

第三章 几何造型技术

- 几何造型技术
 - 参数曲线和曲面
 - Bezier 曲线与曲面
 - B样条曲线与曲面
 - NURBS曲线和曲面
 - Coons曲面
 - 形体在计算机内的表示
 - 求交分类
 - 实体造型系统简介
 - 三角网格

3.2 Bezier 曲线与曲面

- 起源：
 - 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，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UNISURF系统(1972年投入了应用)；
 - 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。

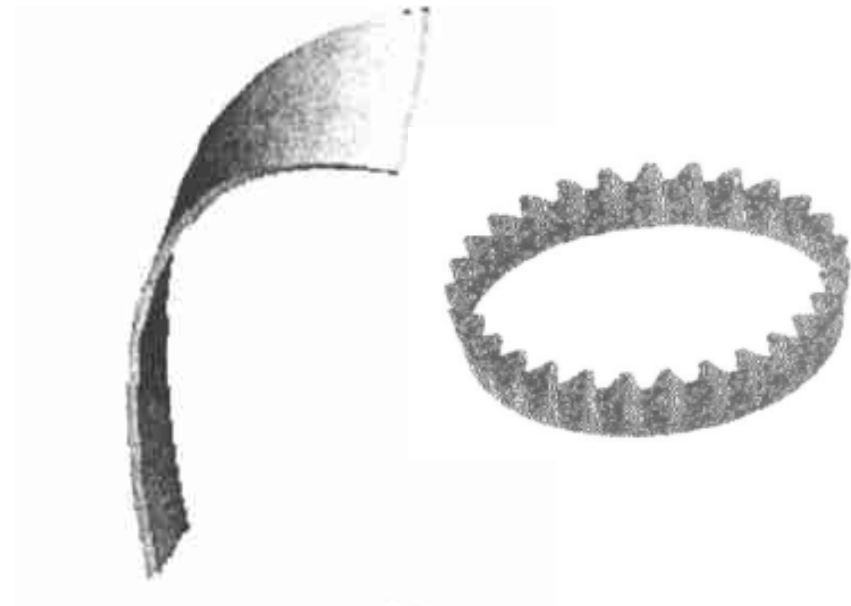
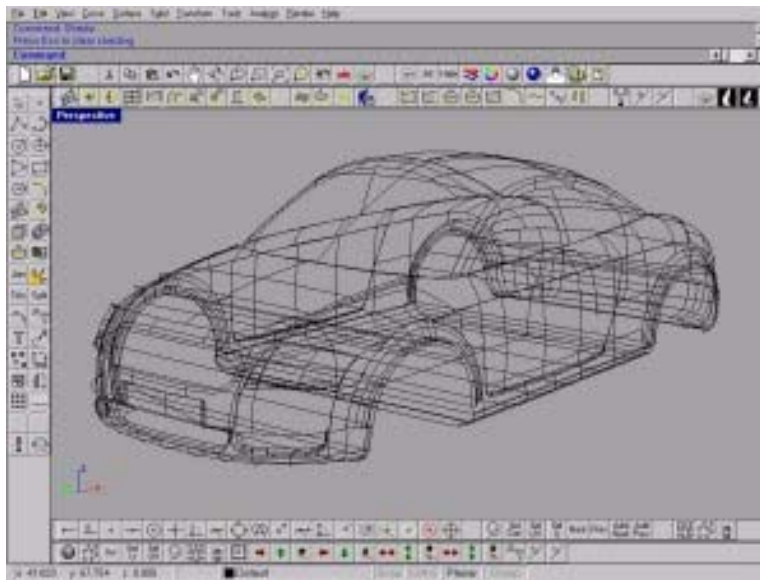
3.2 Bezier 曲线与曲面

- 优点：
 - 输入的控制点与生成曲线之间的关系明确；
 - 能方便地改变曲线的形状和阶次；
 - 无论是直线或曲线都能在数学上予以描述 (为计算机矢量图形学奠定了基础)。



