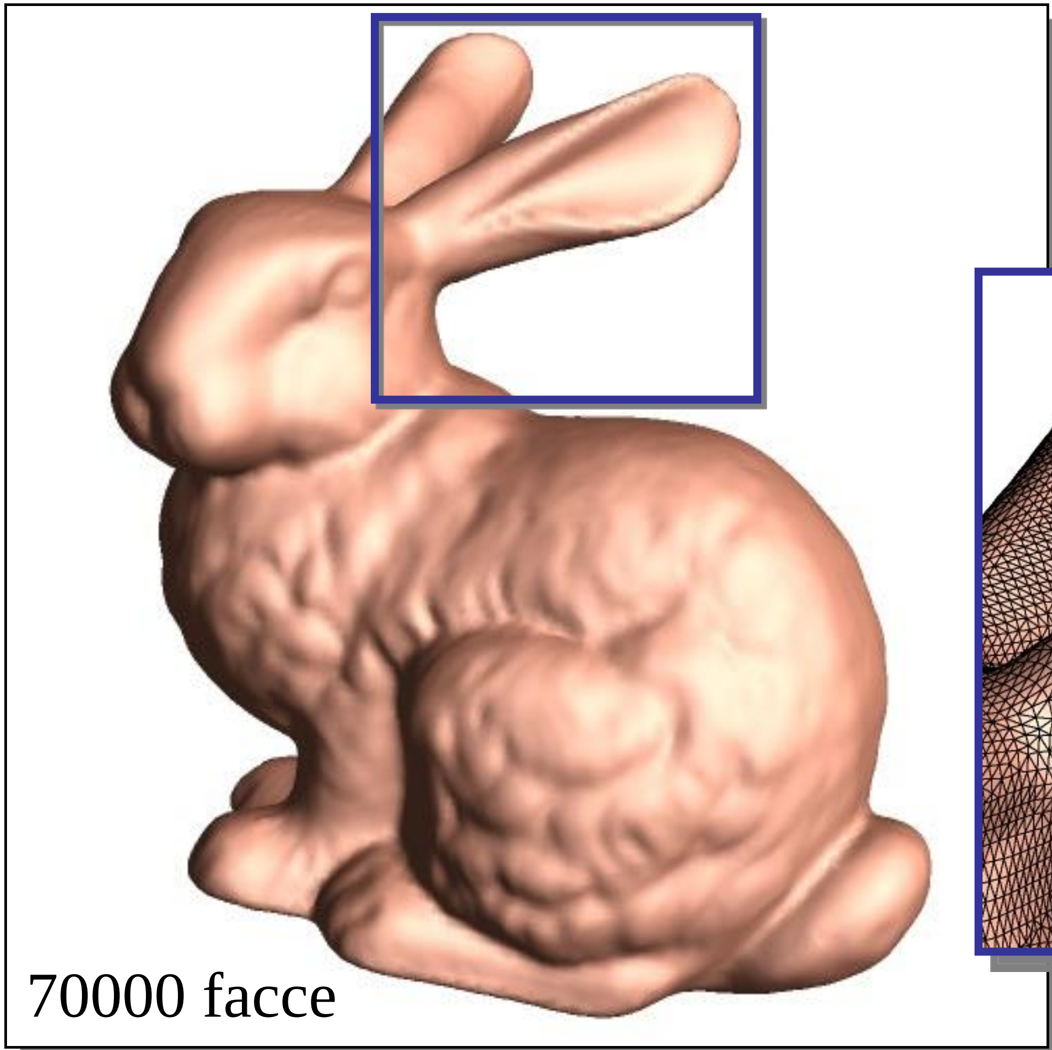
Geometric Modeling: surface

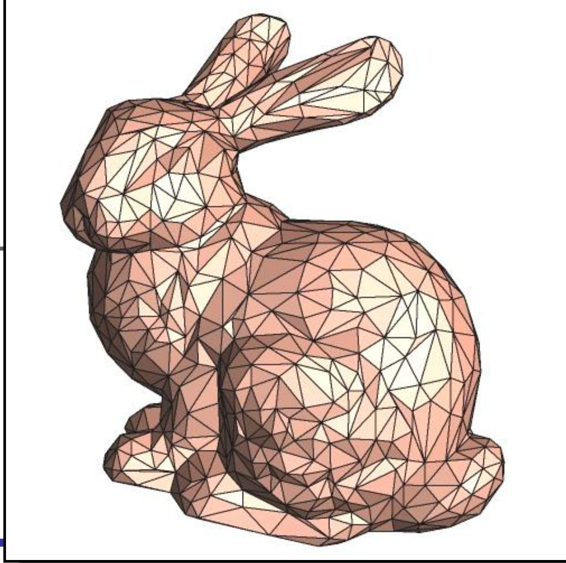
- **Patch:**
  - patch di Bézier
  - NURBS surfaces

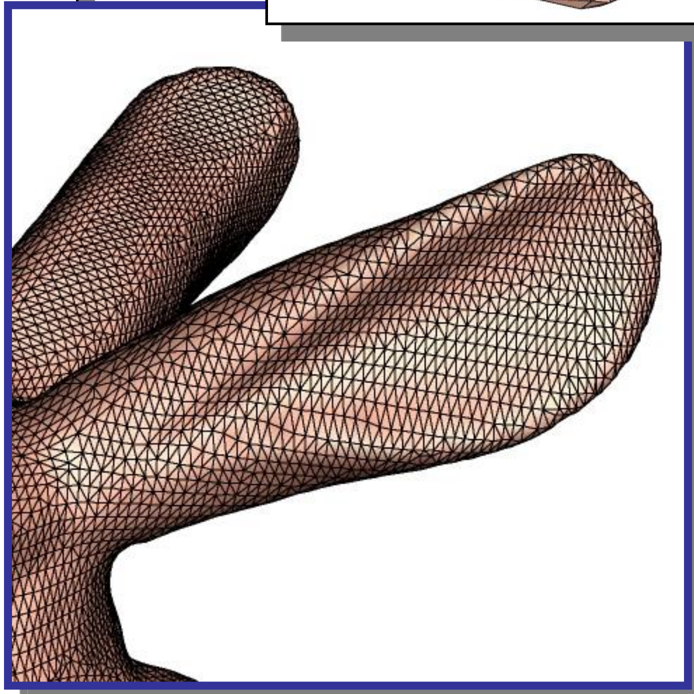
Polygonal Mesh:

piecewise planar surfaces

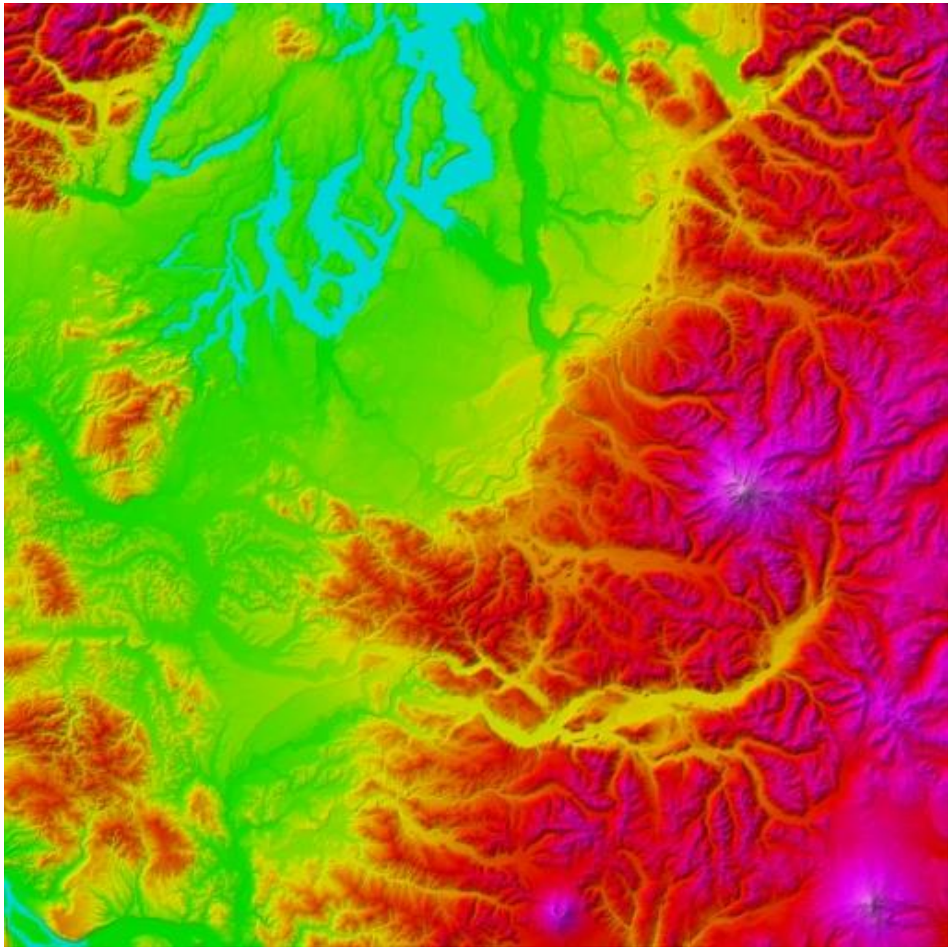


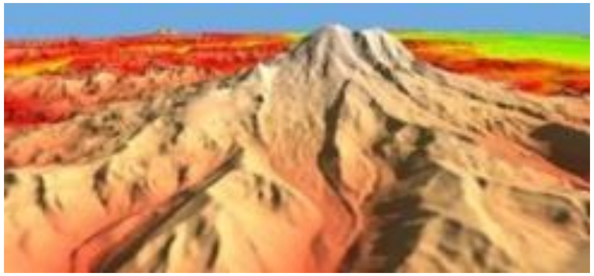
70000 facce





Polygonal Mesh:



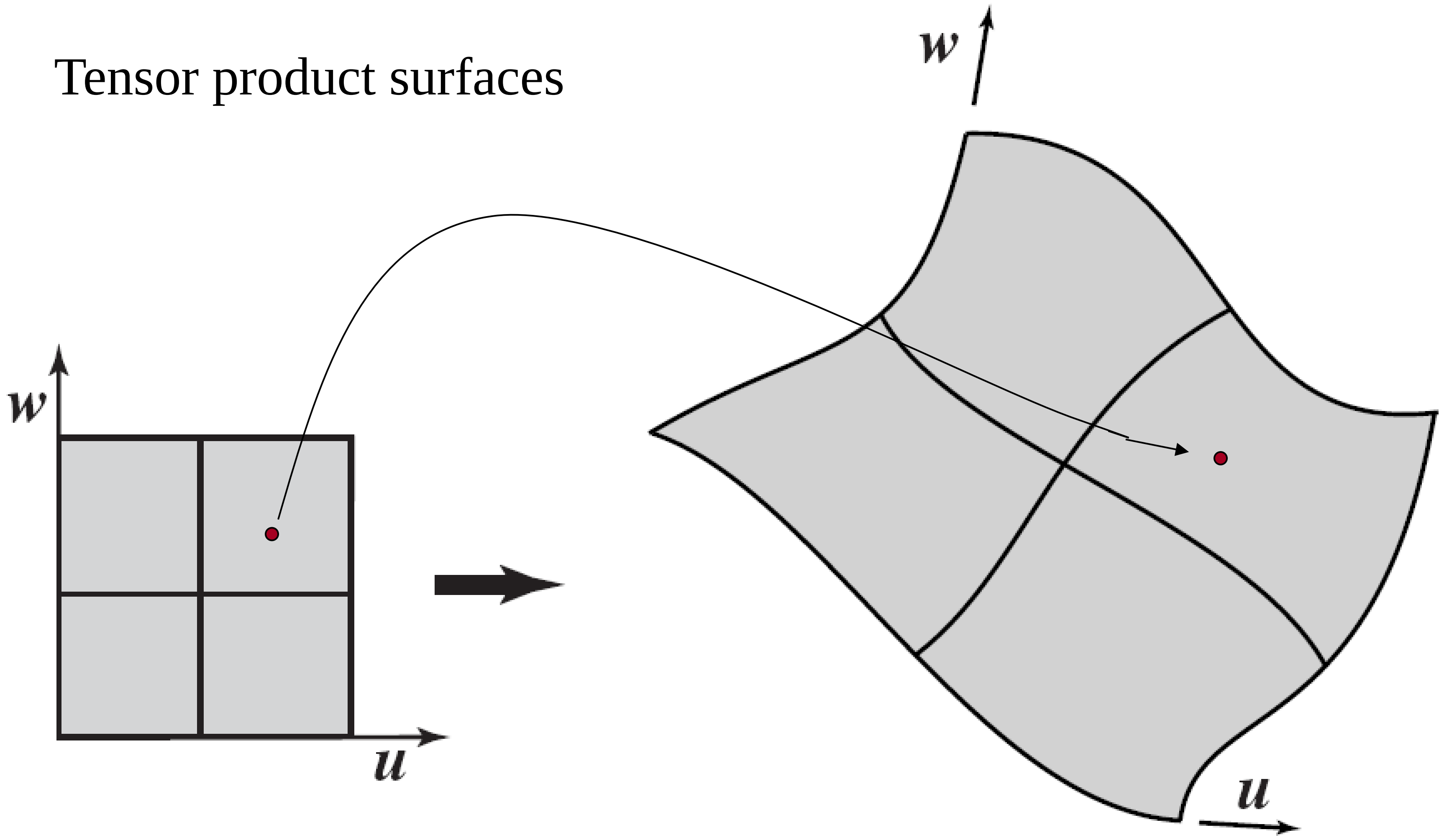


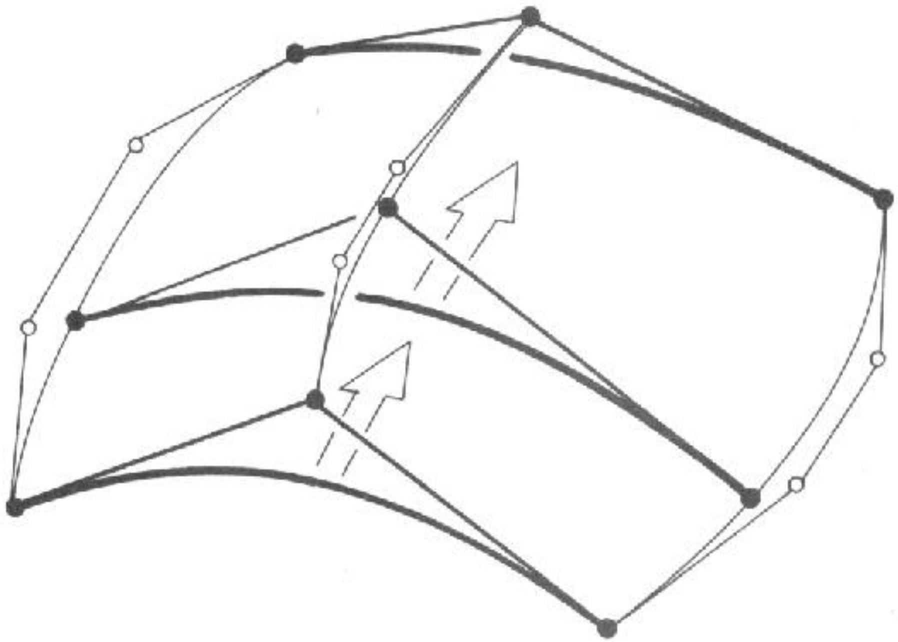
Territorial data

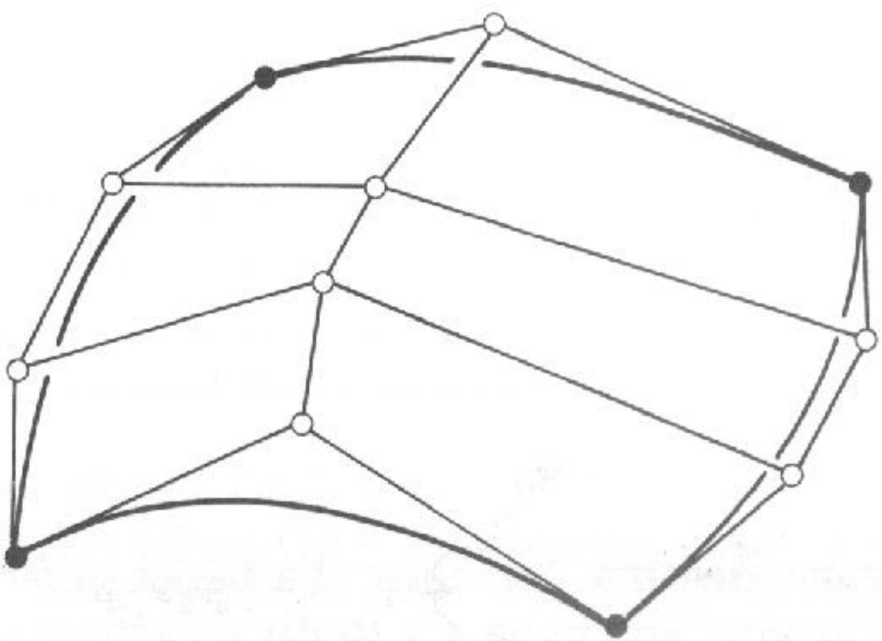
16K x 16K verteces
~537 milion of triangles

