关

4、凸包性：

B样条曲线落在P_i构成的凸包之中。其凸包区域小于或等于同一组控制顶点定义的Bezier曲线凸包区域

凸包就是包含右边这6个顶点的最小凸多边形。凸多边形是把多边形的每条边延长，其它边都在它的同一侧



Bezier曲线的凸包性

该性质导致顺序k+1个顶点重合时，由这些顶点定义的k次B样条曲线段退化到这一个重合点；顺序k+1个顶点共线时，由这些顶点定义的k次B样条曲线形状？

六、B样条曲线类型的划分

1、均匀B样条曲线（uniform B-spline curve）

当节点沿参数轴均匀等距分布，即$u_{i+1}-u_i$ = 常数 > 0时，表示均匀B样条函数

$$\{0,1,2,3,4,5,6\}$$

$$\{0,0.2,0.4,0.6,0.6,0.8,1\}$$

均匀B样条的基函数呈周期性。即给定n和k，所有的基函数有相同形状。每个后续基函数仅仅是前面基函数在新位置上的重复：

$$B_{i,k}(u) = B_{i+1,k}(u+\Delta u) = B_{i+2,k}(u+2\Delta u)$$

其中，△u是相邻节点值的间距，等价地，也可写为：

$$B_{i,k}(u) = B_{0,k}(u - k\Delta u)$$

下面以均匀二次（三阶）B样条曲线为例来说明均匀周期性B样条基函数的计算：

假定有四个控制点，取参数值 n=3，k=3，则 n+m=6

$$u = (0,1,2,3,4,5,6)$$

根据de Boor-Cox递推公式：

$$P(u)=\sum_{i=0}^{n} P_i B_{i,k}(u)$$

$$B_{i,1}(u)=\begin{cases}1 & u_i < x < u_{i+1}\\ 0 & Otherwise\end{cases}$$

$$B_{i,k}(u)=\frac{u-u_i}{u_{i+k-1}-u_i}B_{i,k-1}(u)+\frac{u_{i+k}-u}{u_{i+k}-u_{i+1}}B_{i+1,k-1}(u)$$

$$B_{0,1}(u)=\begin{cases}1 & 0\le \mathrm{u} < 1\\ 0 & \text{其它}\end{cases}$$

$$u=(0,1,2,3,4,5,6)$$

$$B_{0,2}(u)=uB_{0,1}(u)+(2-u)B_{1,1}(u)=uB_{0,1}(u)+(2-u)B_{1,1}(u)$$

$$=\begin{cases}u & 0\le \mathrm{u}<1\\ 2-u & 1\le \mathrm{u}<2\end{cases}$$

$$B_{0,3}(u) = \frac{u}{2} B_{0,2}(u) + \frac{3-u}{2} B_{1,2}(u-1)$$

$$= \begin{cases} \frac{1}{2}u^2 & 0 \le u < 1 \\ \frac{1}{2}u(2-u) + \frac{1}{2}(u-1)(3-u) & 1 \le u < 2 \\ \frac{1}{2}(3-u)^2 & 2 \le u < 3 \end{cases}$$

$$B_{1,3}(u) = \begin{cases} \frac{1}{2}(u-1)^2 & 1 \le u < 2 \\ \frac{1}{2}(u-1)(3-u) + \frac{1}{2}(u-2)(4-u) & 2 \le u < 3 \\ \frac{1}{2}(4-u)^2 & 3 \le u < 4 \end{cases}$$

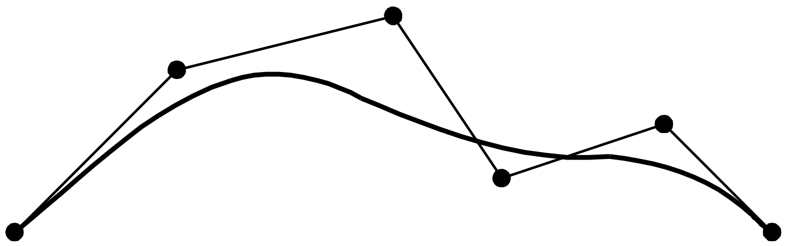
$$B_{3,3}(u) = \begin{cases} \frac{1}{2}(u-3)^2 & 3 \le u < 4 \\ \frac{1}{2}(u-3)(5-u) + \frac{1}{2}(u-4)(6-u) & 4 \le u < 5 \\ \frac{1}{2}(6-u)^2 & 5 \le u < 6 \end{cases}$$