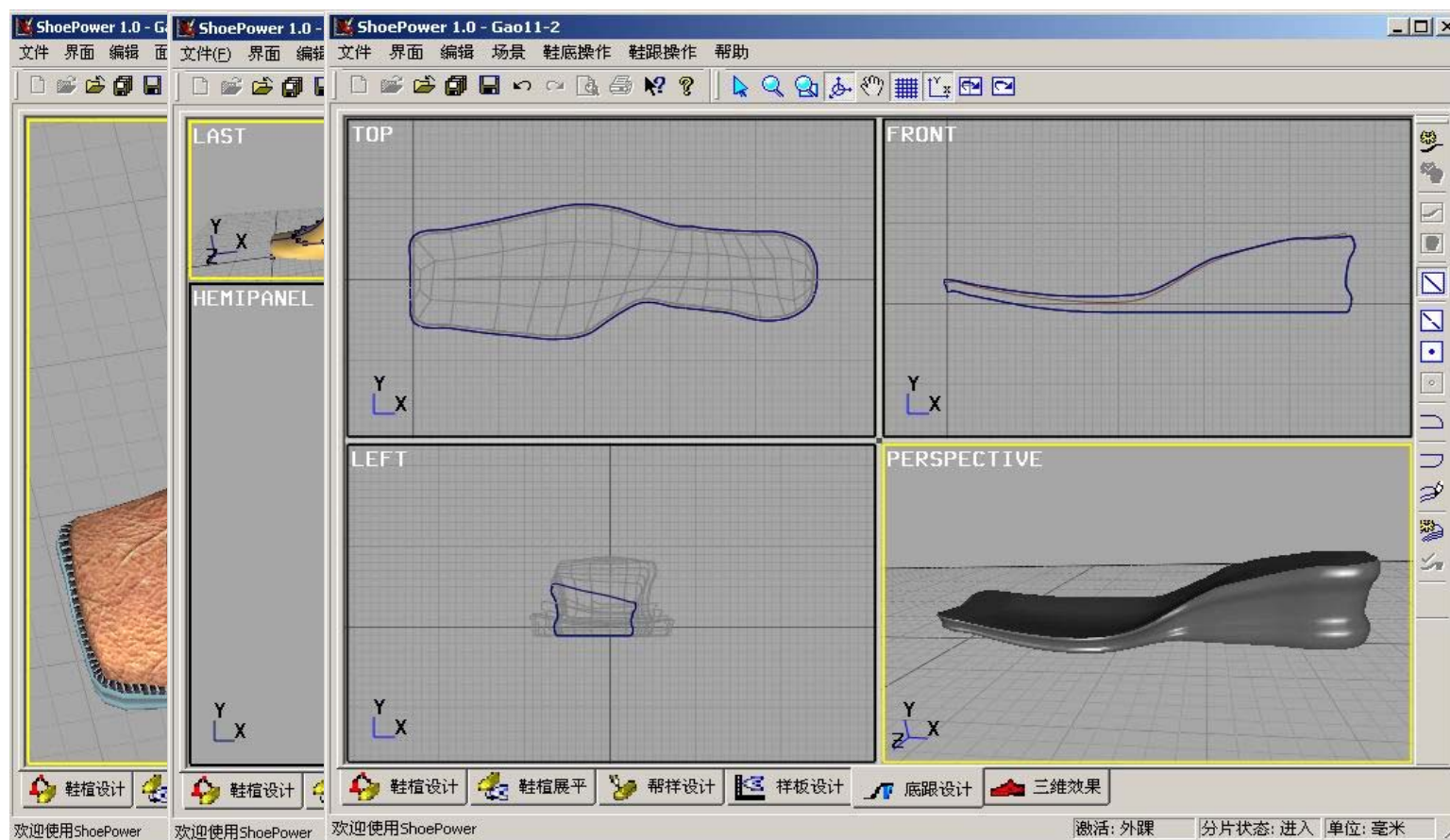
3.2 Bezier 曲线与曲面

- 应用领域
 - 计算机辅助设计与制造（CAD/CAM）
 - 飞机、汽车、船舶外形的设计；
 - 水泵叶轮和齿轮等机械零件的设计。



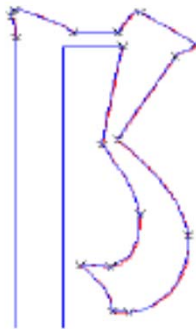
3.2 Bezier 曲线与曲面

— 桥梁建筑物以及日用品的设计



3.2 Bezier 曲线与曲面

- 曲线字形轮廓描述
- 地图图形管理系统



3.2 Bezier 曲线与曲面

- 移动机器人运动规划
 - 处于A点，需要达到D点；
 - AD之间有障碍物；
 - 运动往往是沿着曲线进行的。

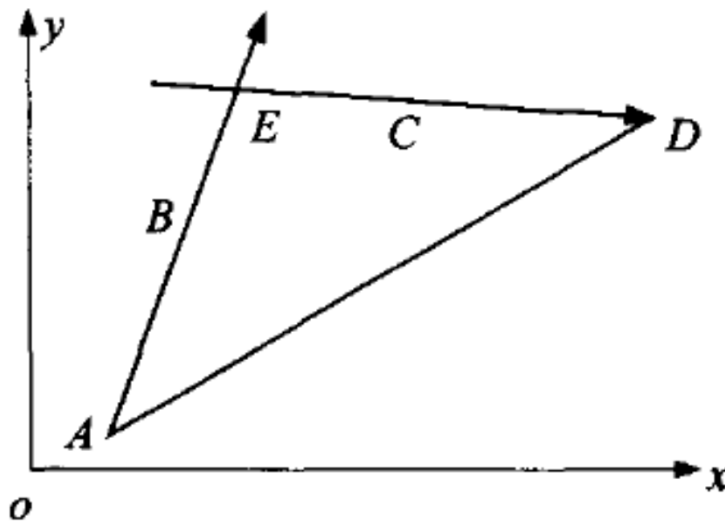


图 1 机器人运动示意图



3.2.1 Bezier 曲线与曲面

- Bezier 曲线
 - 定义
 - 基函数性质、曲线性质
 - 递推算法
 - 拼接
 - 升阶与降阶
- Bezier 曲面
 - 定义、性质、递推算法、拼接
- 三边Bezier曲面片