Parametric Surfaces

Tensor product surfaces





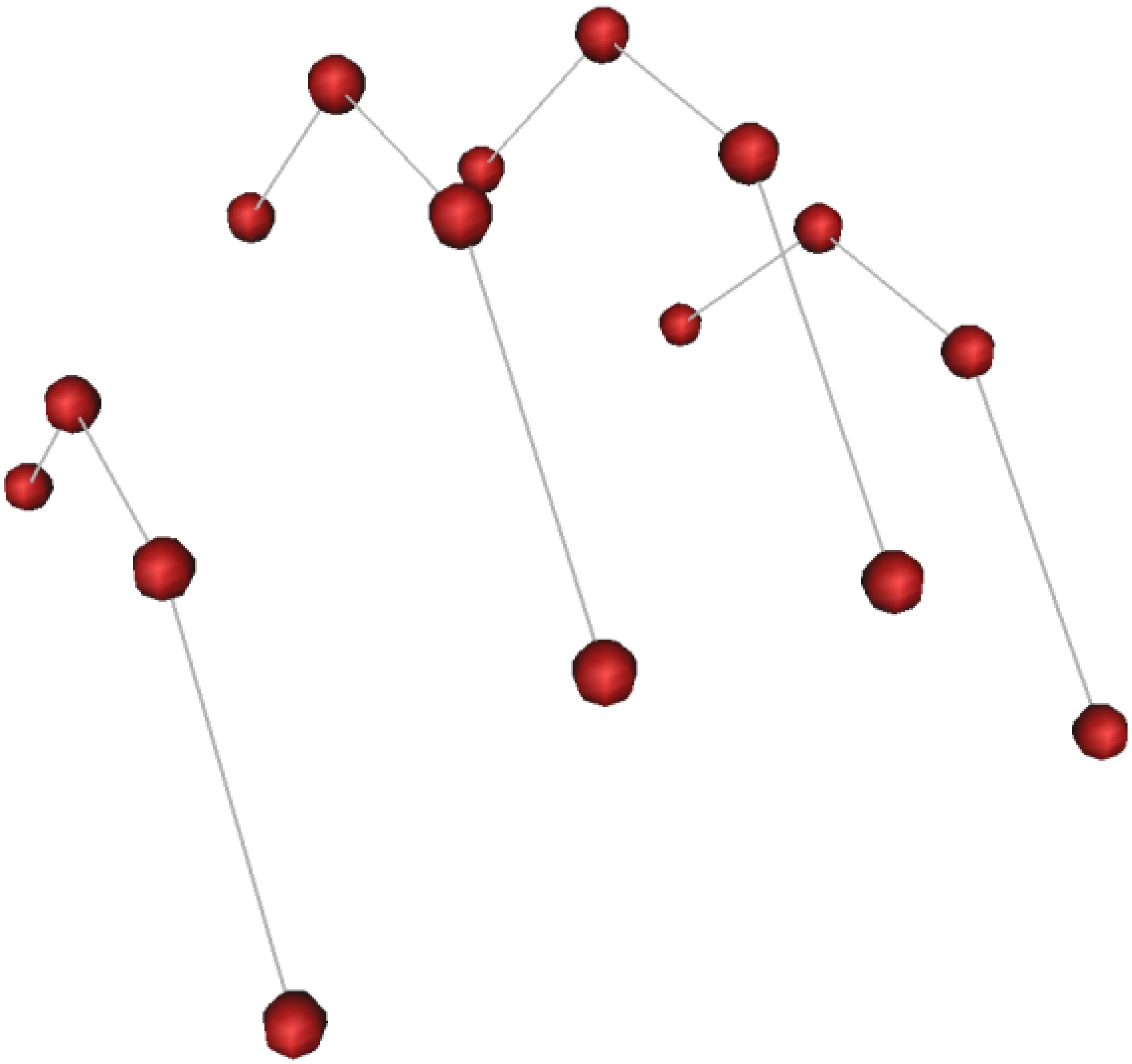


Bézier Patch

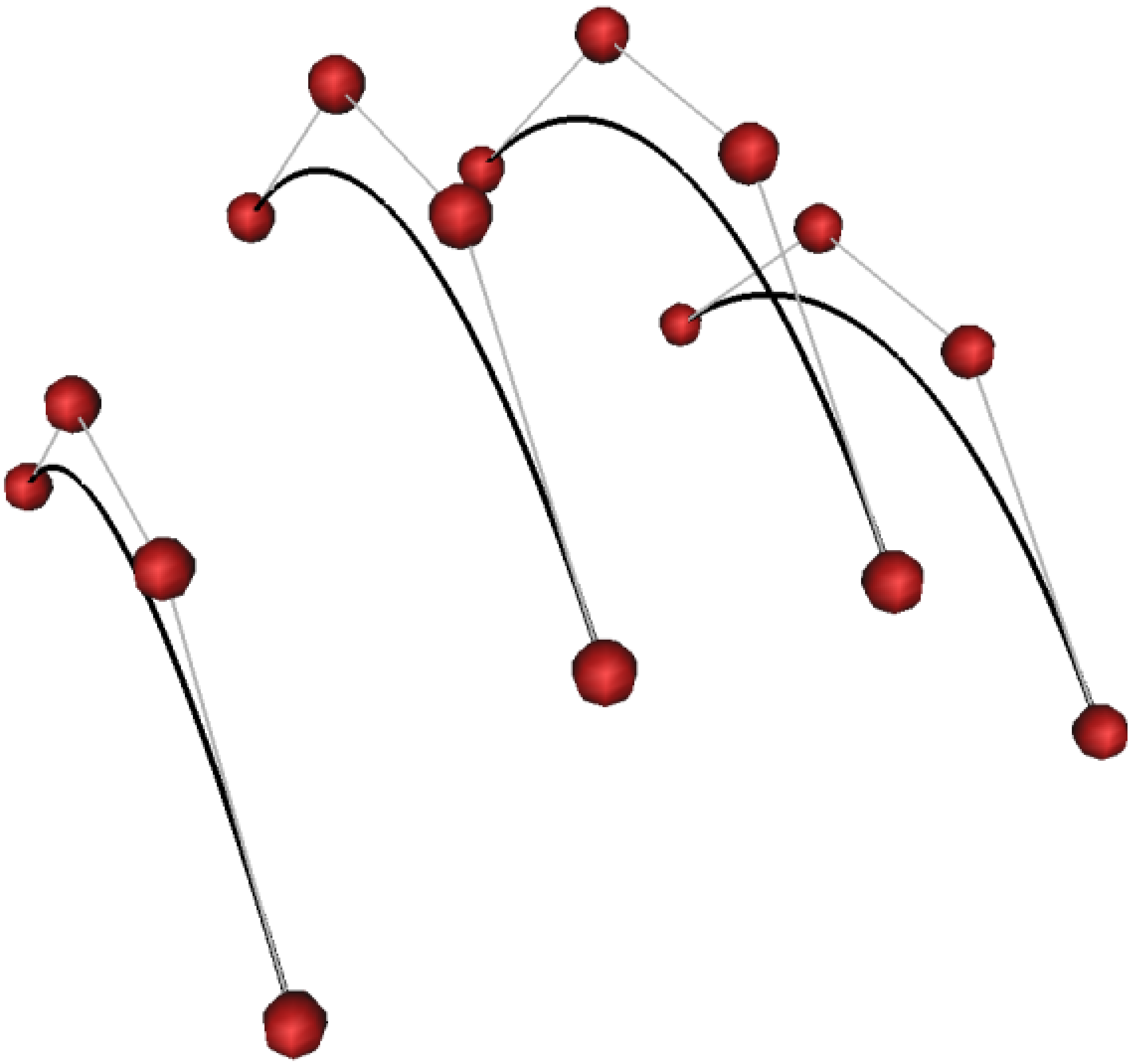
A surface is the locus of a curve that is moving through space and thereby changing its shape.

A surface is obtained by moving the control points of a Bézier curve along other Bézier curves.

