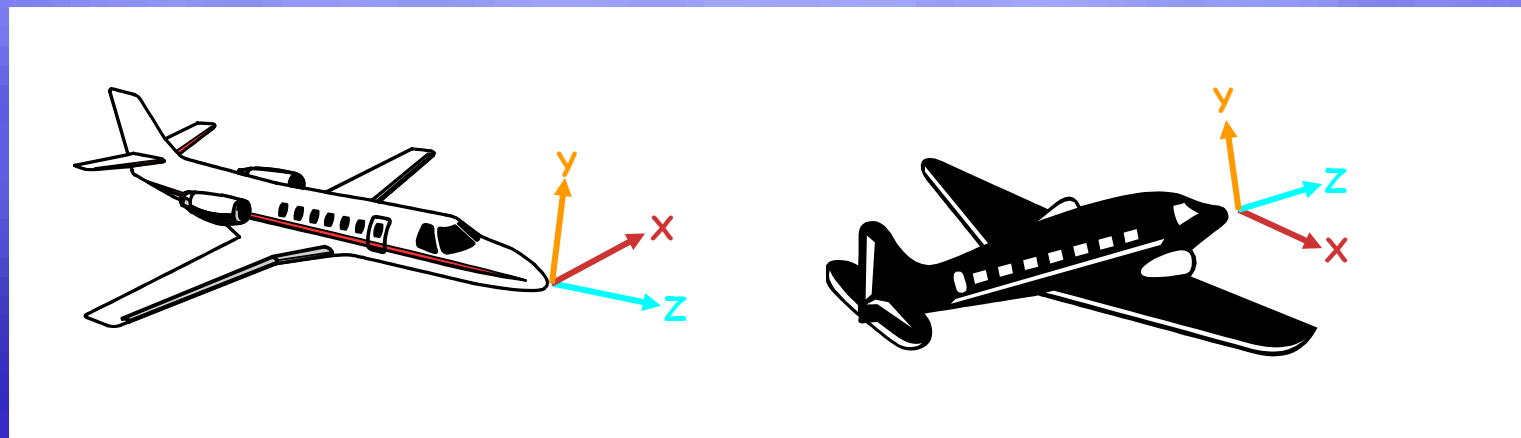
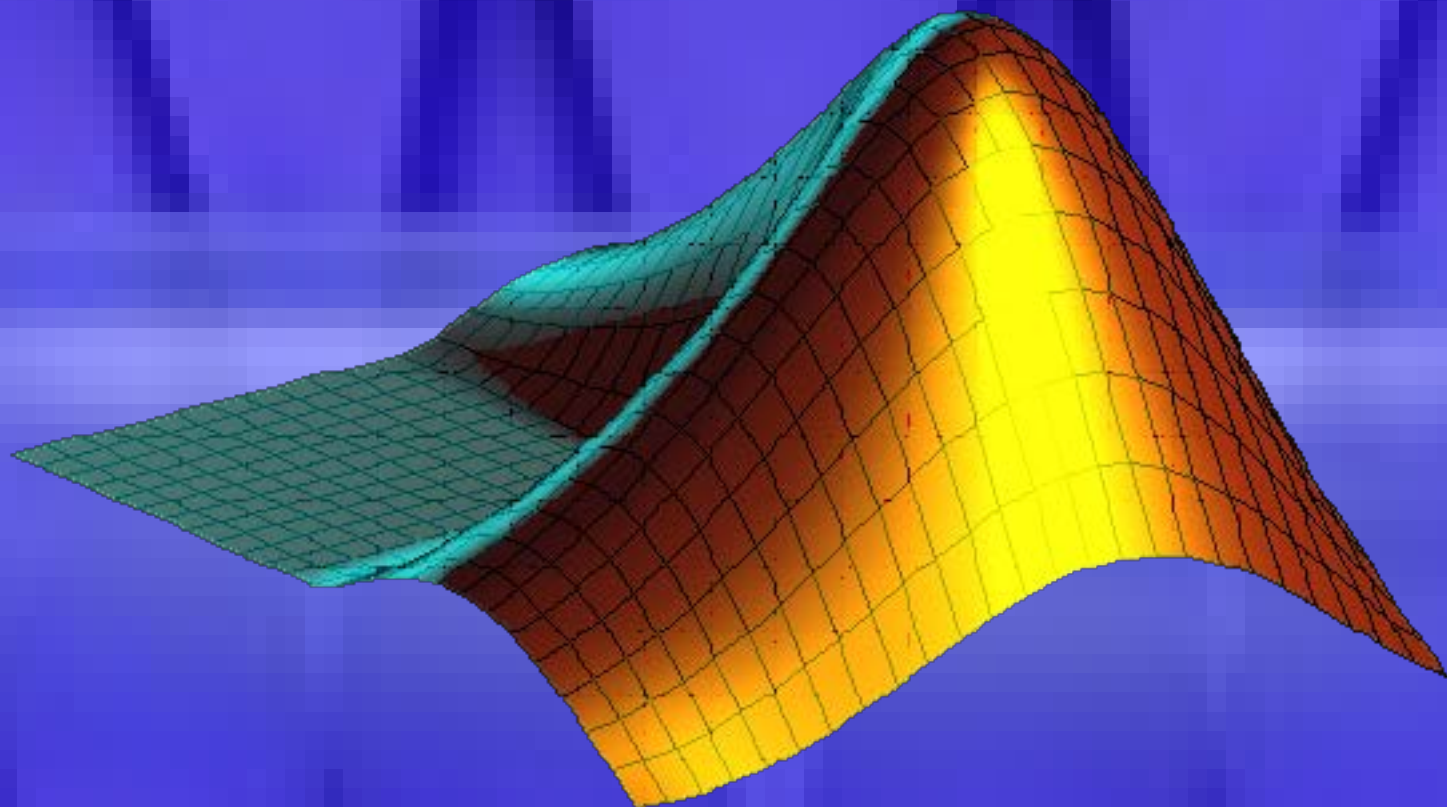