3.2.1 Bezier 曲线的定义与性质

- 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：

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

- 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：

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

- 其中 $0^0=1, 0!=1$

3.2.1 Bezier 曲线的定义与性质

- 曲线实例如右图所示

习题

- 设有控制顶点为 $P_0(0, 0)$, $P_1(48, 96)$, $P_2(120, 120)$, $P_3(216, 72)$ 的三次Bézier曲线 $P(t)$, 试计算 $P(0.4)$ 的 (x, y) 坐标, 并写出 $(x(t), y(t))$ 的多项式表示。
- 设一条三次Bézier曲线的控制顶点为 P_0, P_1, P_2, P_3 。对曲线上一点 $P(0.5)$, 及一个给定的目标点 T , 给出一种调整Bézier曲线形状的方法, 使得 $P(0.5)$ 精确通过点 T 。

3.2.1 Bezier 曲线的定义与性质

- Bernstein基函数的性质

- 正性

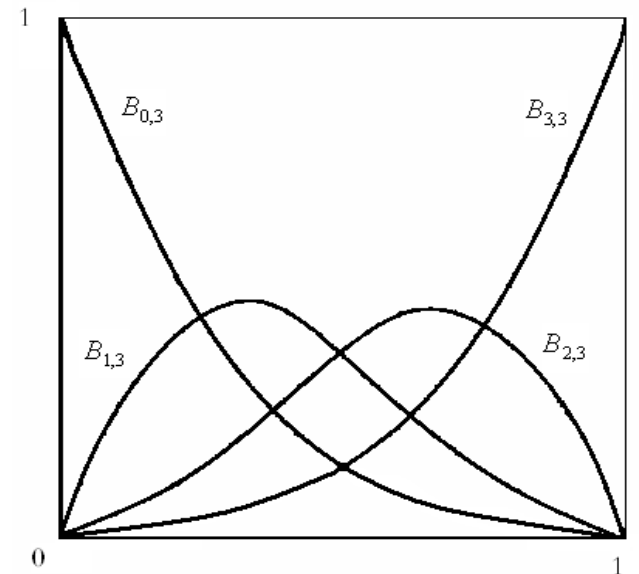
$$B_{i,n}(t) = \begin{cases} = 0 / 1 & t = 0, 1 \\ > 0 & t \in (0, 1), i = 1, 2, \dots, n-1; \end{cases}$$