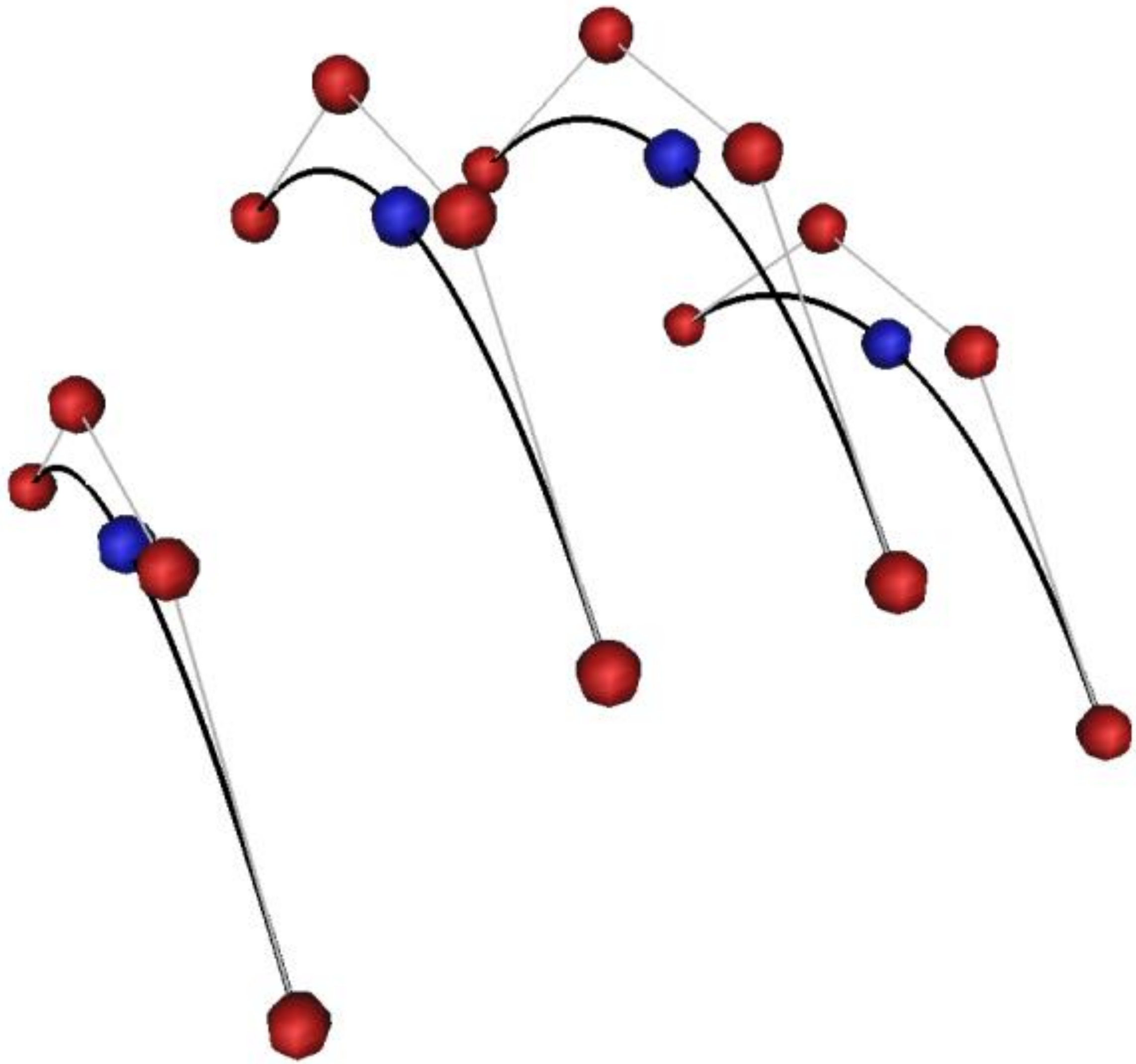
Bézier Patch



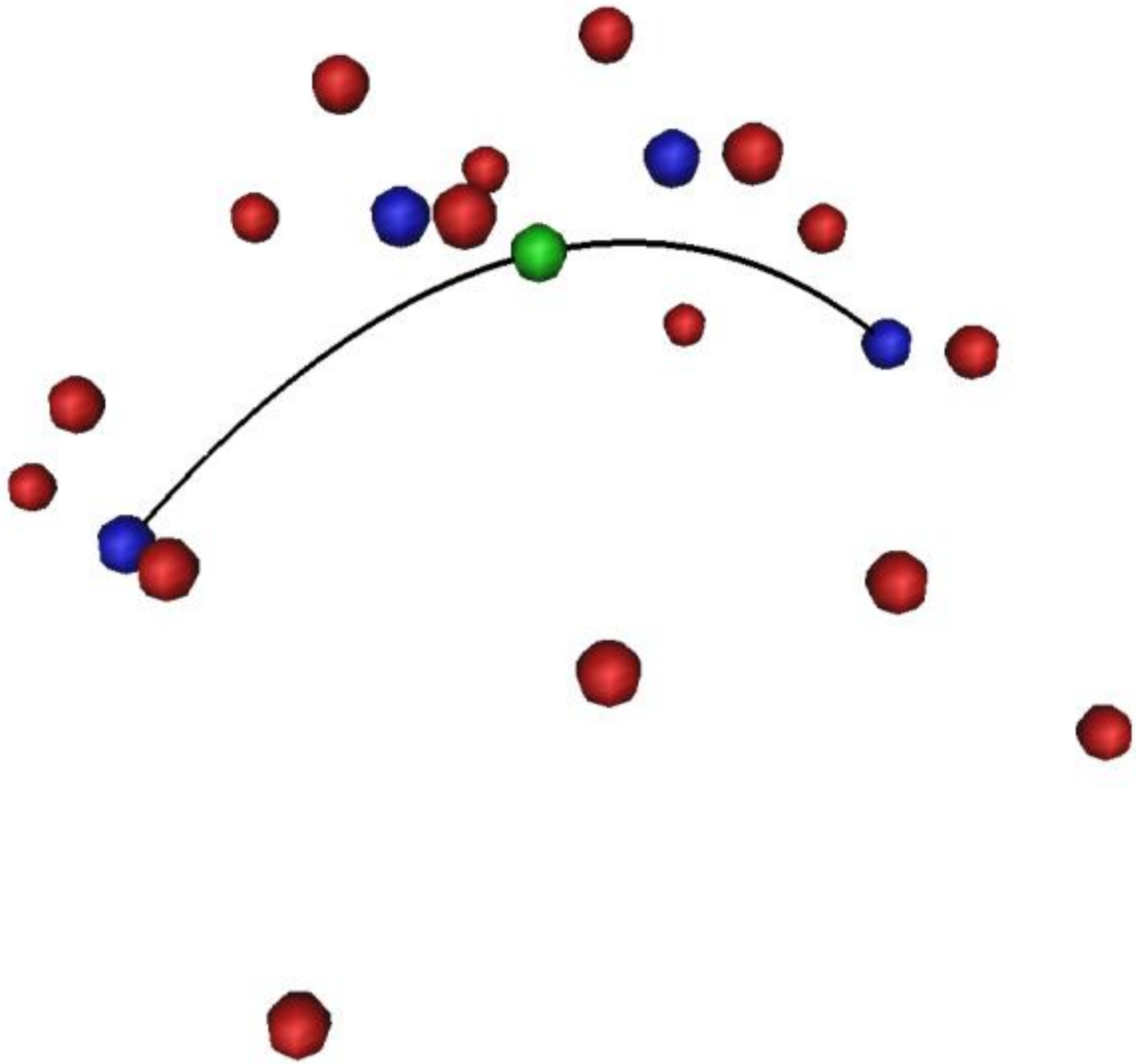
Bézier Patch



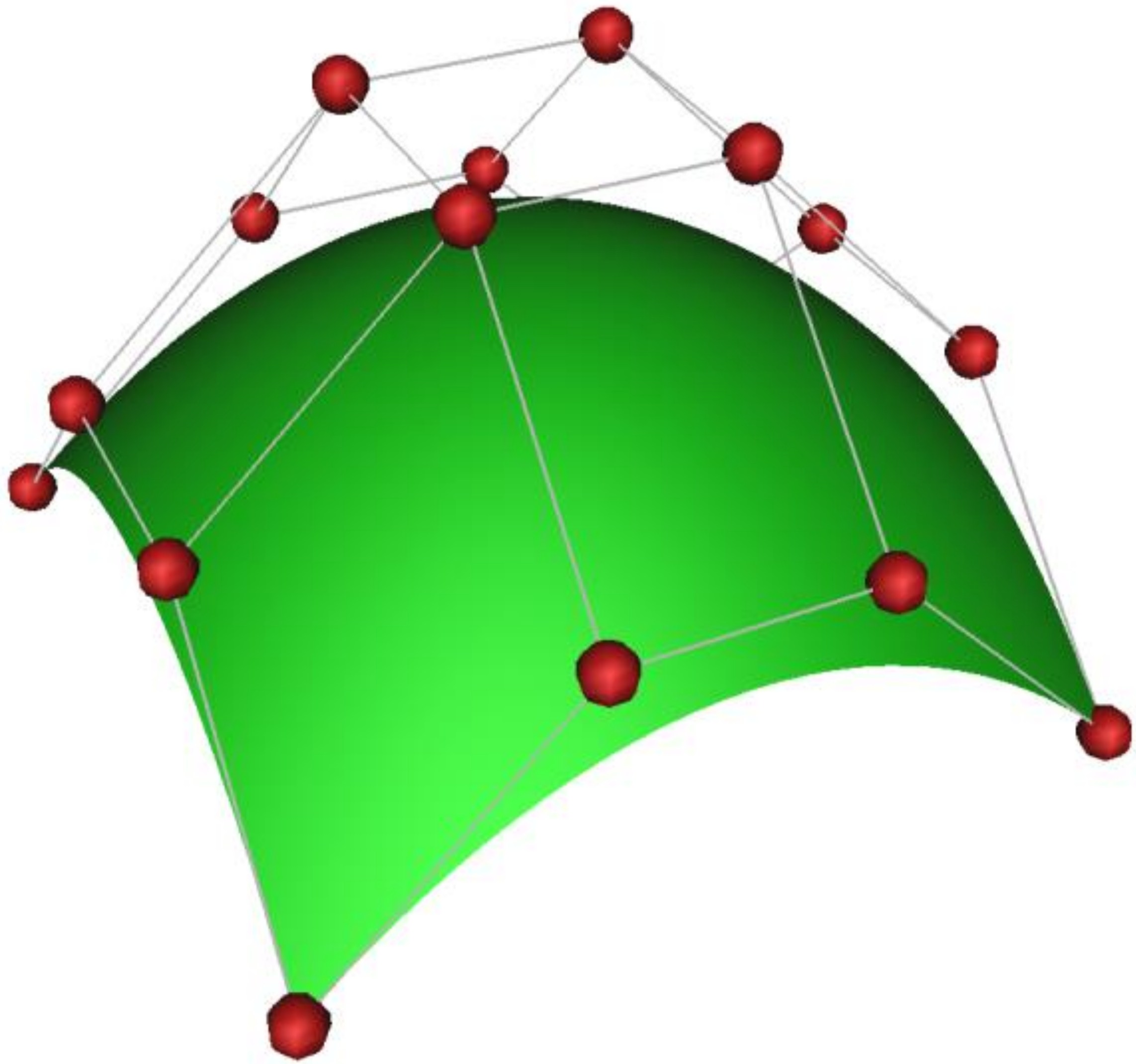
Bézier Patch



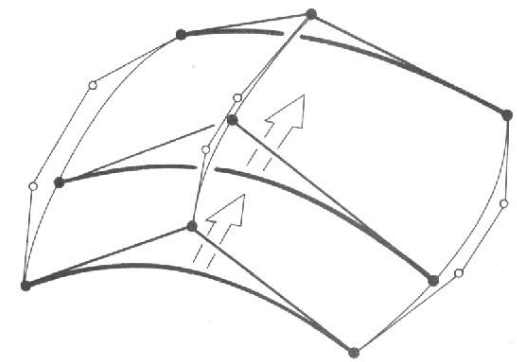
Bézier Patch



Bézier Patch



Bézier Patch



Given an initial Bézier curve of degree m

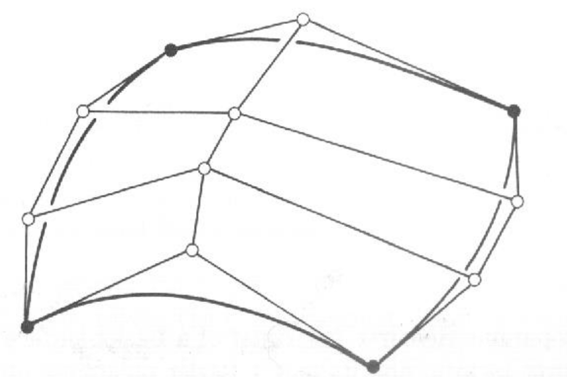
$$c(u) = \sum_{j=0}^{m} b_j B_j(u)$$

Let each control point b_j traverse a Bézier curve of degree n

$$b_j = c_j(v) = \sum_{i=0}^{n} b_{ij} B_i(v)$$

Combine these two eqs. and obtain the surface:

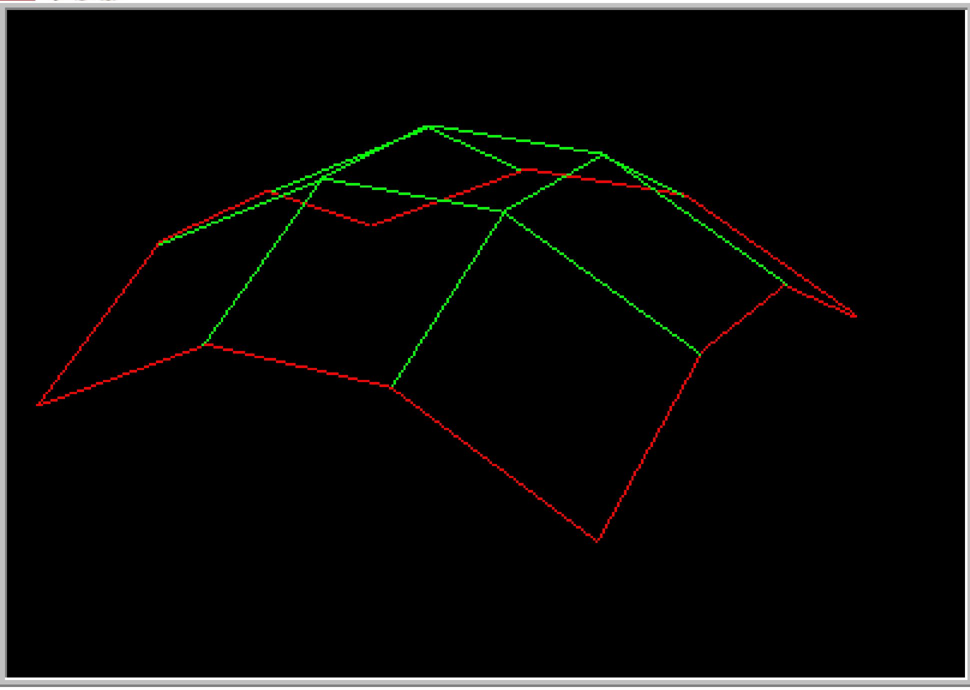
$$s(u,v) = \sum_{j=0}^{m} \sum_{i=0}^{n} b_{ij} B_i(v) B_j(u)$$

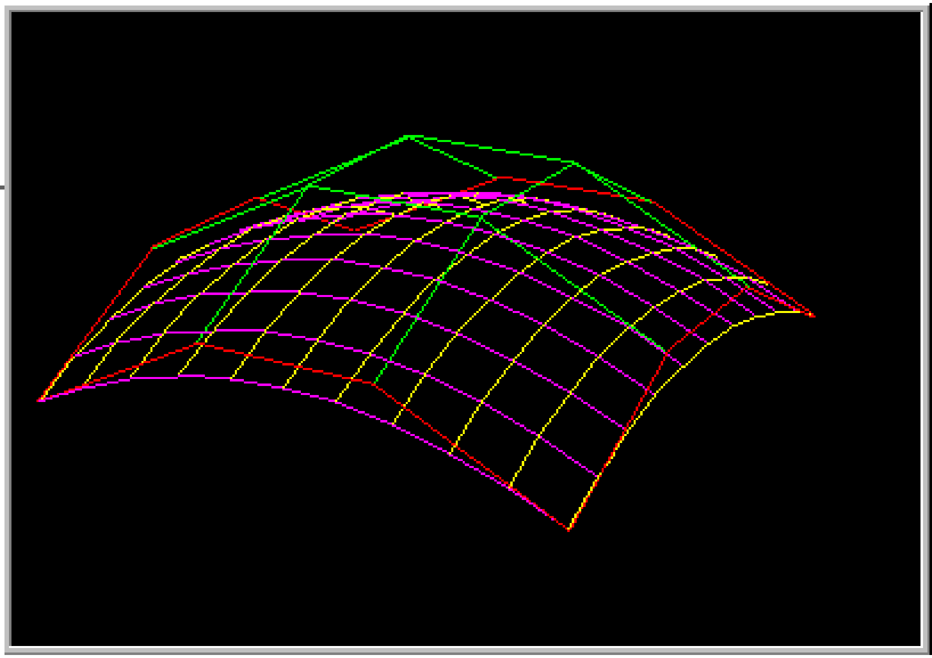


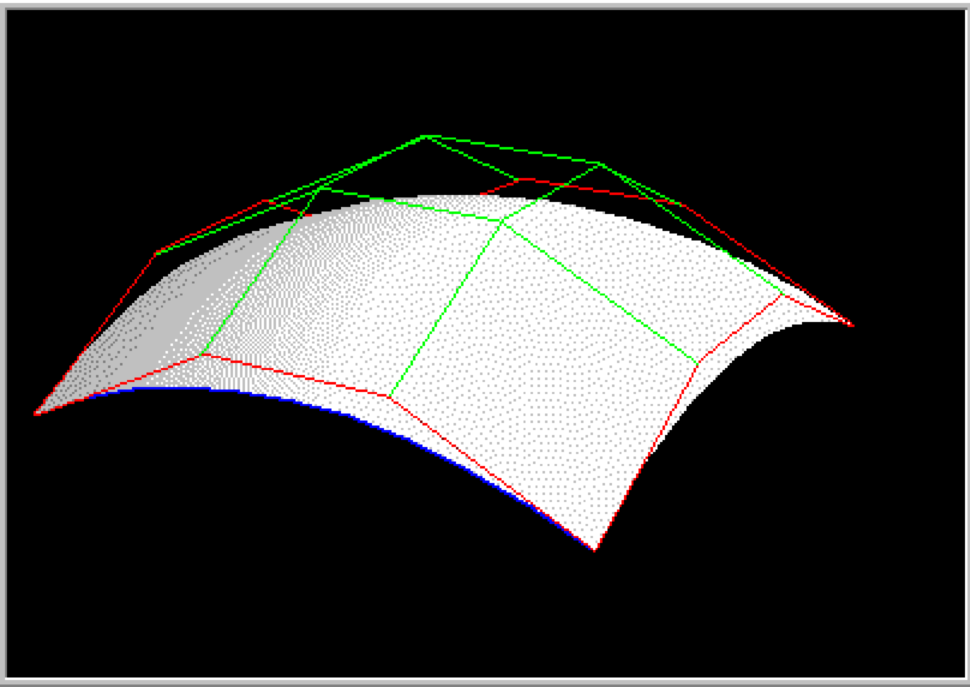
b_{ij} control points

Control mesh : (n+1)x(m+1)

Tensor-product surface defined on a rectangular parameter domain





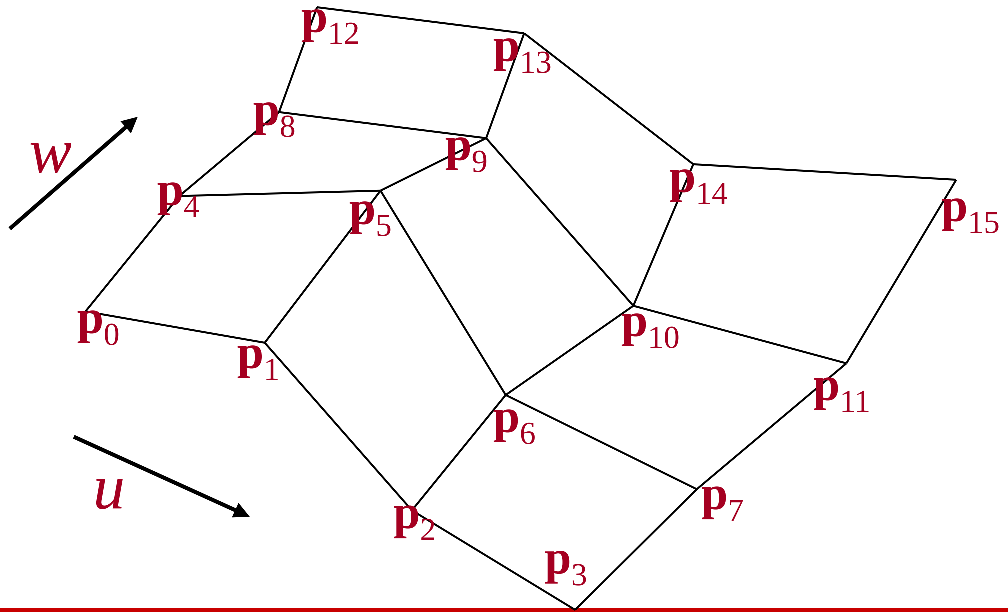


$$s(u,v) = \sum_{i=0}^{3}\sum_{j=0}^{3} b_{ij} B_j^3(u) B_i^3(v)$$

Example:
bicubic Bézier Patch

Control Mesh

- Consider a *bicubic* Bézier surface (bicubic means that it is a cubic function in both the *u* and *w* parameters)
- A cubic curve has 4 control points, and a bicubic surface has a grid of 4x4 control points, $\mathbf{p}_0$ through $\mathbf{p}_{15}$



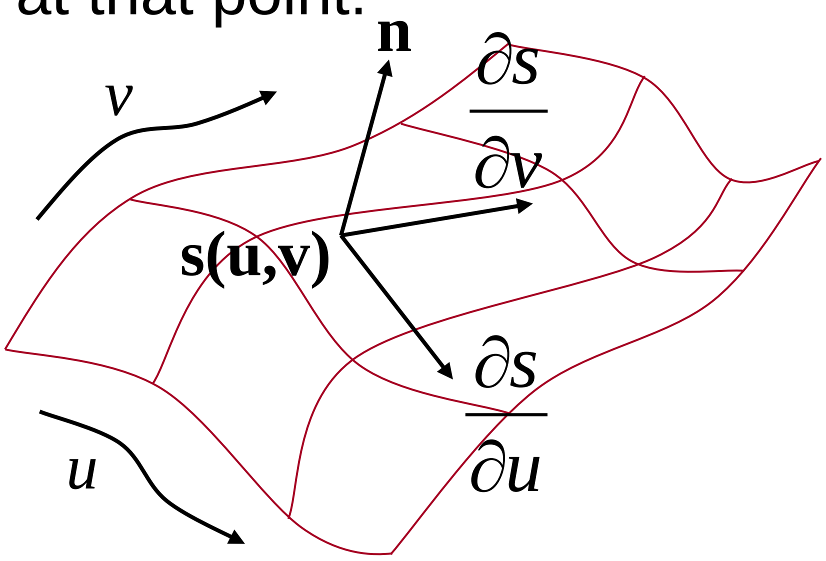
Normal Vectors

The normal vector **n** of a parametric surface is a normalized vector that is normal to the surface in a given point (u,v).