3.1.2 Bezier 曲线与曲面

一个复杂的计算机图形学问题!!



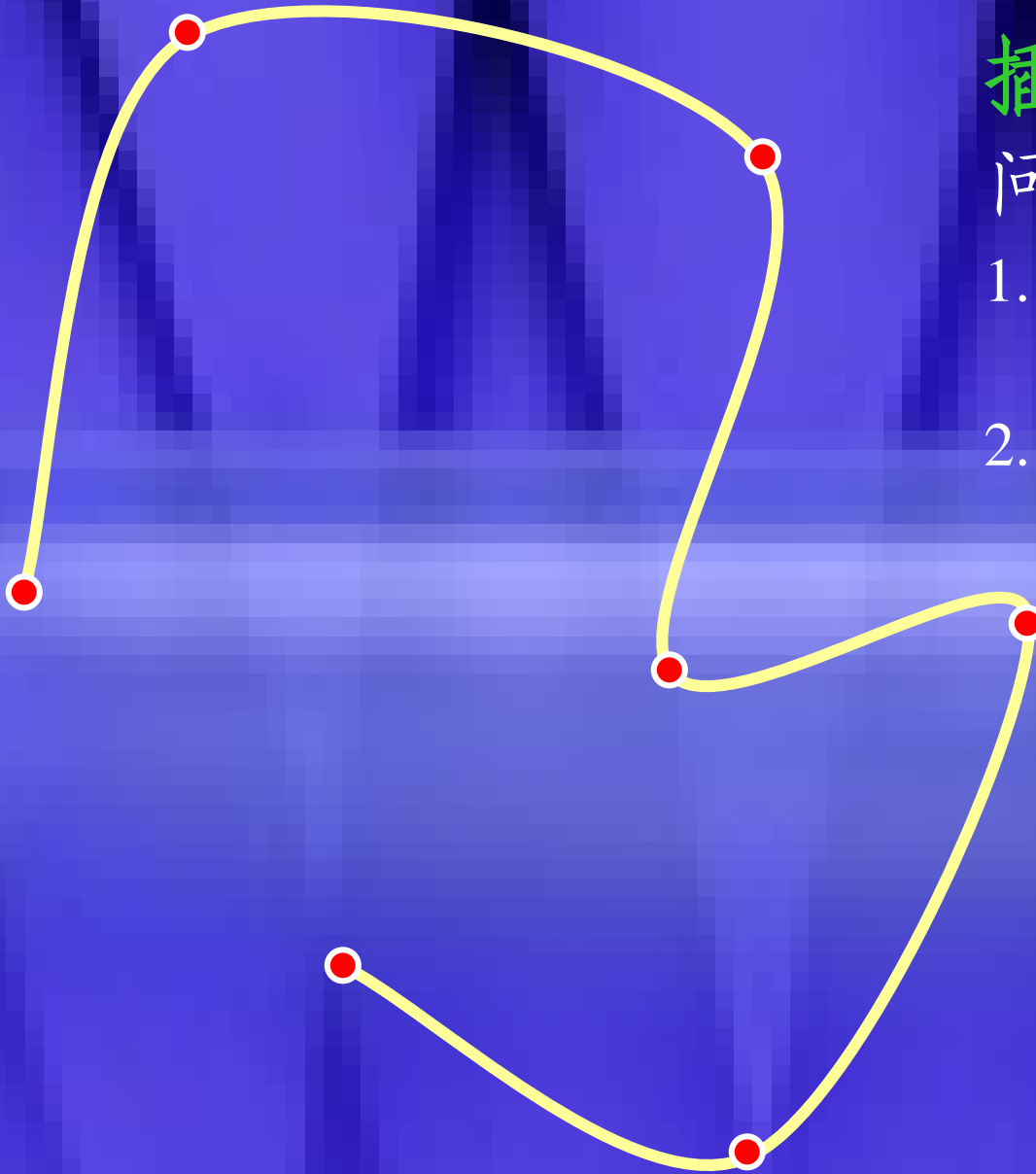
构造曲面！



插值曲线

问题？

1. Lagrange插值，点多，次数高
2. 不适合于设计



3.1.2 Bezier 曲线与曲面

由于几何外形设计的要求越来越高,传统的曲线曲面表示方法,已不能满足用户的需求。1962年,法国雷诺汽车公司的**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,1972年,该系统被投入了应用。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。