- 端点性质

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

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

$$0! = 1, \quad 0^0 = 1。$$



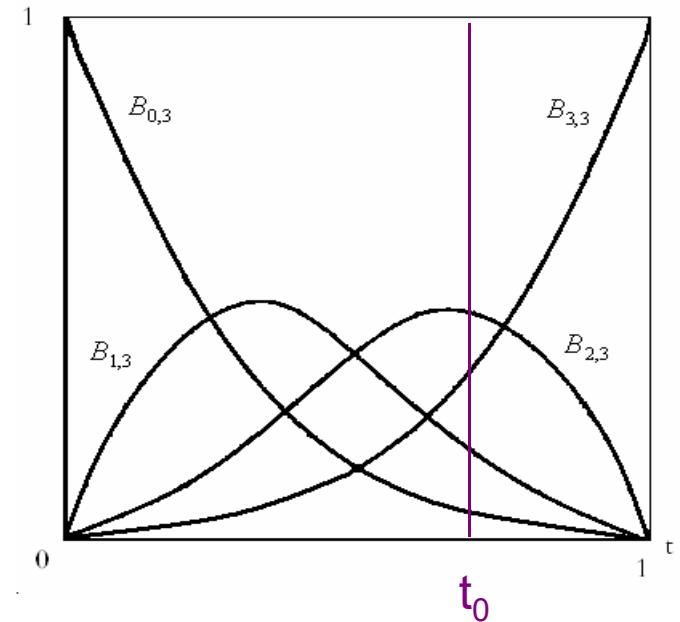
3.2.1 Bezier 曲线的定义与性质

- Bernstein基函数的性质

- 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：

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

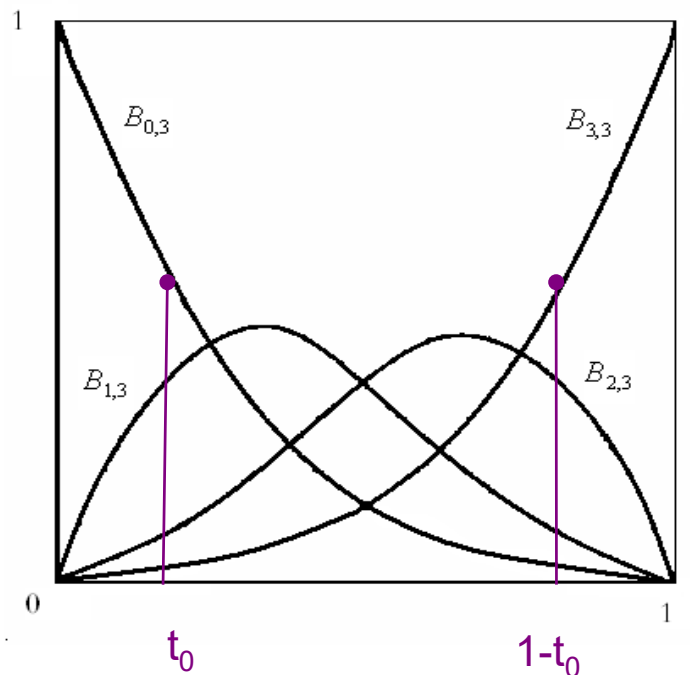


3.2.1 Bezier 曲线的定义与性质

- Bernstein基函数的性质

- 对称性 $B_{i,n}(t) = B_{n-i,n}(1-t)$

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



3.2.1 Bezier 曲线的定义与性质

- Bernstein基函数的性质

- 递推性

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

因为：

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

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

3.2.1 Bezier 曲线的定义与性质

- Bernstein基函数的性质

- 导函数

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

- 证明

$$\begin{aligned} B'_{i,n}(t) &= \frac{n!}{i!(n-i)!} i \cdot t^{i-1} (1-t)^{n-i} - \frac{n!}{i!(n-i)!} (n-i)(1-t)^{n-i-1} \cdot t^i \\ &= \frac{n(n-1)!}{(i-1)!((n-1)-(i-1))!} \cdot t^{i-1} \cdot (1-t)^{(n-1)-(i-1)} - \frac{n(n-1)!}{i!((n-1)-i)!} \cdot t^i \cdot (1-t)^{(n-1)-i} \\ &= n(B_{i-1,n-1}(t) - B_{i,n-1}(t)) \end{aligned}$$

3.2.1 Bezier 曲线的定义与性质

- Bernstein基函数的性质

- 最大值

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

- 积分

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

$$(1+t)^n = C_n^0 1^n t^0 + C_n^1 1^{n-1} t^1 + \dots + C_n^n 1^0 t^n$$

3.2.1 Bezier 曲线的定义与性质

- Bernstein基函数的性质
 - 升阶公式

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

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

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

3.2.1 Bezier 曲线的定义与性质

- Bezier曲线的性质

- 端点性质

- 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。

- 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， $P'(1)=n(P_n-P_{n-1})$ ，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。

3.2.1 Bezier 曲线的定义与性质

- 端点性质
 - 二阶导矢

$$P''(t) = n(n-1) \sum_{i=0}^{n-2} (P_{i+2} - 2P_{i+1} + P_i) B_{i,n-2}(t)$$

- 当 $t=0$ 时, $P''(0) = n(n-1)(P_2 - 2P_1 + P_0)$

- 当 $t=1$ 时, $P''(1) = n(n-1)(P_n - 2P_{n-1} + P_{n-2})$

上式表明：2阶导矢只与相邻的3个顶点有关，事实上， r 阶导矢只与 $(r+1)$ 个相邻点有关，与更远点无关。

3.2.1 Bezier 曲线的定义与性质

– 端点性质

- 二阶导矢

– 将 $P'(0)$ 、 $P''(0)$ 及 $P'(1)$ 、 $P''(1)$ 代入曲率式 $k(t) = \frac{|P'(t) \times P''(t)|}{|P'(t)|^3}$

可以得到Bezier曲线在端点的曲率分别为：

$$k(0) = \frac{n-1}{n} \cdot \frac{|(P_1 - P_0) \times (P_2 - P_1)|}{|P_1 - P_0|^3}$$

$$k(1) = \frac{n-1}{n} \cdot \frac{|(P_{n-1} - P_{n-2}) \times (P_n - P_{n-1})|}{|P_n - P_{n-1}|^3}$$

3.2.1 Bezier 曲线的定义与性质

- Bezier曲线的性质
 - k 阶导函数的差分表示

n 次Bezier曲线的 k 阶导数可用差分公式为:

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

- 其中高阶向前差分矢量由低阶向前差分矢量递推地定义:

$$\Delta^k P_i = \Delta^{k-1} P_{i+1} - \Delta^{k-1} P_i$$

- 例如

$$\Delta^0 P_i = P_i$$

$$\Delta^1 P_i = \Delta^0 P_{i+1} - \Delta^0 P_i = P_{i+1} - P_i$$

$$\Delta^2 P_i = \Delta^1 P_{i+1} - \Delta^1 P_i = P_{i+2} - 2P_{i+1} + P_i$$

习题

- 试证明 n 次Bezier曲线退化为 $(n-1)$ 次Bezier曲线的条件为 $\Delta^n P_0 = 0$ 。

$$\begin{aligned}\Delta^k P_i &= \Delta(\Delta^{k-1} P_i) \\ &= \Delta^{k-1} P_{i+1} - \Delta^{k-1} P_i \\ &= \sum_{j=0}^k (-1)^j C_k^j P_{i+k-j}\end{aligned}$$