It is computed by the cross product of any two vectors that are tangent to the surface at that point:

$$\mathbf{n}^* = \frac{\partial s(u,v)}{\partial u} \times \frac{\partial s(u,v)}{\partial v}$$

$$\mathbf{n} = \frac{\mathbf{n}^*}{\left\| \mathbf{n}^* \right\|}$$



Properties

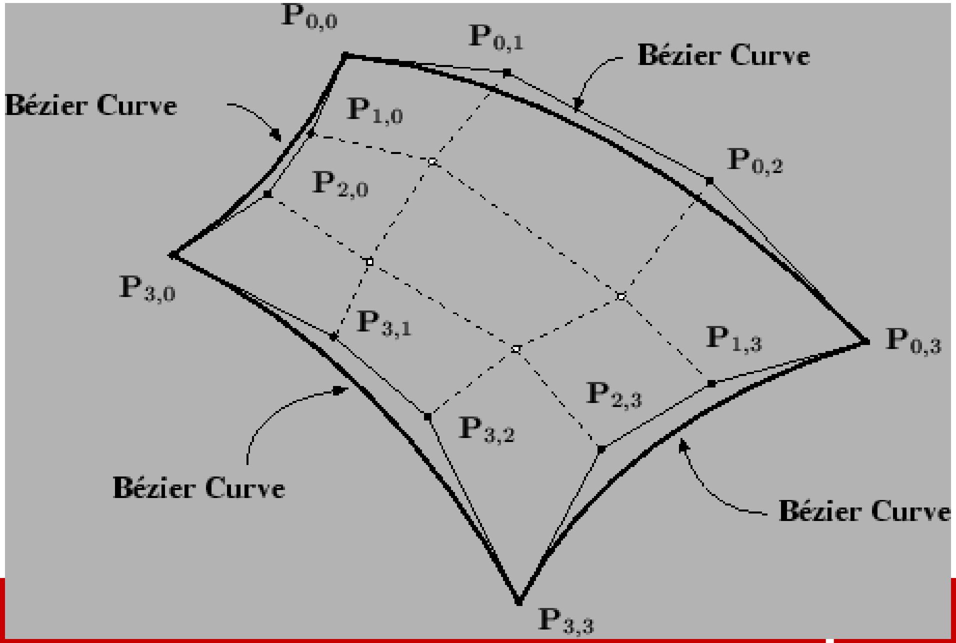
- The patch interpolates the four control points corners of the control mesh

$$s(0,0) = b_{00} \quad s(0,1) = b_{0n}, \quad s(1,0) = b_{n0}, \quad s(1,1) = b_{nn}$$

- Boundary curves: The 4 boundaries of the Bézier surface are just Bézier curves defined by the points on the edges of the surface.

$$s(u,0) = \sum_{i=0}^{n} b_{i0} B_i^n(u)$$

$$s(u,1), s(0,v), s(1,v)$$



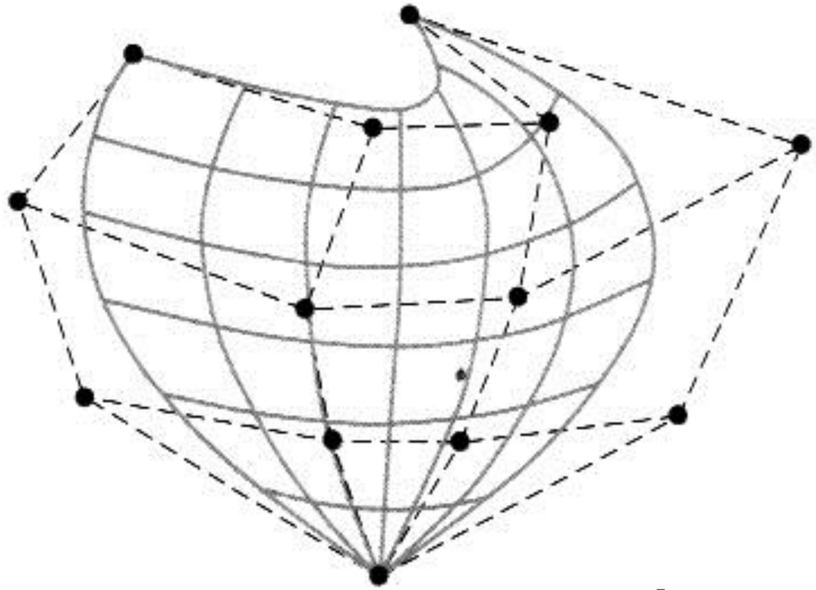
Properties

- Partition of Unity $\sum_{i=0}^{n}\sum_{j=0}^{m} B_j^m(u)B_i^n(v) \equiv 1$
- Convex Hull property: for 0 ≤ u,v ≤1, the terms
  $B_i^n(v), B_j^m(u)$ are non-negative.
  Then, taking into account the partition of unity property,
  $$s(u,v) = \sum_{i=0}^{n}\sum_{j=0}^{m} b_{ij} B_j^m(u) B_i^n(v)$$
  is a convex combination. The Bézier patch will fall within the convex hull of the control points.
- Affine Invariance: each affine transformation of the control mesh defines a new Bézier patch which is the transformation of the original.

Properties

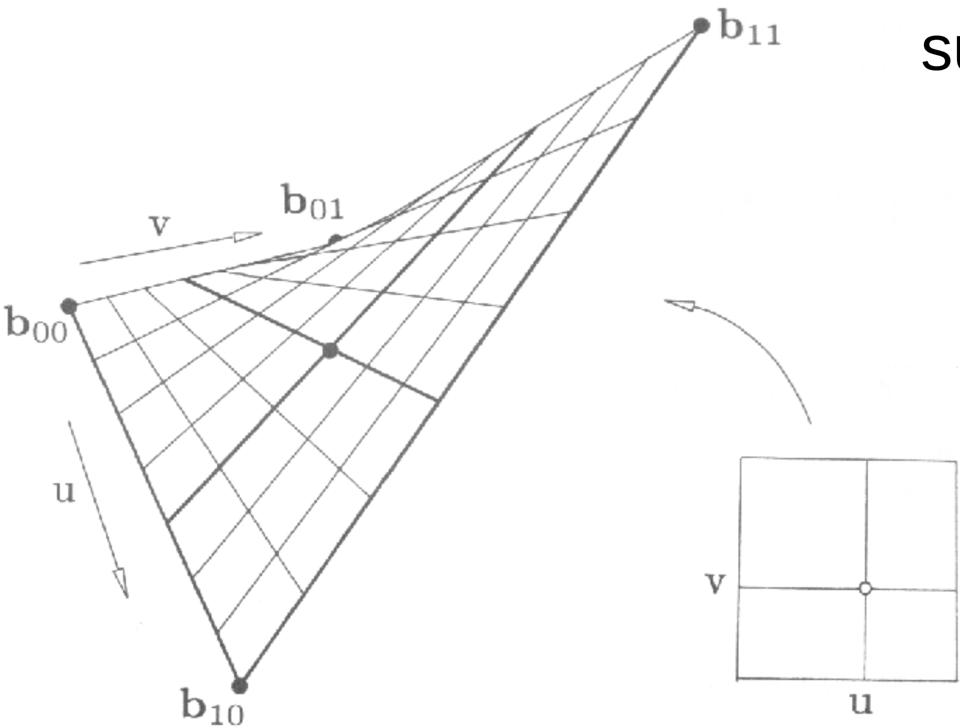
- Linear Precision: when all the control points lie on a plane, then the patch lies on the same plane.

- Shape approximation: the control mesh approximates the shape of the patch

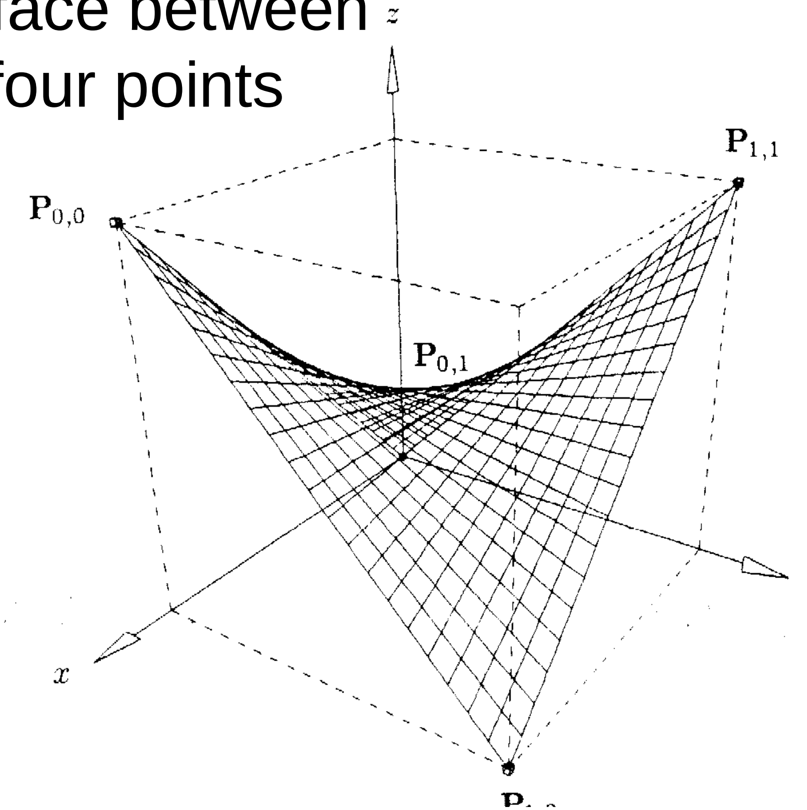


Bilinear Surface: Bézier patch of degree (1,1)

Hyperbolic paraboloid



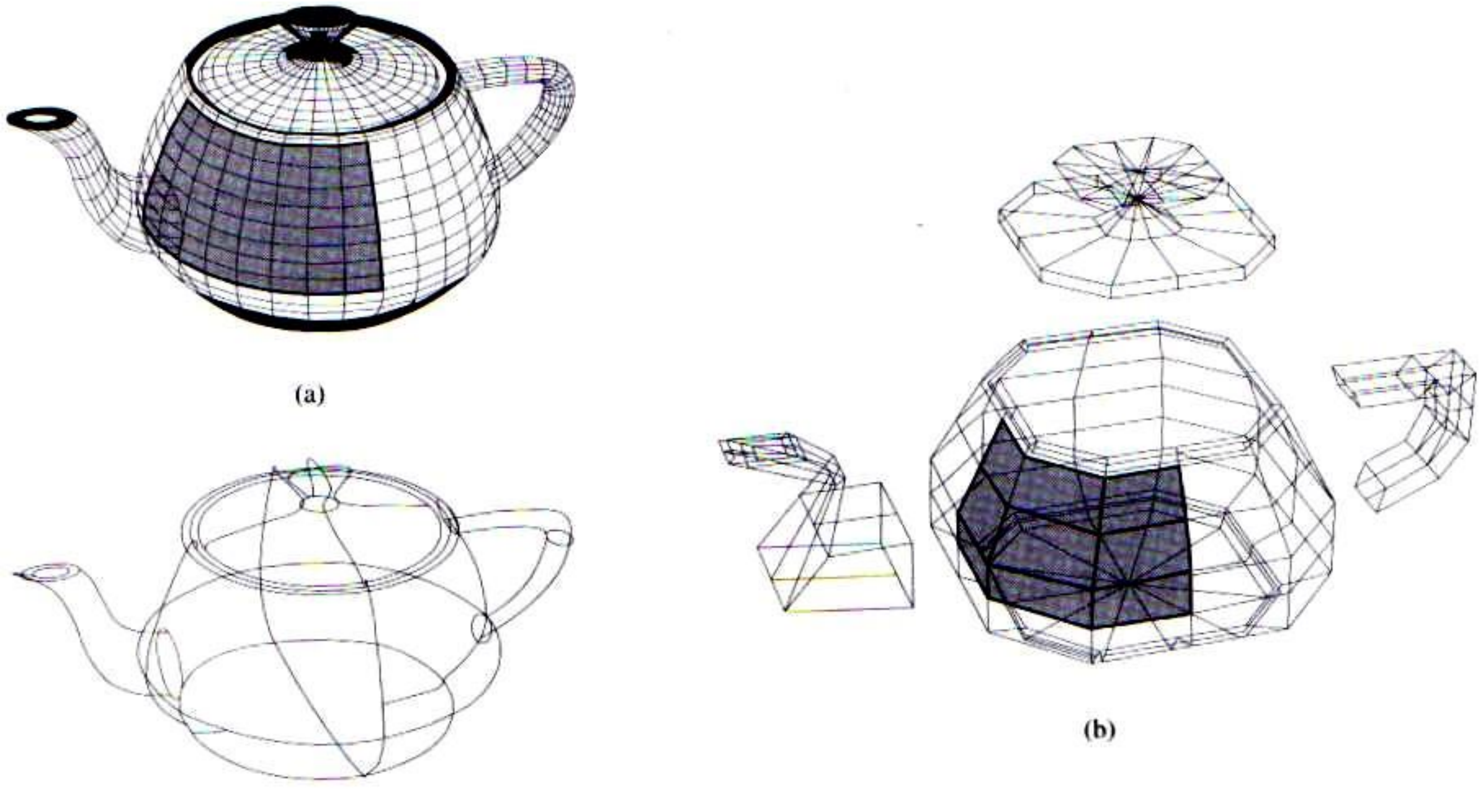
The "simplest" surface between four points



$$s(u,v)=\sum_{i=0}^{1}\sum_{j=0}^{1} b_{ij}B_j(u)B_i(v)=[(1-u)b_{00}+ub_{10}](1-v)+[(1-u)b_{01}+ub_{11}]v$$

A Bézier patch object the Utah teapot

- 32 patches $\times$ 16 control points/patch
  = 288 vertices
  = 288 $\times$ 3 real numbers



(a)

(b)

(c)

Tensor product Spline Surfaces

Given two knot vectors U and V on a parametric domain [0,1]x[0,1],

$$U = \{\underbrace{a,..,a}_{n}, u_1,..,u_K, \underbrace{b,..,b}_{n}\} \quad V = \{\underbrace{a,..,a}_{m}, v_1,..,v_L, \underbrace{b,..,b}_{m}\}$$

the spline surface of orders n and m defined by the bidirectional net of control points P_{ij} is:

$$s(u,v) = \sum_{i=1}^{n+K} \sum_{j=1}^{m+L} P_{ij} N_{i,n}(u) N_{j,m}(v)$$

where the CP define the Control Mesh

$N_{i,n}(u),N_{j,m}(v)$ univariate B-Spline basis functions of degree n-1 and m-1

Spline surfaces

$$s(u,v) = \begin{pmatrix} s_x(u,v) \\ s_y(u,v) \\ s_z(u,v) \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{n+K} \sum_{j=1}^{m+L} x_{ij} N_{i,n}(u) N_{j,m}(v) \\ \sum_{i=1}^{n+K} \sum_{j=1}^{m+L} y_{ij} N_{i,n}(u) N_{j,m}(v) \\ \sum_{i=1}^{n+K} \sum_{j=1}^{m+L} z_{ij} N_{i,n}(u) N_{j,m}(v) \end{pmatrix}$$

$P_{ij=}$ (x_{ij},y_{ij},z_{ij}) control points Control mesh : KxL

Properties

- The properties of the tensor product basis functions

$$N_{i,n}(u)N_{j,m}(v)$$

follow from the corresponding properties of the univariate basis functions (nonnegativity, partition of unity, ..)

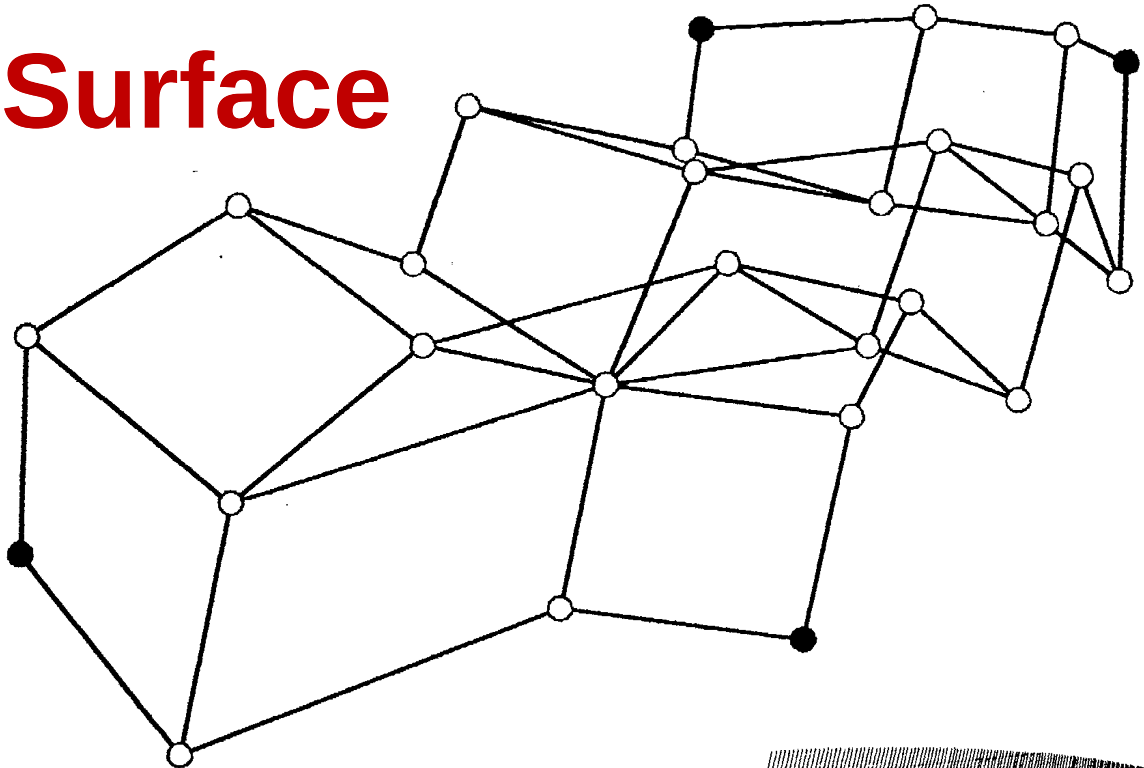
Local support:

$$N_{i,n}(u)N_{j,m}(v)=0$$

$$\textit{if } (u,v) \textit{ is outside the rectangle } [u_i,u_{i+n})\times[v_j,v_{j+m})$$