3.1.2.1 Bezier曲线的定义和性质

1. 定义

给定空间 $n+1$ 个点的位置矢量 $P_i (i = 0, 1, 2, \dots, n)$, 则Bezier参数曲线上各点坐标的插值公式是:

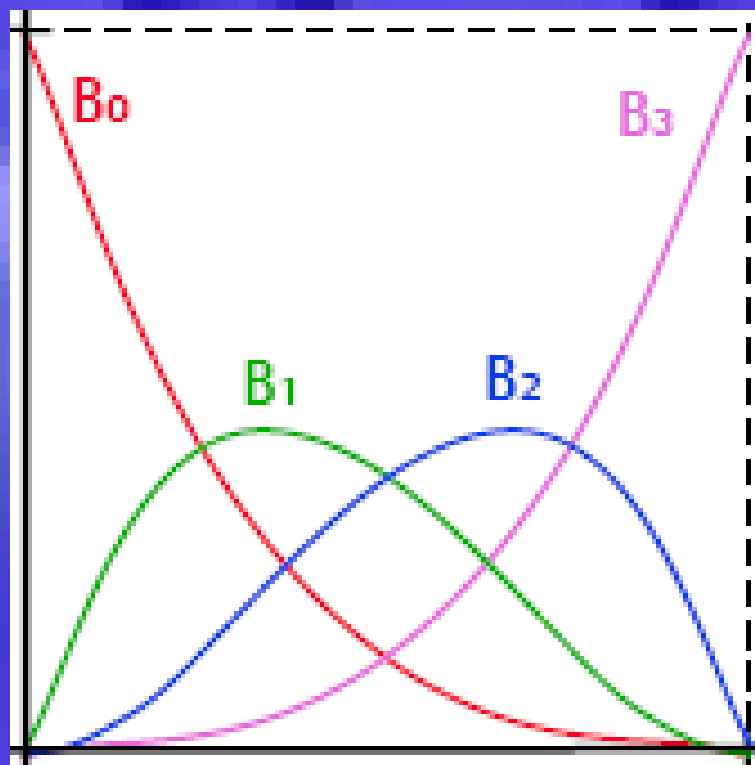
$$P(t) = \sum_{i=0}^n P_i B_{i,n}(t), \quad t \in [0,1]$$

其中, P_i 构成该Bezier曲线的特征多边形, $B_{i,n}(t)$ 是 n 次Bernstein基函数:

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

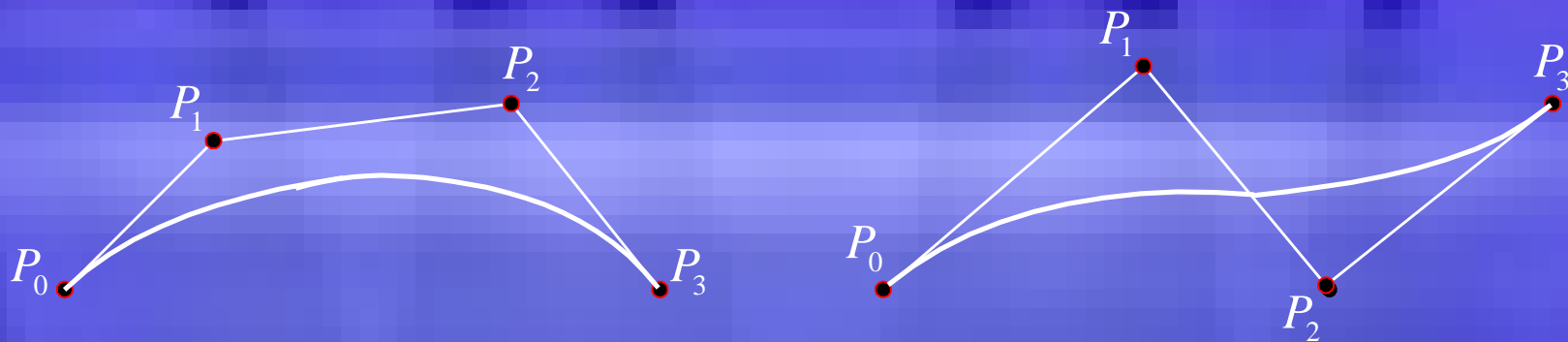
$$0^0=1, 0!=1$$

$n = 3, i = 0, 1, 2, 3$ 时的Bernstein基函数(鼓励用
Matlab软件验证):



Matlab计算组合
函数nchoosek (n,k)

Bezier曲线实例如下图所示:



三次Bezier曲线

2. Bernstein基函数的性质

(1) 正性

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

(2) 端点性质

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

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

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

由二项式定理可知:

$$\sum_{i=0}^n B_{i,n}(t) = \sum_{i=0}^n C_n^i t^i (1-t)^{n-i} = [(1-t) + t]^n \equiv 1$$

(4) 对称性 $B_{i,n}(1-t) = B_{n-i,n}(t)$

因为
$$\begin{aligned} B_{n-i,n}(t) &= C_n^{n-i} t^{n-i} \cdot (1-t)^{n-(n-i)} \\ &= C_n^i t^i (1-t)^i [1-(1-t)]^{n-i} = B_{i,n}(1-t) \end{aligned}$$

(5) 递推性

$$B_{i,n}(t) = (1-t)B_{i,n-1}(t) + tB_{i-1,n-1}(t) \quad (i = 0, 1, \dots, n)$$

即高一次的 Bernstein 基函数可由两个低一次的 Bernstein 调和函数线性组合而成。

因为：

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

(6) 导函数

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

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

(8) 升阶公式

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

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

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

(9) 积分

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

Bezier 曲线
演示

3. Bezier曲线的性质

(1) 端点性质

a) 曲线端点位置矢量

由Bernstein基函数的端点性质可以推得：当 $t = 0$ 时， $P(0) = P_0$ ；当 $t = 1$ 时， $P(1) = P_n$ 。由此可见：**Bezier曲线的起点、终点与相应的特征多边形的起点、终点重合。**

b) 切矢量

因为 $P'(t) = n \sum_{i=0}^{n-1} P_i [B_{i-1,n-1}(t) - B_{i,n-1}(t)]$ ，所以当 $t = 0$ 时，

$P'(0) = n(P_1 - P_0)$ ；当 $t = 1$ 时， $P'(1) = n(P_n - P_{n-1})$ 。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。

(2) 对称性。由控制顶点 $P_i^* = P_{n-i}, (i = 0, 1, \dots, n)$ 构造出的新Bezier曲线，与原Bezier曲线形状相同，走向相反。因为：

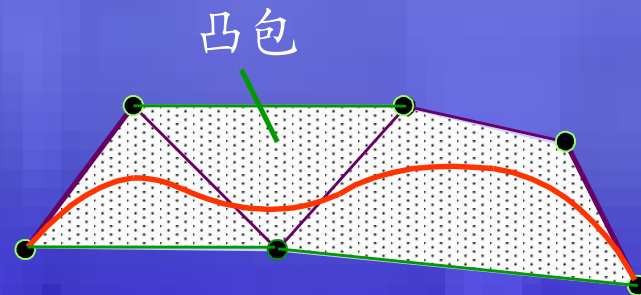
$$\begin{aligned} P^*(t) &= \sum_{i=0}^n P_i^* B_{i,n}(t) = \sum_{i=0}^n P_{n-i} B_{i,n}(t) = \sum_{i=0}^n P_{n-i} B_{n-i,n}(1-t) \\ &= \sum_{i=0}^n P_i B_{i,n}(1-t), \quad t \in [0, 1] \end{aligned}$$