习题

- 试证明 n 次Bezier曲线退化为 $(n-1)$ 次Bezier曲线的条件为 $\Delta^n P_0 = 0$ 。

$$P(t) = P_0 C_n^0 t^0 (1-t)^n + P_1 C_n^1 t^1 (1-t)^{n-1} + \dots + P_n C_n^n t^n (1-t)^0$$

$$\sum_{i=0}^n (-1)^i C_n^{n-i} P_{n-i} = \sum_{i=0}^n (-1)^i C_n^i P_{n-i}$$

$$\begin{aligned} \Delta^k P_i &= \Delta(\Delta^{k-1} P_i) \\ &= \Delta^{k-1} P_{i+1} - \Delta^{k-1} P_i \\ &= \sum_{j=0}^k (-1)^j C_k^j P_{i+k-j} \end{aligned}$$

3.2.1 Bezier 曲线的定义与性质

- Bezier曲线的性质

- 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:

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

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。

3.2.1 Bezier 曲线的定义与性质

- Bezier曲线的性质

- 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。

3.2.1 Bezier 曲线的定义与性质

- Bezier曲线的性质

- 几何不变性:

- 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 $P_i(i=0,1,\cdots,n)$ 的位置有关:

$$\sum_{i=0}^n P_i B_{i,n}(t) = \sum_{i=0}^n P_i B_{i,n}\left(\frac{u-a}{b-a}\right)$$

- 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;
 - 反映了Bezier曲线比其特征多边形的波动还小。

3.2.1 Bezier 曲线的定义与性质

- Bezier曲线的性质

- 仿射不变性

对于任意的仿射变换A:

$$A([P(t)]) = A\left\{\sum_{i=0}^n P_i B_{i,n}(t)\right\} = \sum A[P_i] B_{i,n}(t)$$

- 即在仿射变换下, $P(t)$ 的形式不变。

3.2.1 Bezier 曲线的定义与性质

- Bezier 曲线的矩阵表示形式

- 一次

$$P(t) = [t, 1] \begin{bmatrix} -1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \end{bmatrix}$$

- 二次

$$P(t) = [t^2, t, 1] \begin{bmatrix} 1 & -2 & 1 \\ -2 & 2 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \end{bmatrix}$$

- 三次

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

3.1.2.2 Bezier曲线的递推算法

- de Casteljau(德卡斯特里奥)递推算法

便于计算Bezier曲线上的点

- 如图所示, 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:

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

图3.1.11 n=3时 P_i^n 的递推关系

- 抛物线三切线定理

3.1.2.2 Bezier曲线的递推算法

- de Casteljau递推算法
 - 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(1)、(2)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。

3.1.2.2 Bezier曲线的递推算法

- de Casteljau递推算法
 - 将(1)、(2)式代入第(3)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，它表示了由三顶点 P_0 、 P_1 、 P_2 三点定义的一条二次Bezier曲线；
 - 这二次Bezier曲线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。
- 依次类推，由四个控制点定义的三次Bezier曲线可被定义为分别由 (P_0, P_1, P_2) 和 (P_1, P_2, P_3) 确定的两条二次Bezier曲线的线性组合。

3.1.2.2 Bezier曲线的递推算法

- de Casteljau递推算法

- 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_0^{n-1} 与 P_1^{n-1} 的线性组合:

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

- 由此得到Bezier曲线的递推计算公式:

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

- 著名的de Casteljau算法。

3.1.2.2 Bezier曲线的递推算法

- de Casteljau递推算法

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

- 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。
- 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。

3.1.2.2 Bezier曲线的递推算法

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

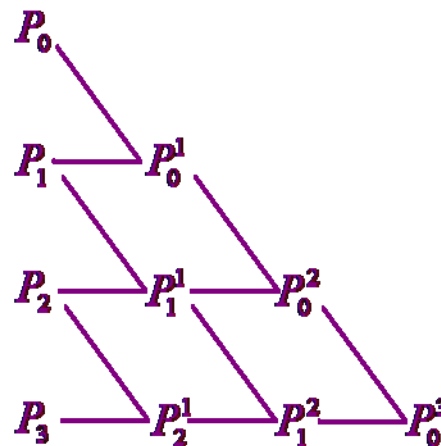


图3. 1. 11 $n=3$ 时 P_i^k 的递推关系

3.1.2.2 Bezier曲线的递推算法

- 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 $P_i^1 (i=0,1,\dots,n-1)$ ；
 - 进行 n 级递推得到中间顶点 P_0^n 即为所求曲线上的点。