2、准均匀B样条曲线（Quasi-uniform B-spline curve）

与均匀B样条曲线的差别在于两端节点具有重复度k，这样的节点矢量定义了准均匀的B样条基

均匀：$u = (0,1,2,3,4,5,6)$

准均匀：$u = (0,0,0,1,2,3,4,5,5,5)$

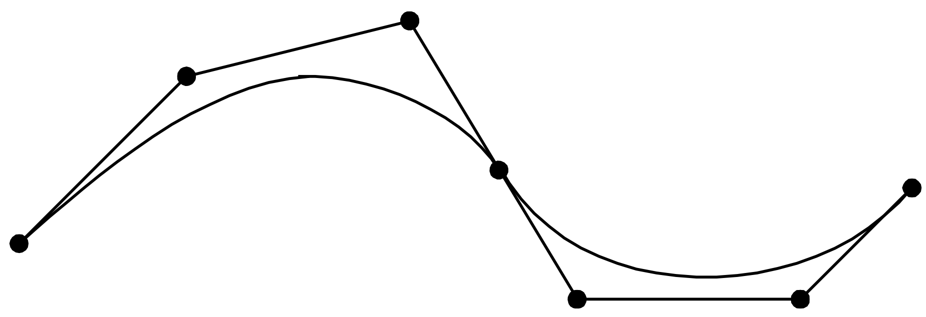


准均匀三次B样条曲线

均匀B样条曲线没有保留Bezier曲线端点的几何性质，采用准均匀的B样条曲线解决了这个问题

3、分段Bezier曲线（Piecewise Bezier Curve）

节点矢量中两端节点具有重复度k，所有内节点重复度为k-1，这样的节点矢量定义了分段的Bernstein基



三次分段Bezier曲线

B样条曲线用分段Bezier曲线表示后，各曲线段就具有了相对的独立性

另外，Bezier曲线一整套简单有效的算法都可以原封不动地采用

缺点是增加了定义曲线的数据，控制顶点数及节点数

4、非均匀B样条曲线（non-uniform B-spline curve）

当节点沿参数轴的分布不等距，即$u_{i+1}-u_i \neq$常数时，表示非均匀B样条函数

七、B样条曲面

给定参数轴u和v的节点矢量

$$U = [u_0, u_1, \cdots, u_{m+p}]$$

$$V = [v_0, v_1, \cdots, v_{n+q}]$$

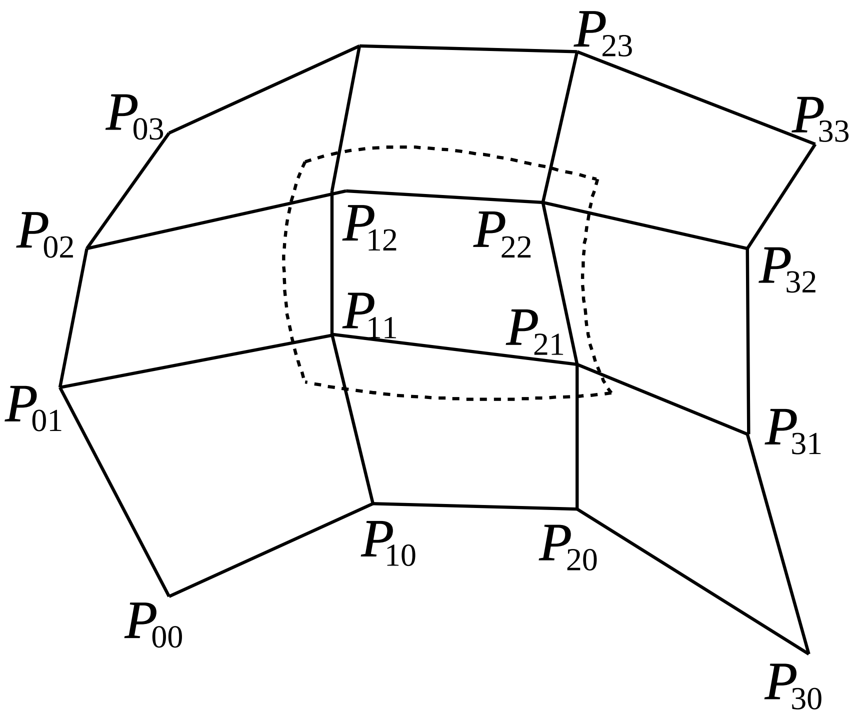
p×q阶B样条曲面定义如下:

$$P(u,v) = \sum_{i=0}^{m} \sum_{j=0}^{n} P_{i,j} N_{i,p}(u) N_{j,q}(v)$$

$$P(u,v)=\sum_{i=0}^{m}\sum_{j=0}^{n}P_{i,j}N_{i,p}(u)N_{j,q}(v)$$

$P_{i,j}$ 构成一张控制网格，称为B样条曲面的特征网格

$N_{i,p}(u)$ 和 $N_{j,q}(v)$ 是B样条基，分别由节点矢量U和V按deBoor-Cox递推公式决定



双三次B样条曲面片