这个性质说明Bezier曲线在起点处有什么几何性质，在终点处也有相同的性质。

(3) 凸包性

由于 $\sum_{i=0}^n B_{i,n}(t) \equiv 1$, 且 $0 \leq B_{i,n}(t) \leq 1$ ($0 \leq t \leq 1, i = 0, 1, \dots, n$), 这

一结果说明: 当 t 在 $[0,1]$ 区间变化时, 对某一个 t 值, $P(t)$ 是特征多边形各顶点的加权平均, 权因子依次是 $B_{i,n}(t)$ 。在几何图形上, 意味着 Bezier 曲线 $P(t)$ 在 $t \in [0,1]$ 中各点是控制点 P_i 的凸线性组合, 即曲线落在 P_i 构成的凸包之中, 如下图所示。



Bezier曲线的凸包性

(4) 几何不变性。 这是指某些几何特性不随坐标变换而变化的特性。Bezier曲线位置与形状与其特征多边形顶点 $P_i (i=0,1,\dots,n)$ 的位置有关，它不依赖坐标系的选择。

(5) 变差缩减性。 若Bezier曲线的特征多边形 $P_0P_1\cdots P_n$ 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。

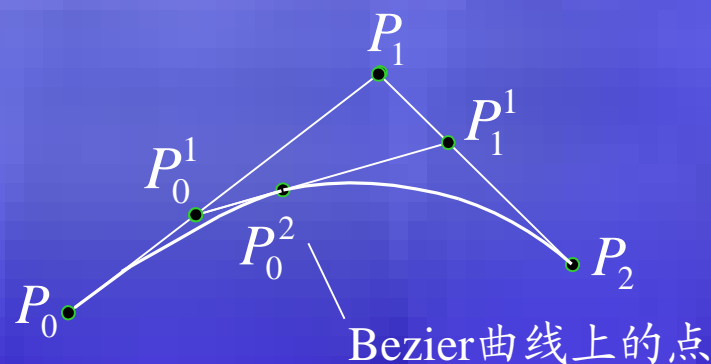
3.1.2.2 Bezier曲线的递推(de Casteljau)算法

计算Bezier曲线上的点，可用Bezier曲线方程，但使用de Casteljau(德卡斯特罗)提出的递推算法则要简单得多。

如下图所示，设 P_0 、 P_0^2 、 P_2 是一条抛物线上顺序三个不同的点。过 P_0 和 P_2 点的两切线交于 P_1 点，在 P_0^2 点的切线交 P_0P_1 和 P_2P_1 于 P_0^1 和 P_1^1 ，则如下比例成立：

$$\frac{P_0P_0^1}{P_0^1P_1} = \frac{P_1P_1^1}{P_1^1P_2} = \frac{P_0^1P_0^2}{P_0^2P_1^1}$$

这是所谓抛物线的三切线定理。



抛物线三切线定理

当 P_0 、 P_2 固定，引入参数 t ，令上述比值为 $t:(1-t)$ ，即有：

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

$$P_1^1 = (1-t)P_1 + tP_2$$

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

t 从0变到1，第一、二式就分别表示两条一次Bezier曲线，其特征多边形分别是 P_0P_1 和 P_1P_2 。将一、二式代入第三式得：

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

当 t 从0变到1时，它表示了特征多边形为 $P_0P_1P_2$ 定义的一条二次Bezier曲线。并且表明：这二次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)确定的一次Bezier曲线的线性组合。依次类推，由四个控制点定义的三次Bezier曲线 P_0^3 可被定义为分别由(P_0, P_1, P_2)和(P_1, P_2, P_3)确定的二条二次Bezier曲线的线性组合；由 $(n+1)$ 个控制点 $P_i(i = 0, 1, \dots, n)$ 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_1^{n-1} 与 P_0^{n-1} 的线性组合：