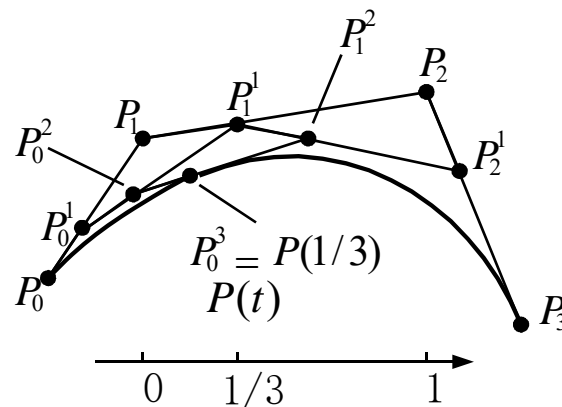


图3. 1. 12 几何作图法求Bezier曲线上一点 ($n=3, t=1/3$)

习题

- 已知三次Bezier曲线上的4个点分别为 $Q_0(50, 0)$, $Q_1(100, 0)$, $Q_2(0, 50)$, $Q_3(0, 100)$, 它们对应的参数分别为0, $1/3$, $2/3$, 1, 求Bezier曲线的控制顶点。
- 计算以 $(30,0), (60,10), (80,30), (90,60), (90,90)$ 为控制顶点的4次Bezier曲线在 $t=1/2$ 处的值,并画出de Casteljau三角形。
- 给出三次Bezier曲线退化为二次Bezier, 控制顶点 P_0, P_1, P_2, P_3 应该满足的条件。
- 设一条二次Bezier曲线的控制顶点为 P_0, P_1, P_2 , 另一条二次Bezier曲线的控制顶点为 Q_0, Q_1, Q_2 , $P_2 = Q_0$. 写出两条曲线可以精确合并(表示)为一条二次Bezier曲线的条件。

3.1.2.3 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。
 - 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；
 - 可采用分段设计，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。
- 两段Bezier曲线达到不同阶几何连续的条件

3.1.2.3 Bezier曲线的拼接

- 给定两条Bezier曲线 $P(t)$ 和 $Q(t)$ ，相应控制点为 $P_i (i=0, 1, \dots, n)$ 和 $Q_j (j=0, 1, \dots, m)$ ，且令 $a_i = P_i - P_{i-1}, b_j = Q_j - Q_{j-1}$ ，如图3.1.13所示，我们现在把两条曲线连接起来。

图3.1.13 Bezier曲线的拼接

3.1.2.3 Bezier曲线的拼接

- 达到 G^0 连续的充要条件是: $P_n = Q_0$;
- 达到 G^1 连续的充要条件是: $P_{n-1}, P_n = Q_0, Q_1$ 三点共线, 即:

$$b_1 = \alpha a_n (\alpha > 0)$$

- 达到 G^2 连续的充要条件是: 在 G^1 连续的条件下, 并满足方程

$$Q''(0) = \alpha^2 P''(1) + \beta P'(1)$$

- 将 $Q''(0)$ 、 $P''(1)$ 和 $P'(1)$, $Q_0 = P_n$ 、 $Q_1 - Q_2 = \alpha(P_n - P_{n-1})$ 代入, 并整理, 可以得到:

$$Q_2 = \left(\alpha^2 + 2\alpha + \frac{\beta}{n-1} + 1 \right) P_n - \left(2\alpha^2 + 2\alpha + \frac{\beta}{n-1} \right) P_{n-1} + \alpha^2 P_{n-2}$$

3.1.2.3 Bezier曲线的拼接

$$Q_2 = \left(\alpha^2 + 2\alpha + \frac{\beta}{n-1} + 1 \right) P_n - \left(2\alpha^2 + 2\alpha + \frac{\beta}{n-1} \right) P_{n-1} + \alpha^2 P_{n-2}$$

- 选择 α 和 β 的值, 可以利用该式确定曲线段 $Q(t)$ 的特征多边形顶点 Q_2 ;
- 顶点 Q_0 、 Q_1 已被 G^1 连续条件所确定, 要达到 G^2 连续的话, 只剩下顶点 Q_2 可以自由选择;
- 如果上式的两边都减去 P_n , 则等式右边可以表示为 $(P_n - P_{n-1})$ 和 $(P_{n-1} - P_{n-2})$ 的线性组合:

$$Q_2 - P_n = \left(\alpha^2 + 2\alpha + \frac{\beta}{n-1} \right) (P_n - P_{n-1}) - \alpha^2 (P_{n-1} - P_{n-2})$$

- 这表明 P_{n-2} 、 P_{n-1} 、 $P_n = Q_0$ 、 Q_1 和 Q_2 五点共面。

3.1.2.4 Bezier曲线的升阶与降阶

- Bezier曲线的升阶
 - 升阶：保持Bezier曲线的形状与方向不变，增加定义它的控制顶点数，提高该Bezier曲线的次数。
 - 增加了控制顶点数，增加了对曲线进行形状控制的灵活性，还在构造曲面方面有着重要的应用：
 - 对于由曲线生成曲面的算法，要求那些曲线必须是同次的。
 - 应用升阶的方法，可以把低于最高次数的曲线提升到最高次数，而获得相同的次数。