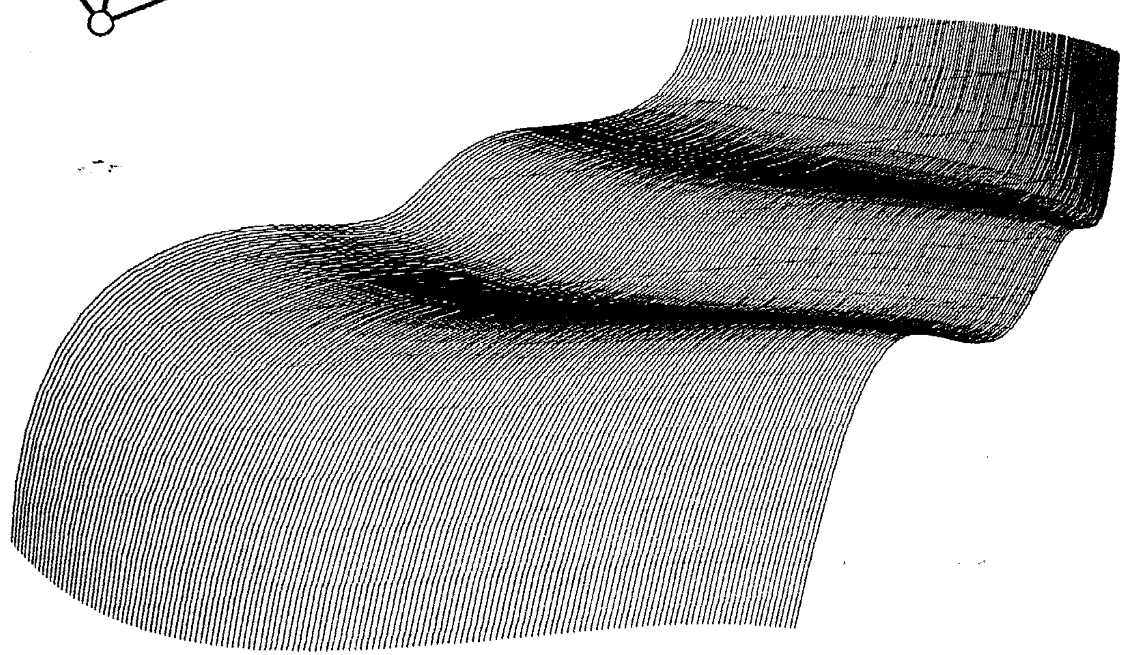
- The spline surface have the following properties:
  - Affine invariance
  - Local convex hull property
  - Local modification: if P_{ij} is moved it affects the surface only in the rectangle $[u_i,u_{i+n})\times[v_j,v_{j+m})$
  - No variation diminishing property for spline surface

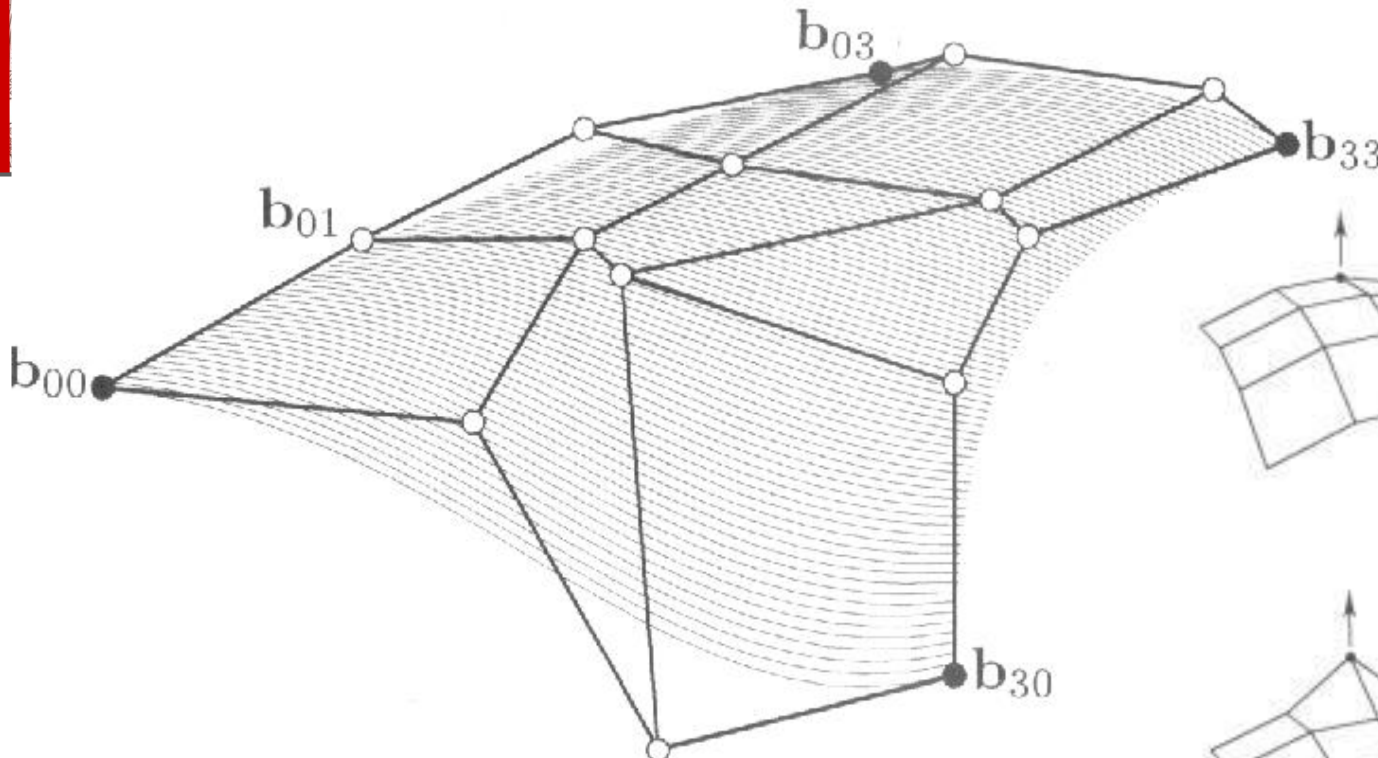
Spline Surface

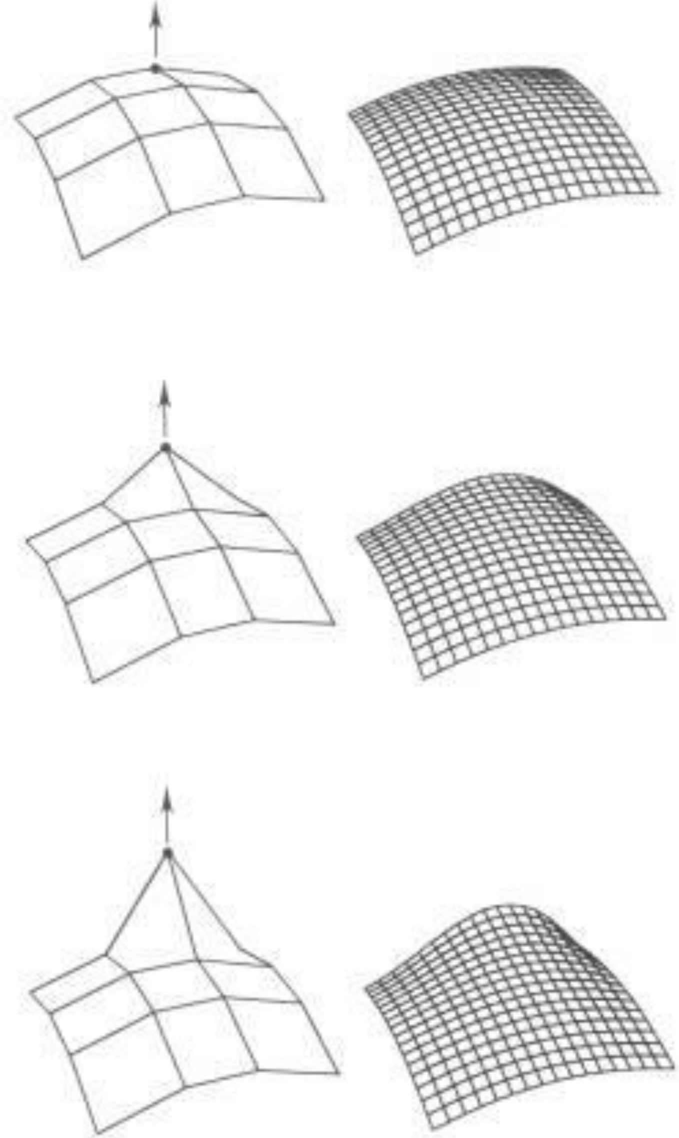


UxV=[0,1]x[0,1]
Parametric domain

Example of tensor product bicubic spline (6x3 patch)







- Surface shape control

Non Uniform Rational B-Splines (NURBSs)

Let s^w be a spline surface in the homogeneous space 4D:

$$s^w(u,v)=\sum_{i=1}^{n+K}\sum_{j=1}^{m+L}P_{ij}^{\,w}N_{i,n}(u)N_{j,m}(v)$$

Project into 3D

$$s(u,v)=\begin{pmatrix} s_x(u,v)\\ s_y(u,v)\\ s_z(u,v)\end{pmatrix}=\frac{\sum_{i=1}^{n+K}\sum_{j=1}^{m+L}w_{ij}P_{ij}N_{i,n}(u)N_{j,m}(v)}{\sum_{i=1}^{n+K}\sum_{j=1}^{m+L}w_{ij}N_{i,n}(u)N_{j,m}(v)}$$

P_{ij} control points -- Control mesh : **KxL**

UxV=[0,1]x[0,1] Parametric domain, knot vectors

$N_{i,n}(u),N_{j,m}(v)$ B-Spline basis functions

w_{ij} **weights ≥0**

Non Uniform Rational B-Splines (NURBSs)

A NURBS surface of degree n-1 in the u-direction and m-1 in the v direction is a bivariate vector-valued piecewise rational function of the form:

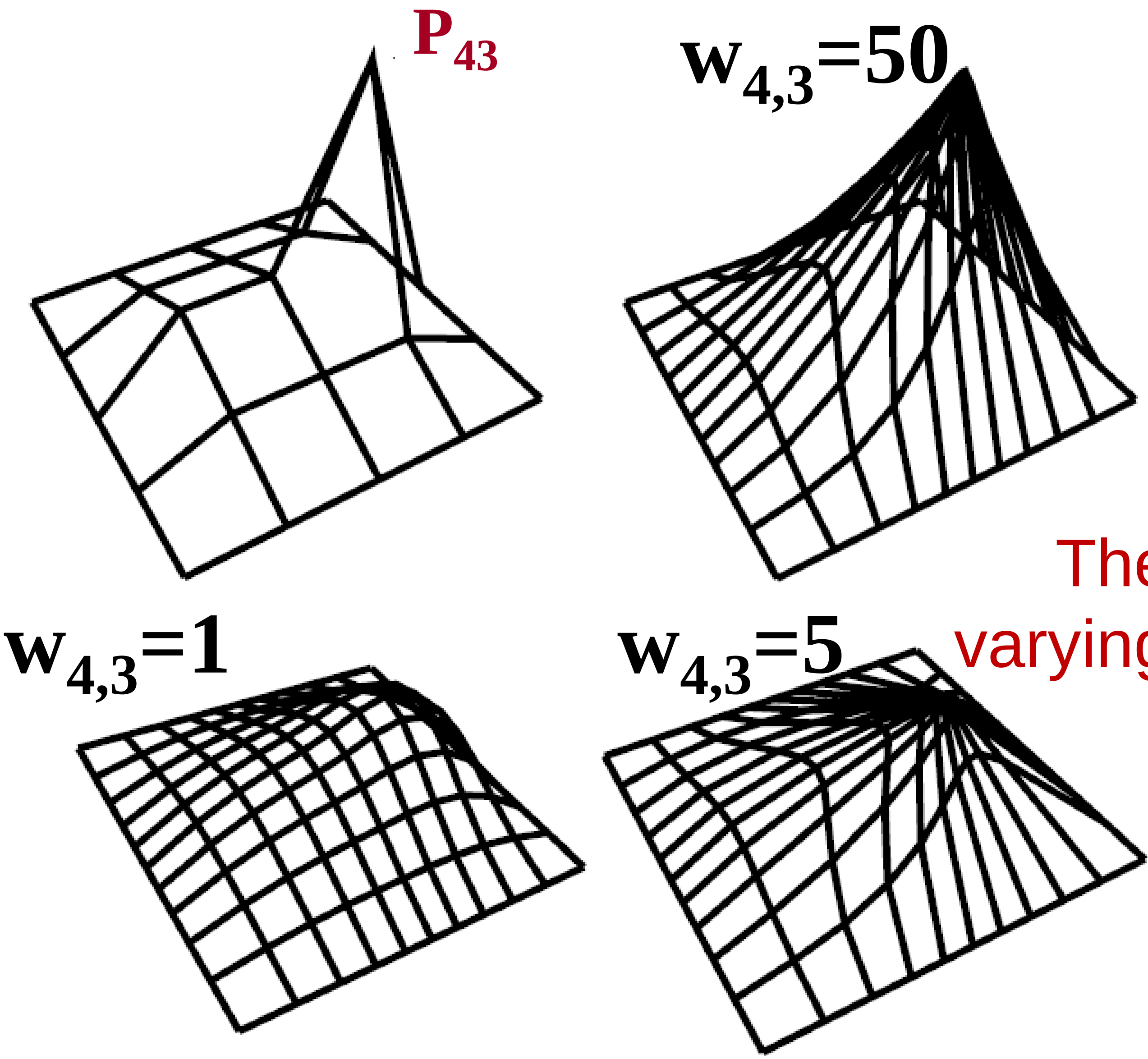
$$s(u,v) = \frac{\sum_{i=1}^{n+K} \sum_{j=1}^{m+L} w_{ij} P_{ij} N_{i,n}(u) N_{j,m}(v)}{\sum_{i=1}^{n+K} \sum_{j=1}^{m+L} w_{ij} N_{i,n}(u) N_{j,m}(v)}$$

Introducing the piecewise rational basis functions:

$$R_{ij}(u,v) = \frac{w_{ij} N_{i,n}(u) N_{j,m}(v)}{\sum_{i=1}^{n+K} \sum_{j=1}^{m+L} w_{ij} N_{i,n}(u) N_{j,m}(v)}$$

The surface can be written as

$$\boxed{s(u,v) = \sum_{i=1}^{n+K} \sum_{j=1}^{m+L} P_{ij} R_{i,j}(u,v)}$$



The effect of varying the weights

If w_i=1 for all i, then the NURBS is a non-rational spline surface

- Rational Bézier Patch: A NURBS surface without internal knots and open knot vector

- Closed Surface (no periodic): First column of the CP grid (or first row) coincides with the last column (or row)

- Corner: If all the CP in a subgrid (n-1)x(m-1) coincide, then the surface interpolates that CP, modelling a corner.

  By using multiply coincident CP, visual discontinuities can be created where there are no corresponding discontinuity in the basis functions.

- Planarity: If all the CP in a subgrid nxm lie on a plane, then the surface lies on the same plane.

How to store a spline/NURBS surface

FILENAME: namefile.snurbs

DEGREE_U_V

2 2

N.C.P._U_V

5 9

N.KNOTS_U_V

8 12

COORD.C.P.(X,Y,Z,W)

0.000000e+00 0.000000e+00 1.000000e+00 0.00000e+00

….

KNOTS_U

0.000000e+00

….