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

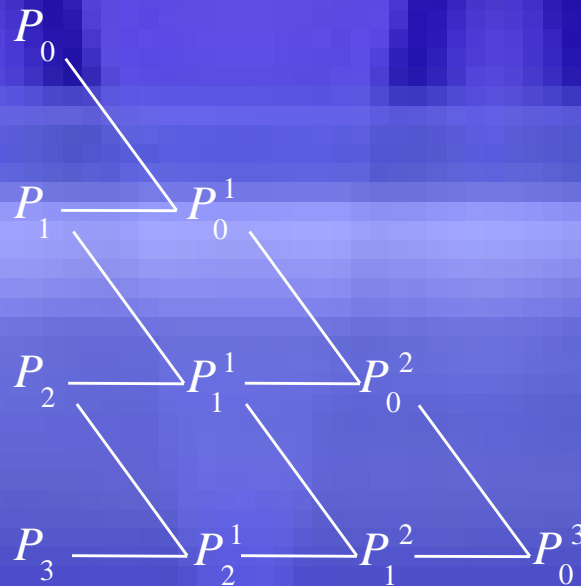
由此得到Bezier曲线的递推计算公式:

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

这便是著名的de Casteljau(德卡斯特罗)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。上式中 $P_i^0 = P_i$ 是定义Bezier曲线的控制点, P_0^n 即为曲线 $P(t)$ 上具有参数 t 的点。

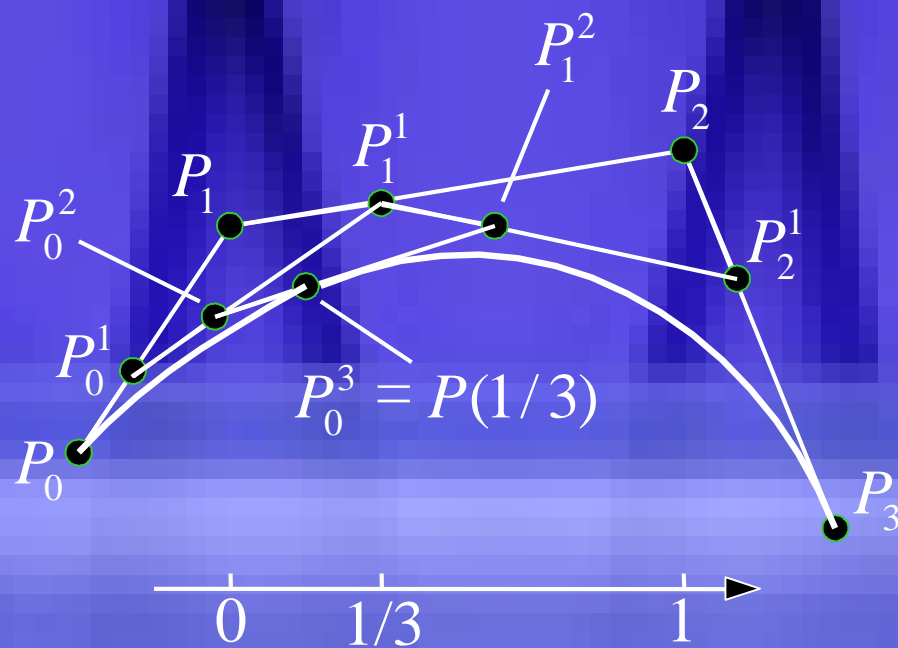
de Casteljau(德卡斯特罗)算法稳定可靠，直观简便，可以编出十分简捷的程序，是计算Bezier曲线的基本算法和标准算法。

当 $n = 3$ 时，de casteljau算法递推出的 P_i^k 呈直角三角形，对应结果如下图所示。从左向右递推，最右边点 P_0^3 即为曲线上的点。



$n = 3$ 时 P_i^n 的递推关系

这一算法可用简单的几何作图来实现。给定参数 $t \in [0,1]$ ，就把定义域分成长度为 $t:(1-t)$ 的两段。依次对原始控制多边形每一边执行同样的定比分割，所得分点就是由第一级递推生成的中间顶点 $P_i^1 (i=0,1,\dots,n-1)$ ，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 $P_i^2 (i=0,1,\dots,n-2)$ 。重复进行下去，直到 n 级递推得到一个中间顶点 P_0^n 即为所求曲线上的点 $P(t)$ ，如下图所示。



几何作图法求Bezier曲线
上一点 ($n = 3, t = 1/3$)