3.1.2.4 Bezier曲线的升阶与降阶

- 新控制顶点的计算

- 设给定原始控制顶点 P_0, P_1, \dots, P_n ，定义了一条 n 次Bezier曲线：

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

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

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

- 对上式左边乘以 $(t + (1-t))$ ，得到：

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

3.1.2.4 Bezier曲线的升阶与降阶

- 新控制顶点的计算

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

- 设比较等式两边 $t^i (1-t)^{n+1-i}$ 项的系数, 得到:

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

- 化简即得:

$$P_i^* = \frac{i}{n+1} P_{i-1} + \left(1 - \frac{i}{n+1}\right) P_i \quad (i = 0, 1, \dots, n+1)$$

- 其中 $P_{-1} = P_{n+1} = (0, 0)$

3.1.2.4 Bezier曲线的升阶与降阶

– 上述结果说明：

- 新的控制顶点 P_i^* 是以参数值 $\frac{i}{n+1}$ 按分段线性插值从原始特征多边形得出的
- 升阶后的新的特征多边形在原始特征多边形的凸包内
- 特征多边形更靠近曲线。

三次Bezier曲线的升阶实例如图3.1.14所示。

习题

- 推导Bezier曲线的升阶公式；给定三次Bezier曲线的控制顶点(0,0), (0,100), (100,0), (100,100), 计算升阶一次后的控制顶点。

$$P_i^* = \frac{i}{n+1} P_{i-1} + \left(1 - \frac{i}{n+1}\right) P_i \quad (i = 0, 1, \dots, n+1)$$

3.1.2.4 Bezier曲线的升阶与降阶

- Bezier曲线的降阶
 - 给定一条由原始控制顶点定义的 n 次Bezier曲线，要求找到一条由新控制顶点定义的 $n-1$ 次Bezier曲线来逼近原始曲线。
- 新控制点的计算
 - 假定 P_i 是由 P_i^* 升阶得到，则由升阶公式有：

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

3.1.2.4 Bezier曲线的升阶与降阶

- 新控制点的计算

- 从前述方程可以导出两个递推公式：

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

和

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

- 其中第一个递推公式在靠近 P_0 处趋向生成较好的逼近，而第二个递推公式在靠近 P_n 处趋向生成较好的逼近。

3.5.1 基本概念

- 假设参数曲面片方程为 $P(u,v)$, $u,v \in [0,1]$, $P(1,v)$ 称为曲面片的四条边界, $P(0,0), P(0,1), P(1,0), P(1,1)$ 称为曲面片的四个角点。 $P(u,v)$ 的 u 向和 v 向求偏导矢:

$$P_u(u,v) = \frac{\partial P(u,v)}{\partial u} \quad P_v(u,v) = \frac{\partial P(u,v)}{\partial v}$$

分别称为 u 向和 v 向上的切矢。

- 边界线 $P(u, 0)$ 上的切矢为:

$$P_u(u, 0) = \left. \frac{\partial P(u,v)}{\partial u} \right|_{v=0}$$

- 同理: $P_u(u, 1), P_v(0, v), P_v(1, v)$ 也是边界线上的切矢。

3.5.1 基本概念

- 边界曲线 $P(u,0)$ 上的法向(指参数 v 向)偏导矢:

$$P_v(u,0) = \frac{\partial P(u,v)}{\partial v} \Big|_{v=0}$$

称为边界曲线的跨界切矢。同理, $P_v(u,1), P_u(0,v), P_u(1,v)$ 也是边界曲线的跨界切矢。

- 角点 $P(0,0)$ 的 u 向和 v 向切矢:

$$P_v(0,0) = \frac{\partial P(u,v)}{\partial v} \Big|_{\substack{u=0 \\ v=0}}$$

$$P_u(0,0) = \frac{\partial P(u,v)}{\partial u} \Big|_{\substack{u=0 \\ v=0}}$$

在曲线片的每个角点上都有两个这样的切矢量。

3.5.1 基本概念

- 混合偏导矢或扭矢:

$$P_{uv}(u, v) = \frac{\partial P(u, v)}{\partial u \partial v}$$

它反映了 P_u 对 v 的变化率或 P_v 对 u 的变化率。

- 角点 $P(0,0)$ 的扭矢:

$$P_{uv}(0,0) = \frac{\partial P(u, v)}{\partial u \partial v} \bigg|_{\substack{u=0 \\ v=0}}$$

显然，曲面片上的每个角点都有这样的扭矢。

3.1.2.5 Bezier曲面

- 定义

- 设 $P_{ij} (i = 0, 1, \dots, m; j = 0, 1, \dots, n)$ 为 $(m+1) \times (n+1)$ 个空间点列, 则 $m \times n$ 次张量积形式的Bezier曲面定义为:

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

- 其中 $B_{i,m}(u) = C_m^i u^i (1-u)^{m-i}$, $B_{j,n}(v) = C_n^j v^j (1-v)^{n-j}$ 是Bernstein基函数。
- 依次用线段连接点列 $P_{ij} (i = 0, 1, \dots, m; j = 0, 1, \dots, n)$ 中相邻两点所形成的空间网格, 称之为特征网格。

3.1.2.5 Bezier曲面

- 定义

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

- Bezier曲面的矩阵表示式是:

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

- 在一般实际应用中 m, n 不大于4。

3.1.2.5 Bezier曲面

- 性质

- 几何不变性。
- 对称性。
- 凸包性。

除变差缩减性质外，Bezier曲线的其它性质可推广到Bezier曲面：

- 曲面特征网格的四个角点正好是Bezier曲面的四个角点，即 $P(0,0) = P_{00}$, $P(1,0) = P_{m0}$, $P(0,1) = P_{0n}$, $P(1,1) = P_{mn}$ ；

3.1.2.5 Bezier曲面

- 曲面特征网格最外一圈顶点定义曲面的四条边界;
- **Bezier**曲面边界的跨界切矢只与定义该边界的顶点及相邻一排顶点有关, 且 $P_{00}P_{01}P_{10}$ 、 $P_{0n}P_{1n}P_{0,n-1}$ 、 $P_{m0}P_{m-1,0}P_{m1}$ 和 $P_{mn}P_{m,n-1}P_{m-1,n}$ 所在平面分别在对应角点处与平面相切; 其跨界二阶导矢只与定义该边界的及相邻两排顶点有关。

3.1.2.5 Bezier曲面

- 曲面的拼接
 - 设两张 $m \times n$ 次Bezier曲面片

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

$$u, v \in [0, 1]$$

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

分别由控制顶点 P_{ij} 和 Q_{ij} 定义。

3.1.2.5 Bezier曲面

- 曲面的拼接
 - 如要两曲面片达到 G^0 连续，则它们有公共边界，即：

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

- 如果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，即曲面的法向应当是跨界连续的：

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

- 满足条件的两种方法：
 - 鉴于 G^0 的条件，上式最简单的解：

$$Q_u(0, v) = \alpha(v) P_u(1, v)$$

3.1.2.5 Bezier曲面

– 满足条件 $Q_u(0,v) \times Q_v(0,v) = \alpha(v) P_u(1,v) \times P_v(1,v)$ 两种方法:

- 条件: $Q_u(0,v) = \alpha(v) P_u(1,v)$

要求合成曲面上 v 为常数的所有曲线, 在跨界时有切向的连续性。为了保证等式两边关于 v 的多项式次数相同, 必须取 $\alpha(v) = \alpha$ 于是有:

$$\overrightarrow{Q_{1i}Q_{0i}} = \alpha \overrightarrow{P_{ni}P_{n-1,i}} \quad (\alpha > 0, i = 0, 1, \dots, m)$$

即:
$$Q_{1i} - Q_{0i} = \alpha(P_{ni} - P_{n-1,i}) \quad (\alpha > 0, i = 0, 1, \dots, m)$$

只涉及曲面 $P(u,v)$ 和 $Q(u,v)$ 的两列控制顶点, 比较容易控制。这种方法匹配合成的曲面的边界, u 向和 v 向是光滑连续的。

该式的限制是苛刻

3.1.2.5 Bezier曲面

- 为了构造合成曲面时有更大的灵活性。Bezier在1972年放弃把上式作为 G^1 连续的条件，而以

$$Q_u(0,v) = \alpha(v)P_u(1,v) + \beta(v)P_v(1,v)$$

代替，要求 $Q_u(0,v)$ 位于 $P_v(1,v)$ 和 $P_u(1,v)$ 所在的同一个平面内，即曲面片 $P(u,v)$ 边界上相应点处的切平面。

- 增加余地，跨界切矢在跨越曲面片的边界时不再连续
- 为保证等式两边关于 v 的多项式次数相同， α 须为任意正常数， $\beta(v)$ 是 v 的任意线性函数。

3.1.2.5 Bezier曲面

- 递推算法
 - Bezier曲线的递推算法可推广至Bezier曲面的情形。
 - 若给定Bezier曲面特征网格的控制顶点和一对参数值 (u, v) ，则： $P_{ij} (i = 0, \dots, m; j = 0, \dots, n)$

$$P(u, v) = \sum_{i=0}^{m-k} \sum_{j=0}^{n-l} P_{ij}^{k,l} B_{i,m}(u) B_{j,n}(v) = \dots = P_{00}^{m,n} \quad u, v \in [0, 1] \quad (3.1.14)$$

$$P_{ij}^{k,l} = \begin{cases} P_{i,j} & (k = l = 0) \\ (1-v)P_{i,j}^{0,l-1} + vP_{i,j+1}^{0,l-1} & (k = 0; l = 1, 2, \dots, n) \\ (1-u)P_{i,0}^{k-1,n} + uP_{i+1,0}^{k-1,n} & (k = 1, 2, \dots, m; l = n) \end{cases} \quad (3.1.15)$$

3.1.2.5 Bezier曲面

- 递推算法

- 或

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

- 上两式中的下标 i, j 的变化范围已在前式中给出。

- 上面给出了确定曲面上一点的两种方案

- 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

- $P_{0j}^{m0} (j=0,1,\dots,n)$ 构成的中间多边形。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} , 即所求曲线上的点 $P(u,v)$ 。