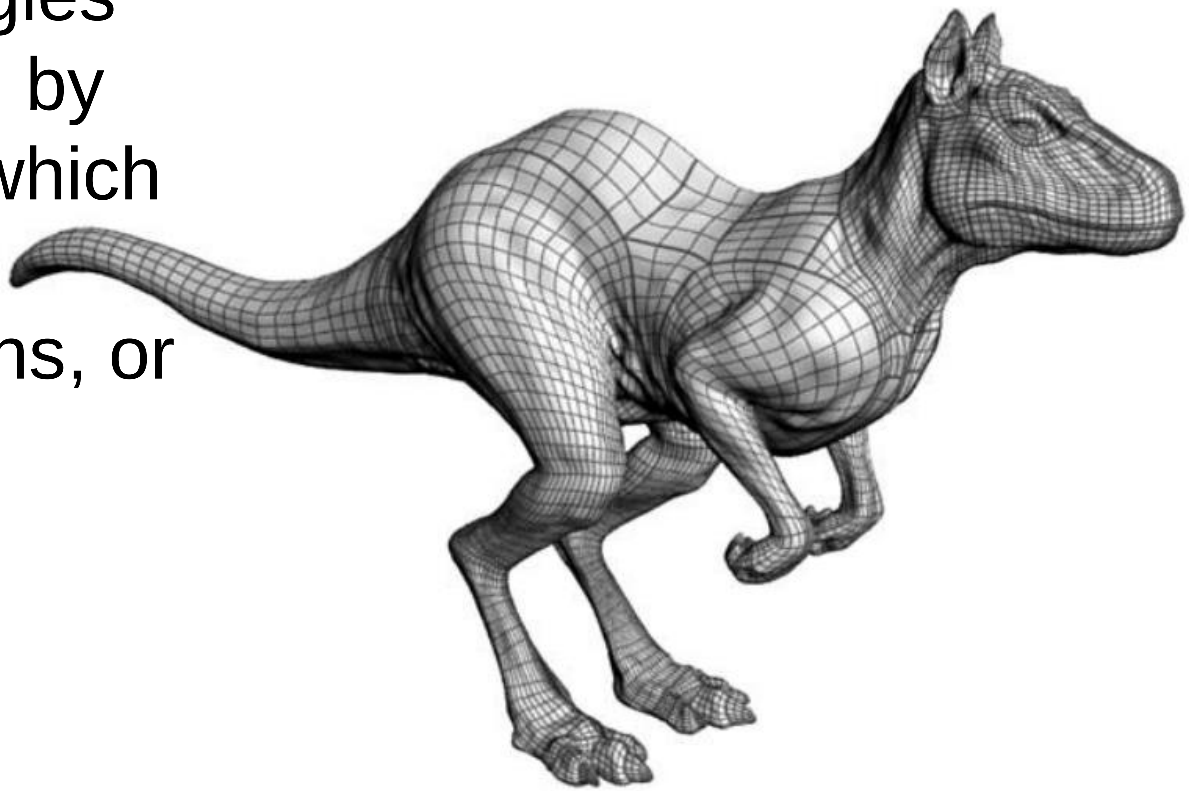
KNOTS_V

0.000000e+00

….

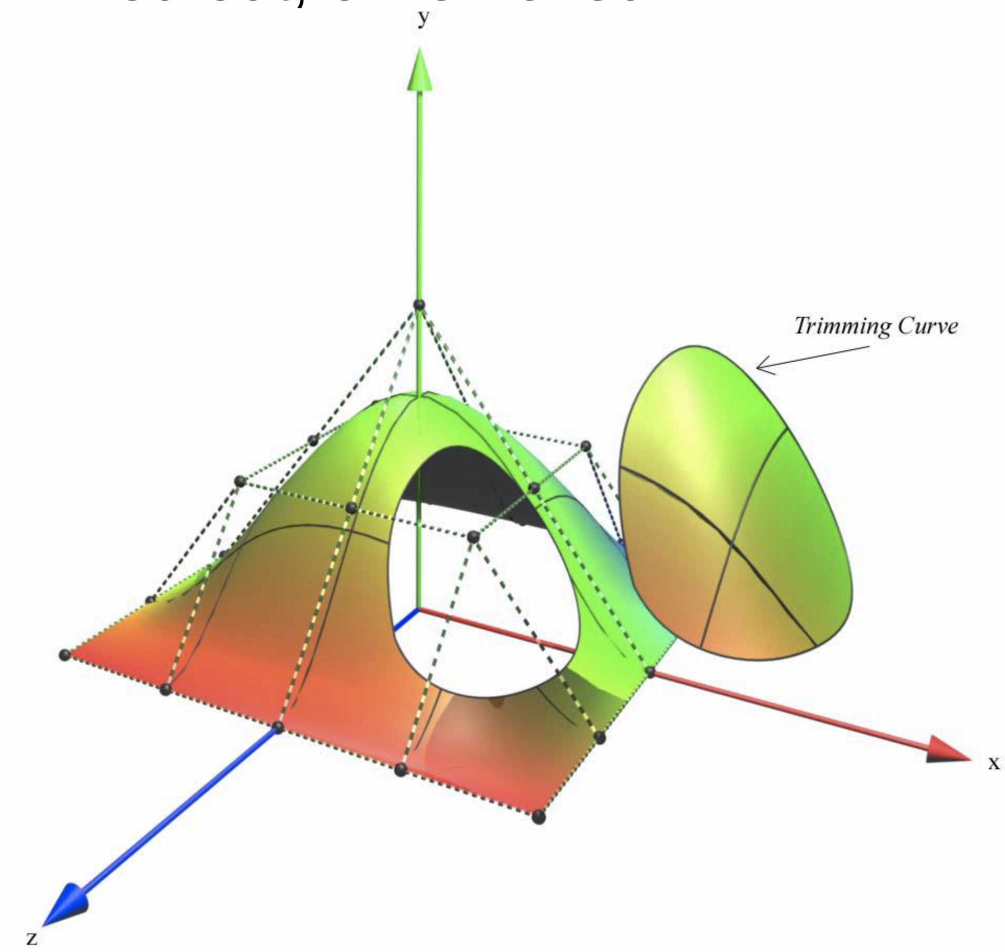
NURBS limits

- NURBS surfaces have rectangular topology
- Arbitrary topologies can be obtained by collapsing CP, which can cause bad parameterizations, or by joining patch together



Trimmed NURBS surfaces

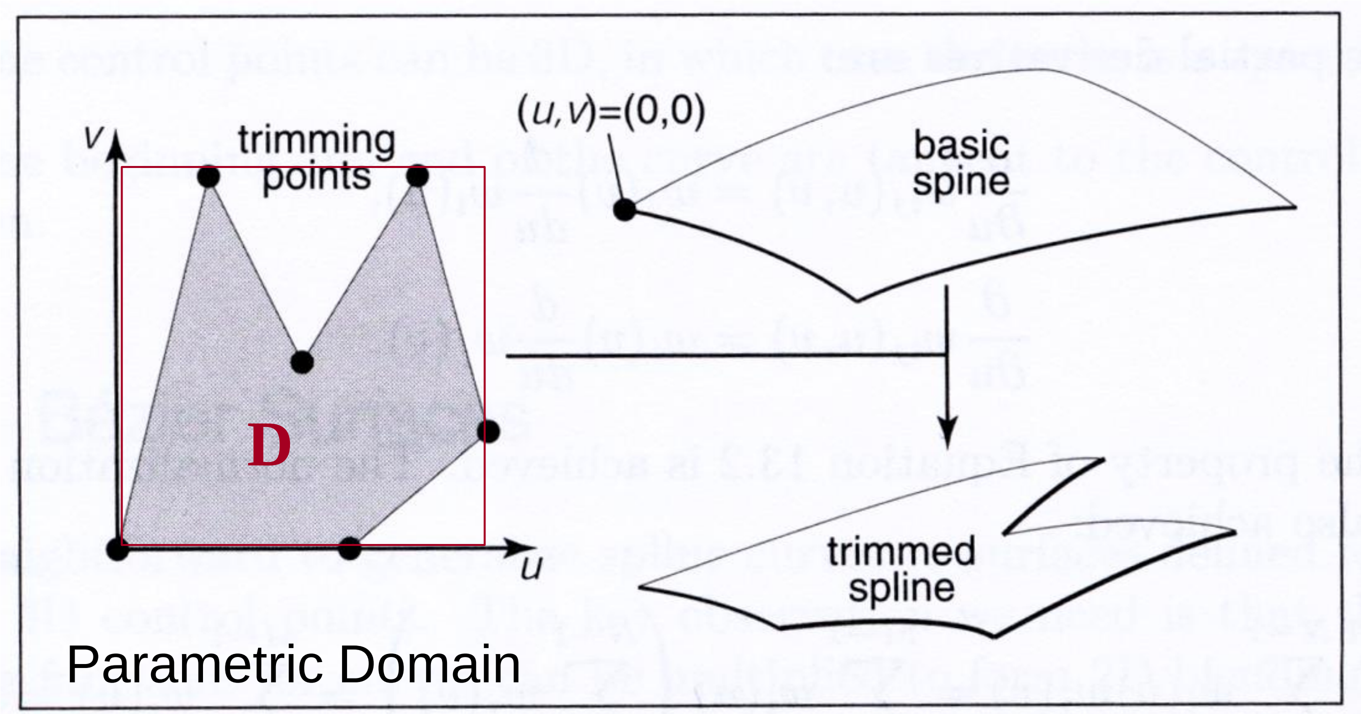
A trimmed NURBS surface is one in which specified patches have been trimmed out, or removed.



Trimmed NURBS surfaces

The surface S(u,v) is limited to a subdomain D in UxV of the parametric space, called Trimming Region (TR)

- TR specifies in the parametric domain regions of interest
- Visualize the original surface S(u,v) only on TR

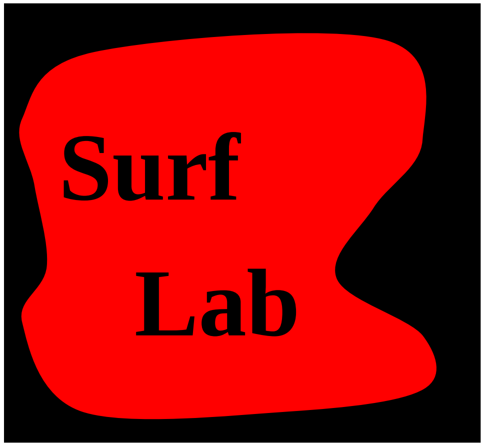


Trimming Region

- A trimming region is defined by a set of closed trimming loops in the parameter space of a surface.
- A trimming loop consists of a closed NURBS curve and/or piecewise linear curve.
- Self intersecting curves are not allowed.

$$c_k(t) = (x_k(t), y_k(t)) =$$

$$c_k(t) = \frac{\sum_{i=1}^{pk+K} w_i P_i N_{i,pk}(t)}{\sum_{i=1}^{pk+K} w_i N_{i,pk}(t)} \qquad k = 1,..,M$$

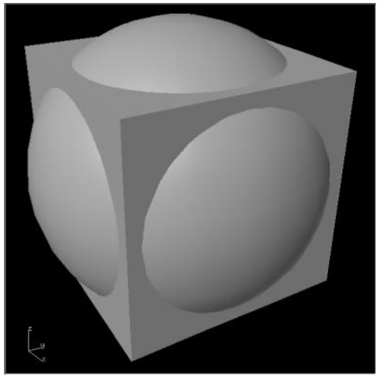
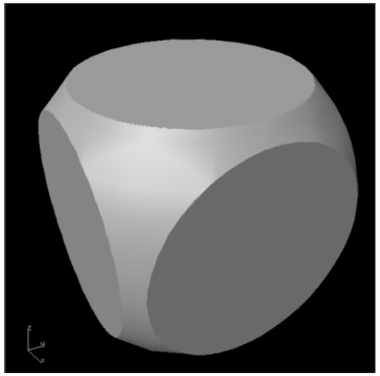
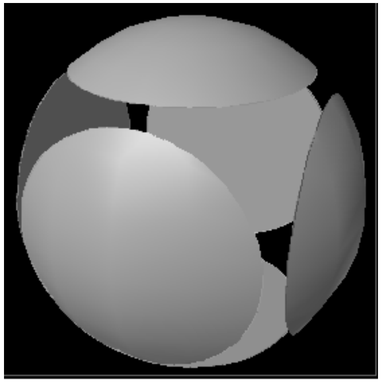


Parametric Domain

Trimmed NURBS surfaces

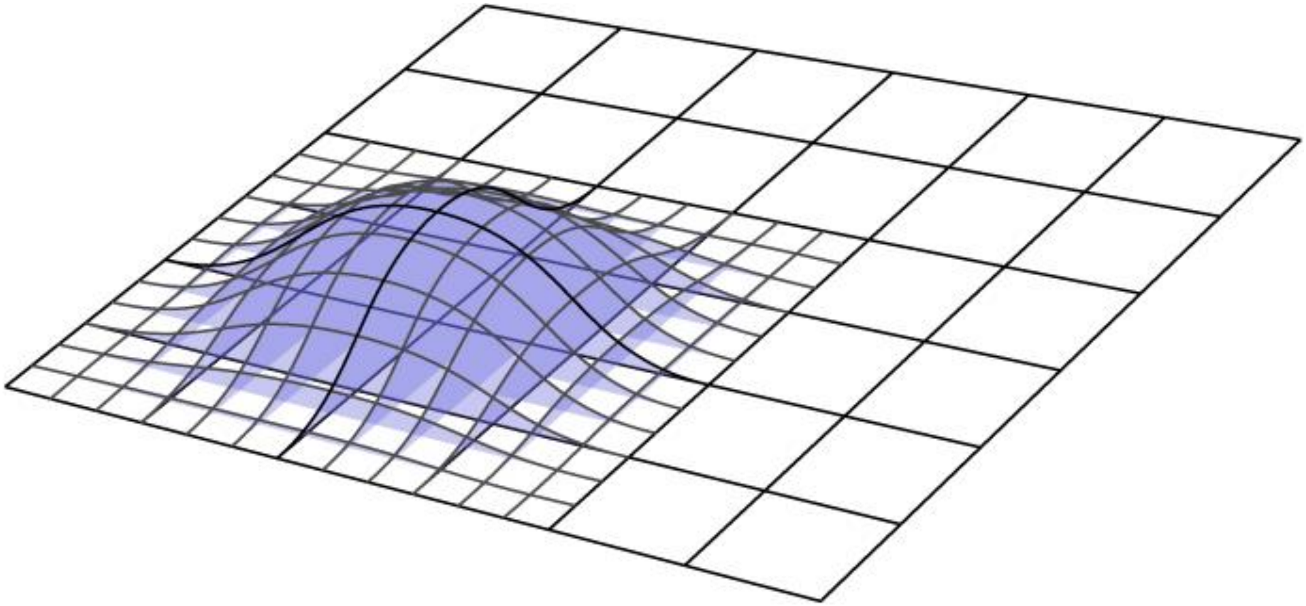
Applications:

- **Hierarchical modelling**
- Composing solids by Boolean Operations

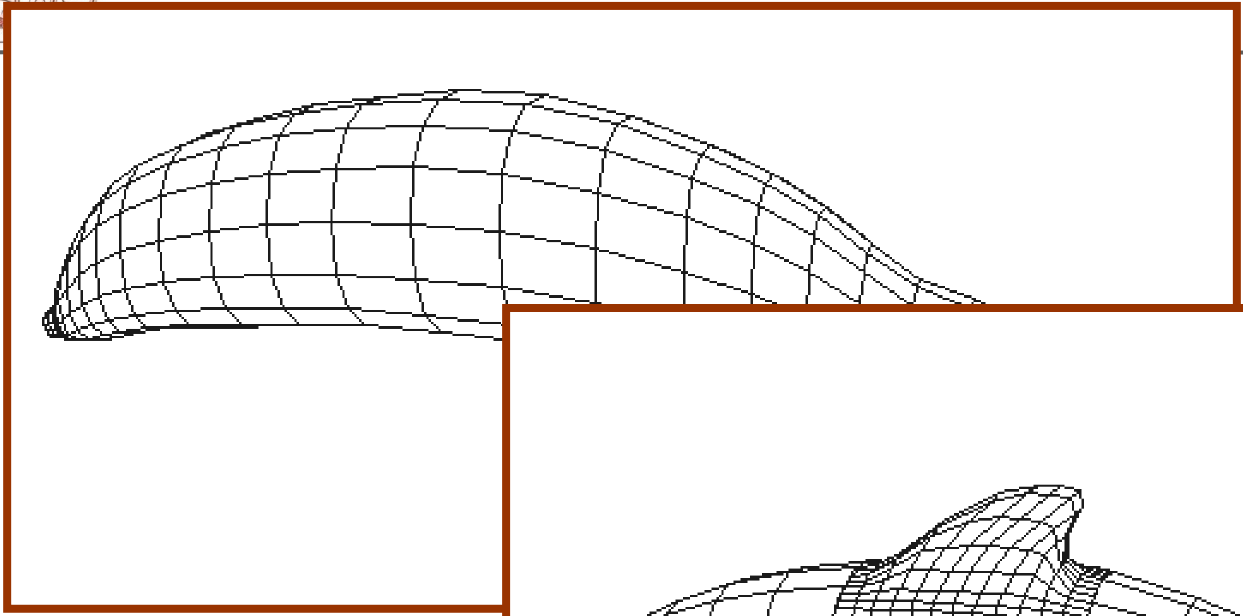
  

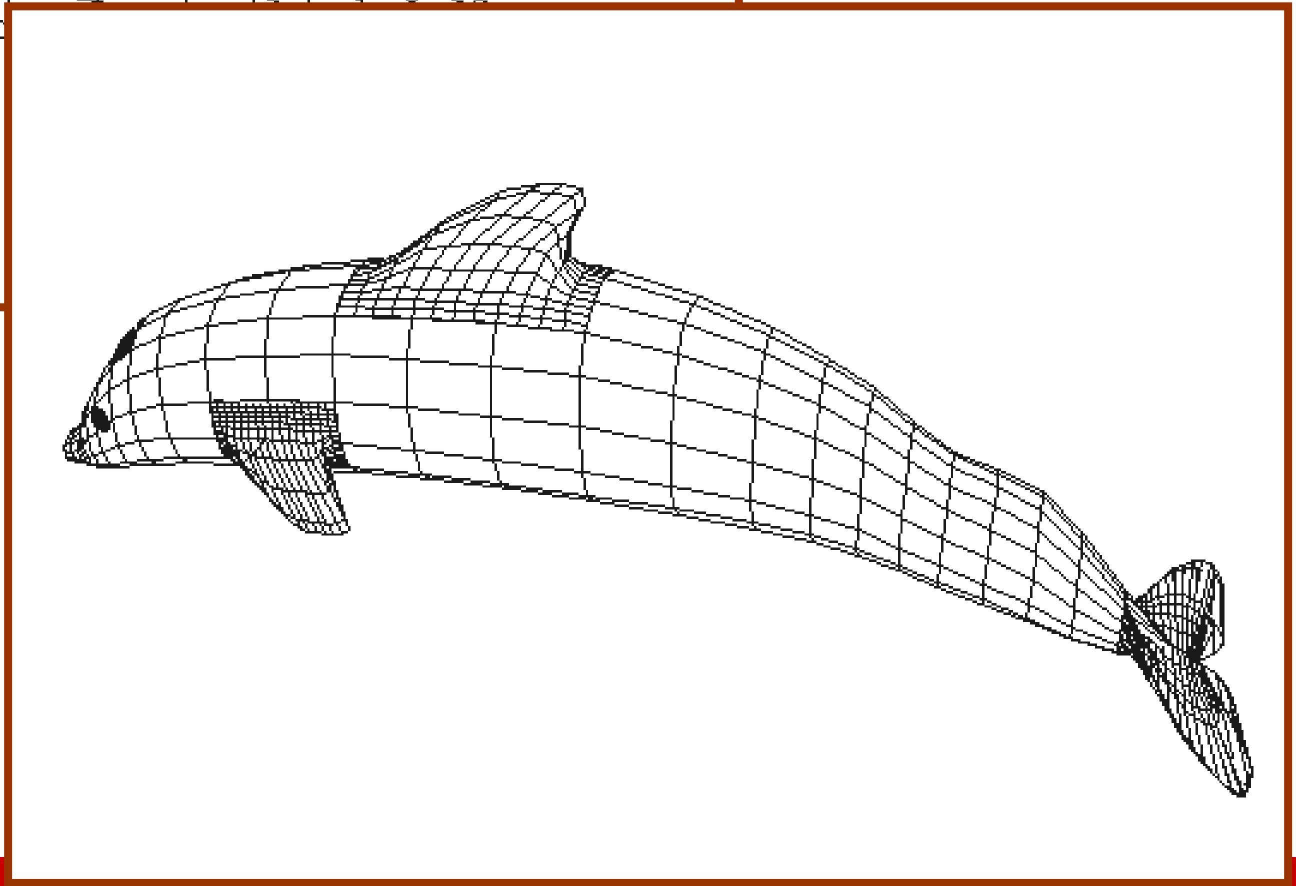
Trimmed NURBS surfaces: hierarchical modelling

- Using Trimmed NURBS
  - Restrict the domain into regions of interest
  - The original surface is unchanged
  - Construct local details on the trimming regions
- Locally modify the surface adding geometric details without any parametric changes.

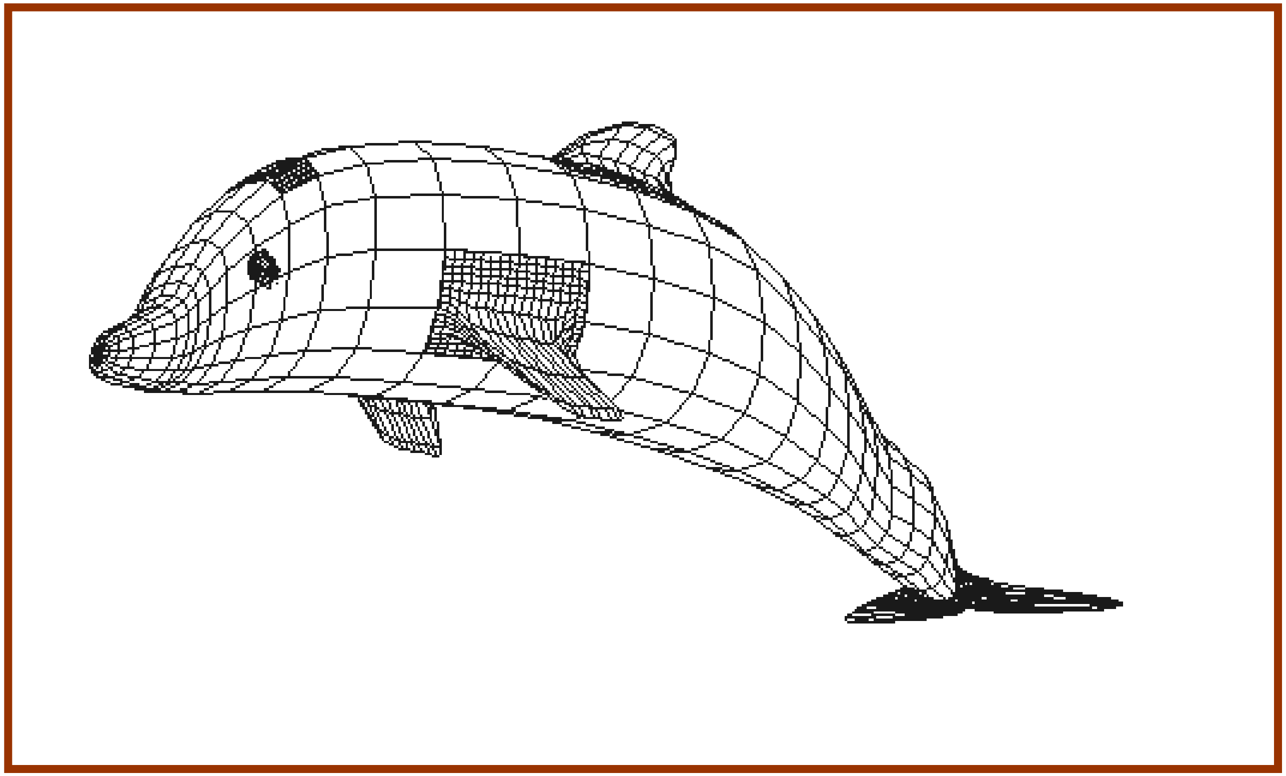


Hierarchical modelling





Hierarchical modelling



Hierarchical modeling

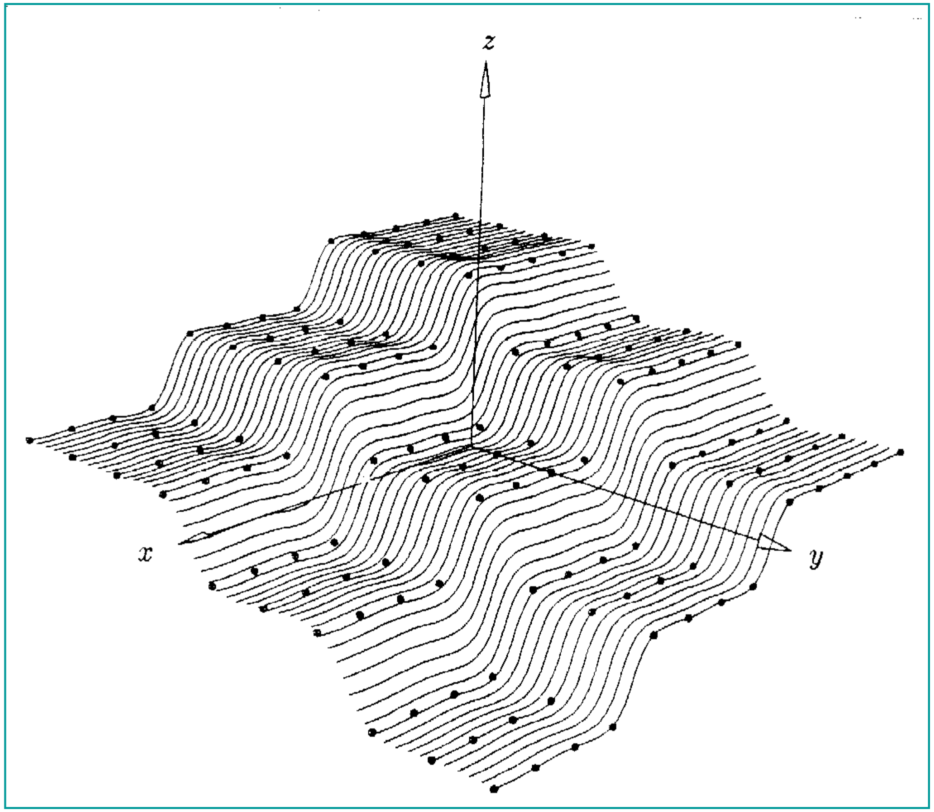
Courtesy of CGGroup , xcmodel , Univ. Bologna

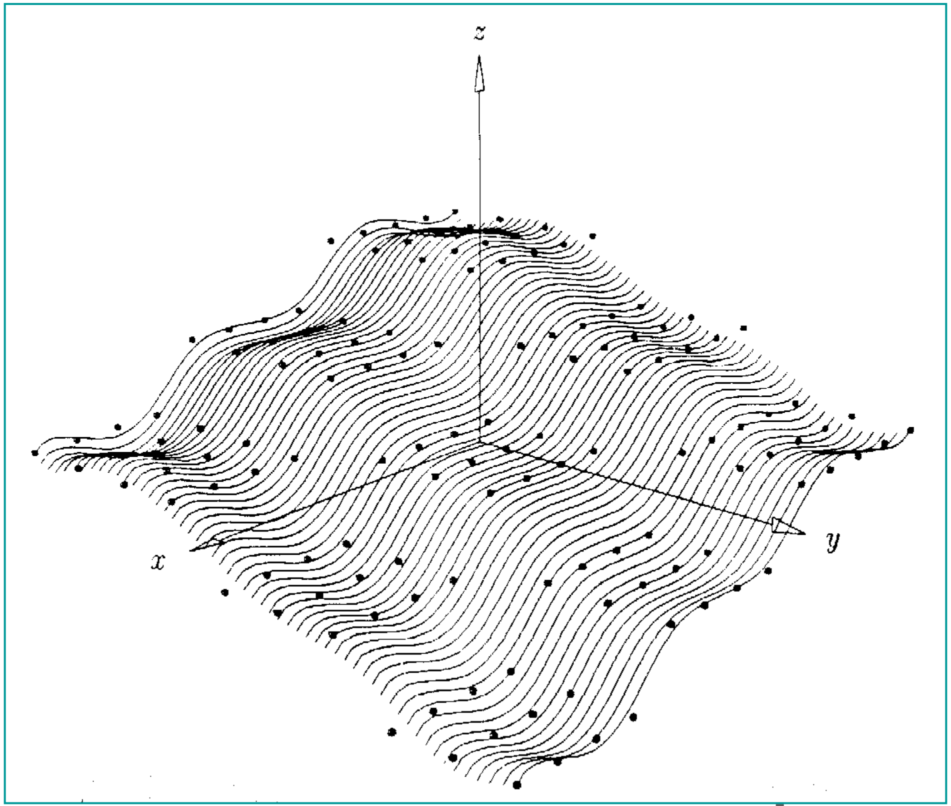
Geometric Modelling of surfaces

- Modelling/ design methods:
  - Cross-section design
  - Interactive design by manipulating the control mesh
  - Creating a patch net or mesh from a set of 3D points representing a real object (surface interpolation/surface fitting)

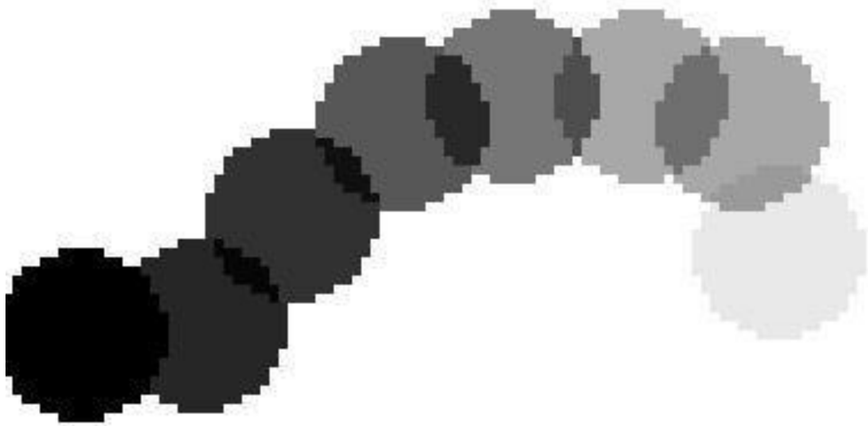
Surface fitting

interpolant **approximant**

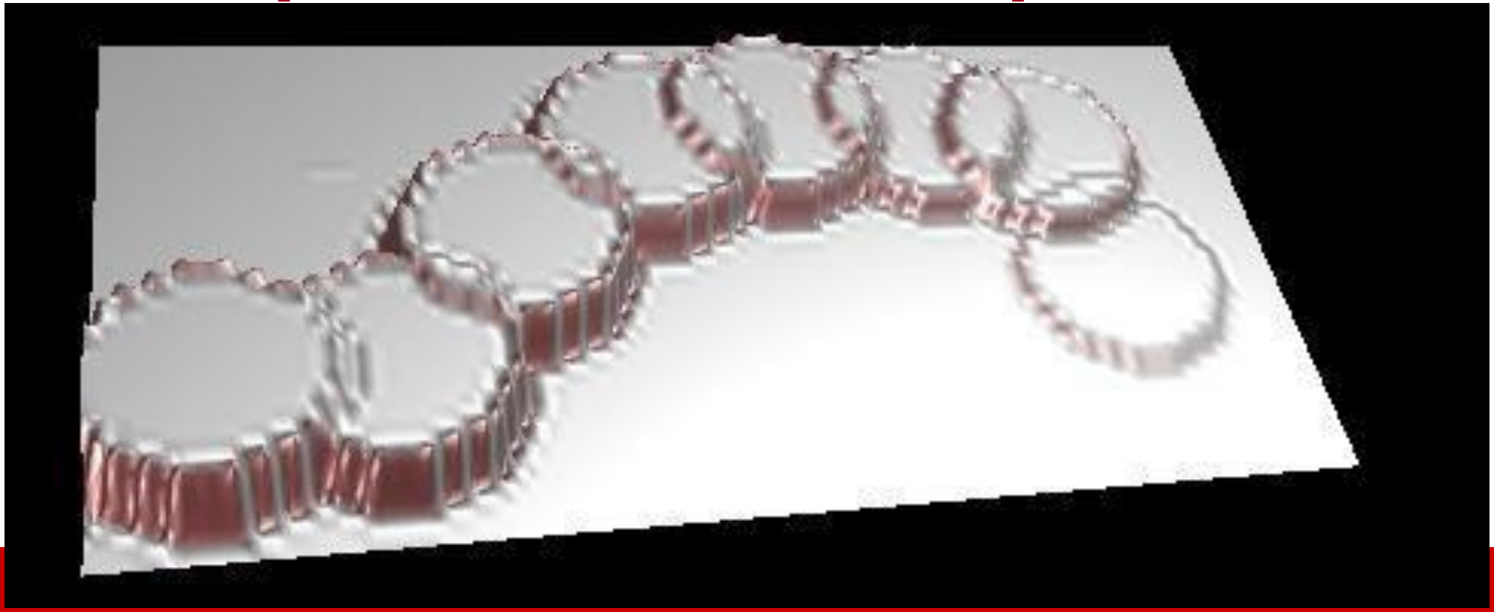




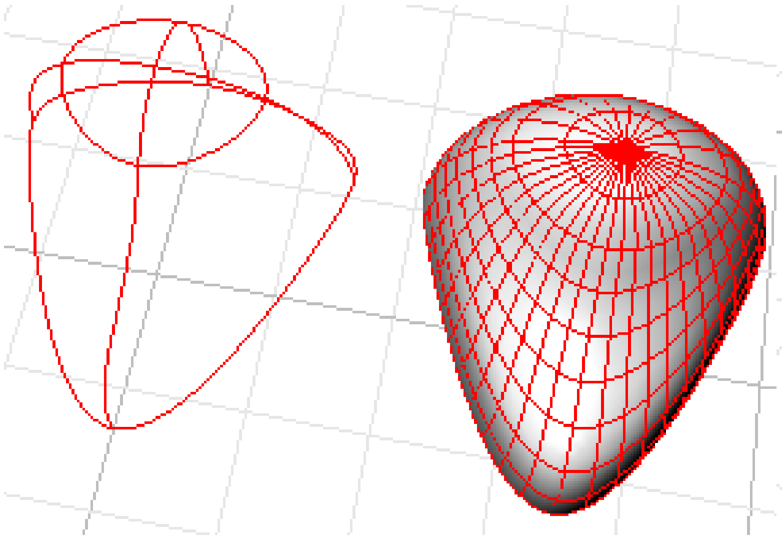
From a gray scale image

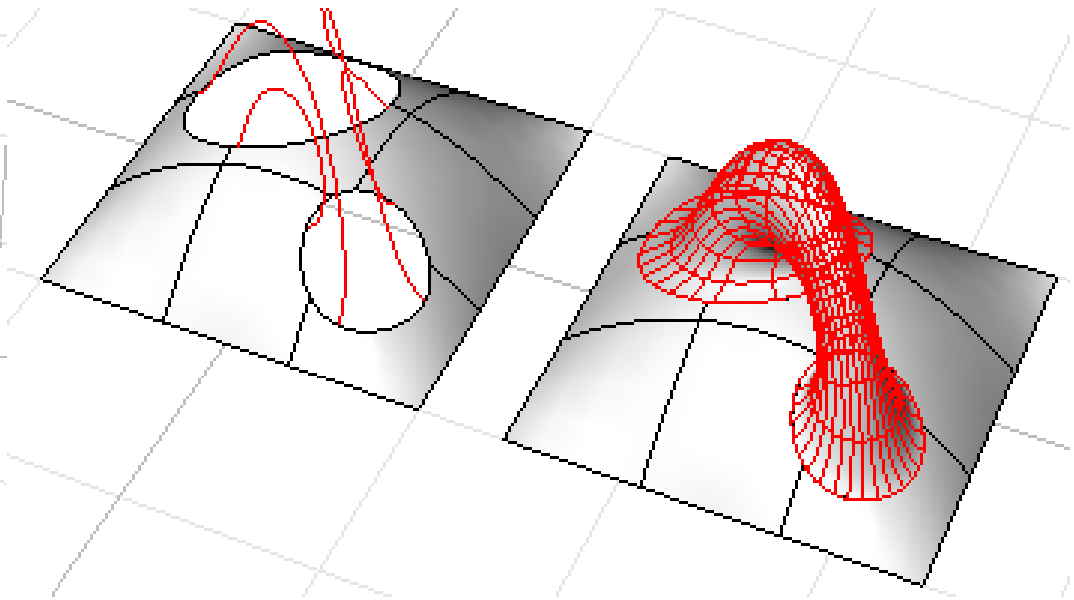


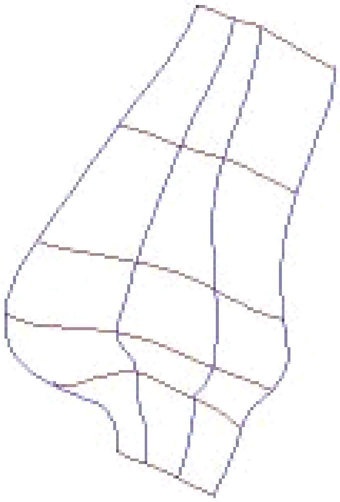
.. **to a spline surface interpolant**

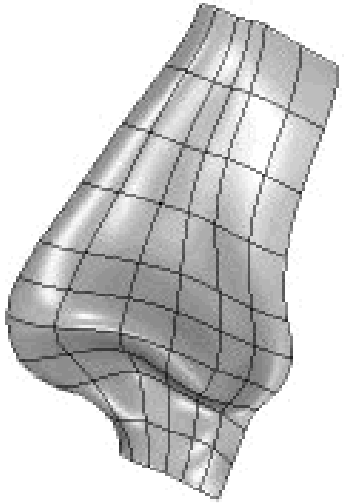


Spline Surface by interpolating a network of curve









Cross sectional design: from curves to surfaces

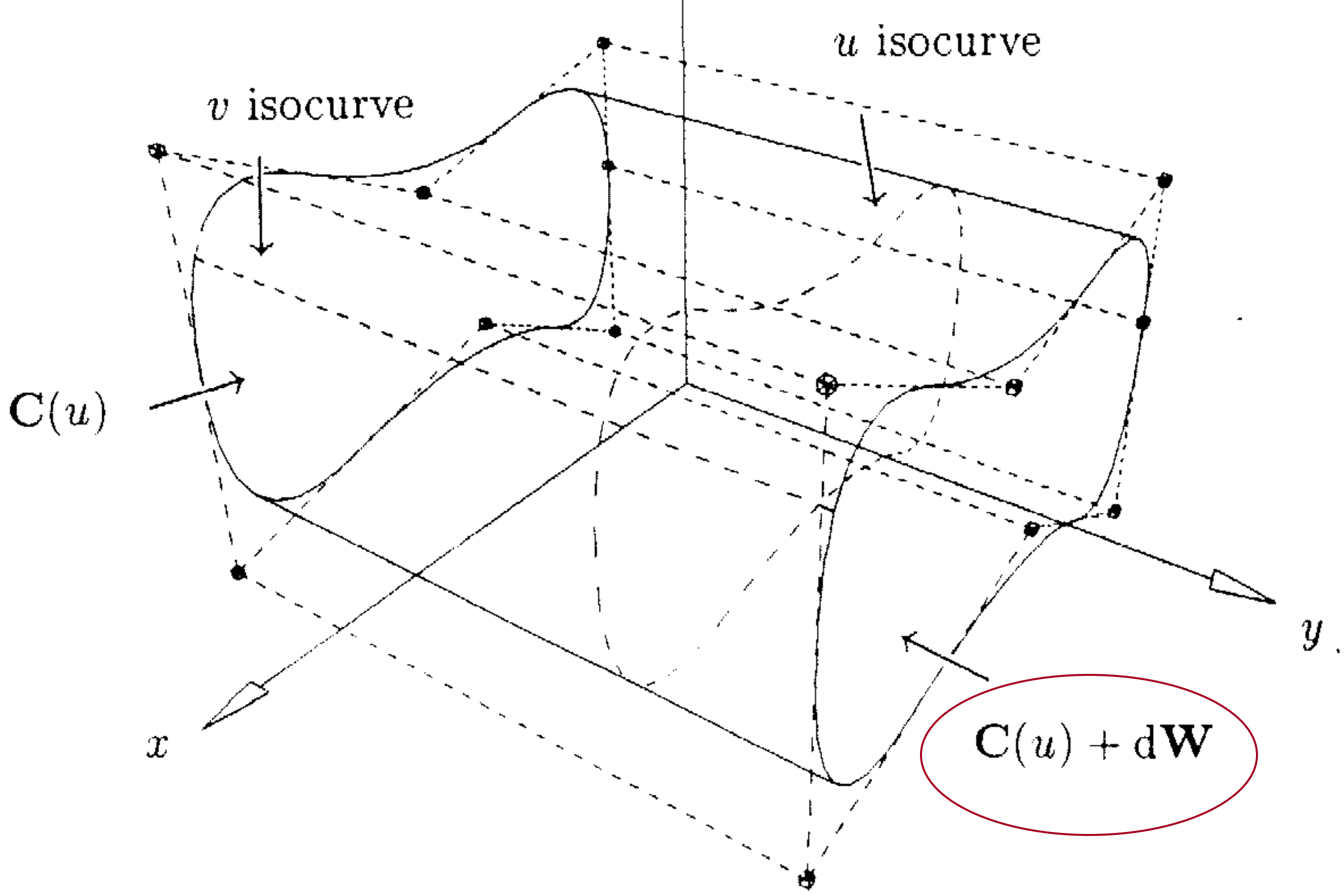
Model the shape of a surface by modifing its 3D CP in a 2D window is a difficult task,

We need tools to construct surfaces from curves automatically.

- **Extruded Surface**
- **Ruled Surface**
- Surfaces of Revolution
- **Skinned Surface**
- **Swept Surface**
- **....**

Extruded Surface

Obtained by moving a profile curve c(u) in a given direction W for a a given distance d



Extruded Surface

$$c(u) = \sum_{i=1}^{n+K} P_i N_{i,n}(u) \quad \text{knot vector } U$$

- Extrusion direction: W (unitary vector in v direction)
- Extrusion offset: d

- For fixed $\underline{u}$, $s(\underline{u},v)$ is a straight segment from $c(\underline{u})$ to $c(\underline{u})+dW$
- For fixed $\underline{v}$, $s(u,\underline{v})$ is the curve

$$s(u,\underline{v}) = c(u) + \underline{v}dW = \sum_{i=1}^{n+K} (P_i + \underline{v}dW) N_{i,n}(u)$$

Extruded Surface

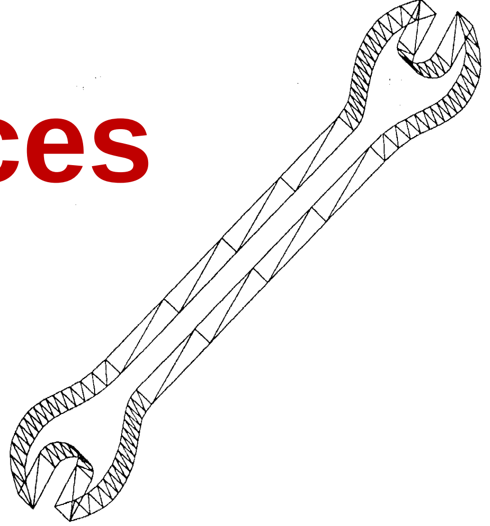
Extruded surface :

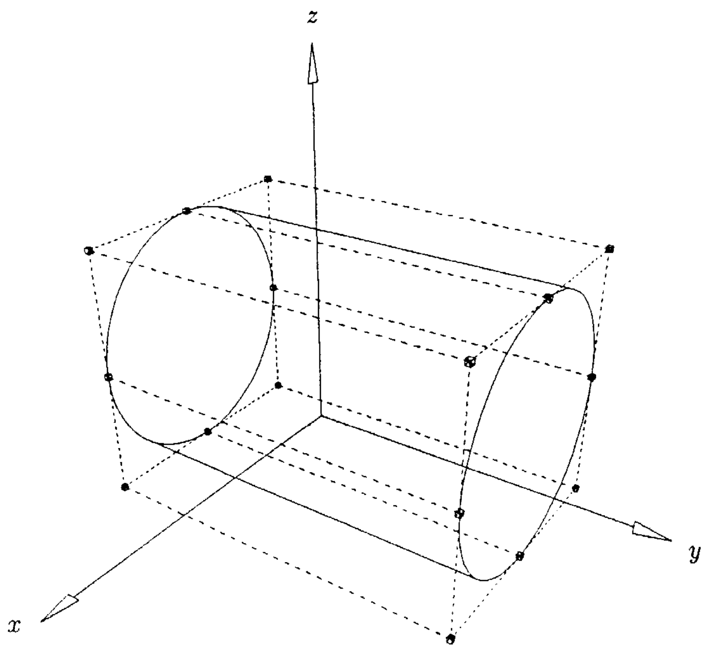
$$s(u,v) = \sum_{i=1}^{n+K} \sum_{j=1}^{2} P_{ij} N_{i,n}(u) N_{j,2}(v)$$

- Control points $\mathbf{P_{i1}=P_i;}$ $\mathbf{P_{i2}=P_i+dW}$;
- Knot vectors: UxV, **V=[0,0,1,1]**

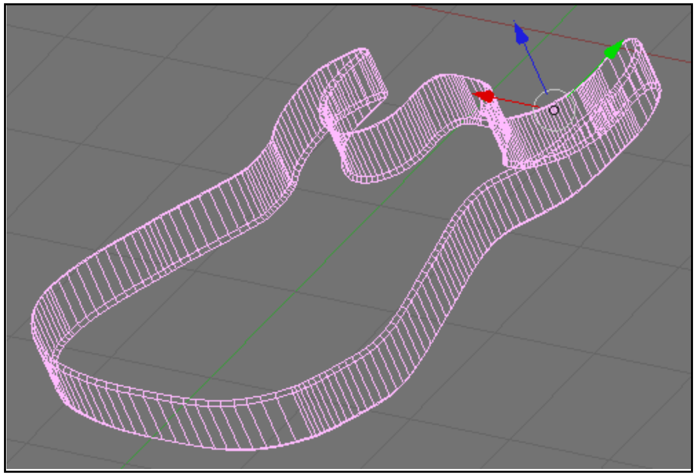
- If c(u) is rational (NURBS) with weights w_i, then s(u,v) is rational with weights $\mathbf{w_{i1}=w_{i2}=w_i}$

Extruded Surfaces





$$s(u,v)=\sum_{i=1}^{9}\sum_{j=1}^{2}P_{ij}N_{i,3}(u)N_{j,2}(v)$$



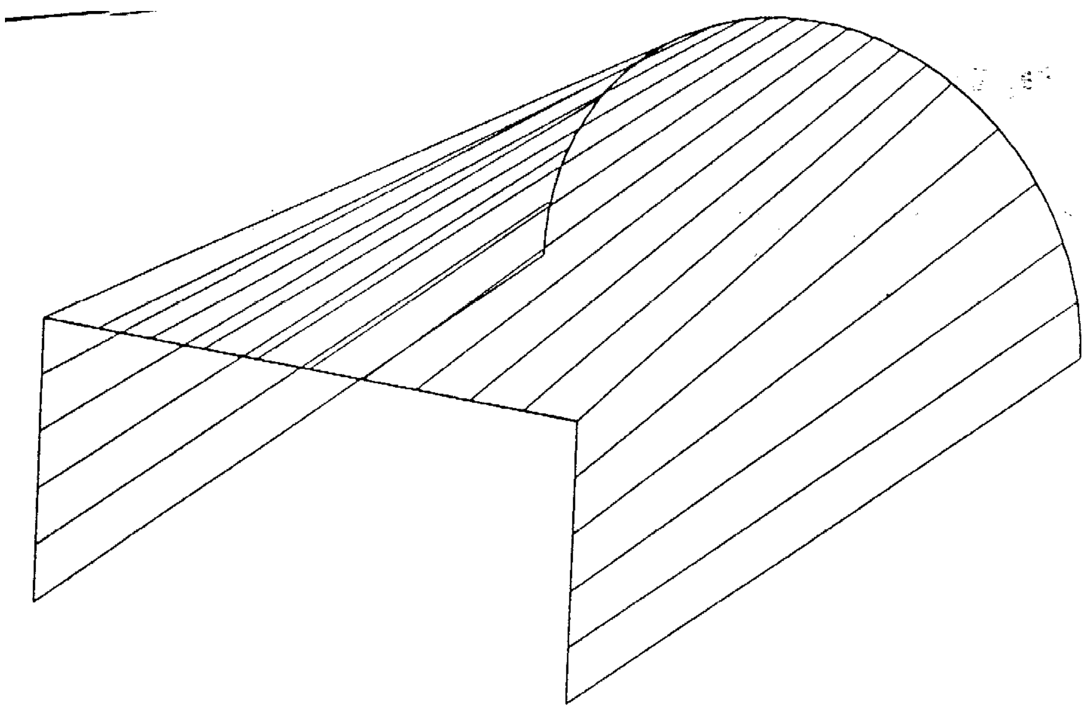
The cylinder is obtained by traslating the NURBS circle (9 points) a distance d along a vector normal to the plane of the circle.

Ruled Surface

Obtained by linear interpolation in v direction between curves **$c_1(u)$** and **$c_2(u)$** defined on U parametric domain.
For fixed $\underline{u}$, s($\underline{u}$,v) is a straight segment joining c_1 and c_2

$$c_1(u) = \sum_{i=1}^{n+K} P_i N_{i,n}(u)$$

$$c_2(u) = \sum_{j=1}^{n+K} T_j N_{j,n}(u)$$



Same degree n-1
Same knot vector U

Ruled Surface

The spline ruled surface on the parametric domain UxV, V=[0,0,1,1] is defined as :

$$s(u,v) = \sum_{i=1}^{n+K} \sum_{j=1}^{2} w_{ij} P_{ij} N_{i,n}(u) N_{j,2}(v)$$

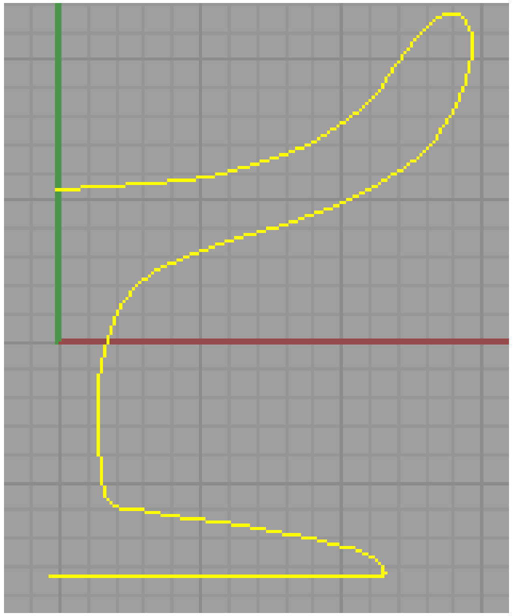
$$p_{i1} = P_i; \quad p_{i2} = T_i;$$

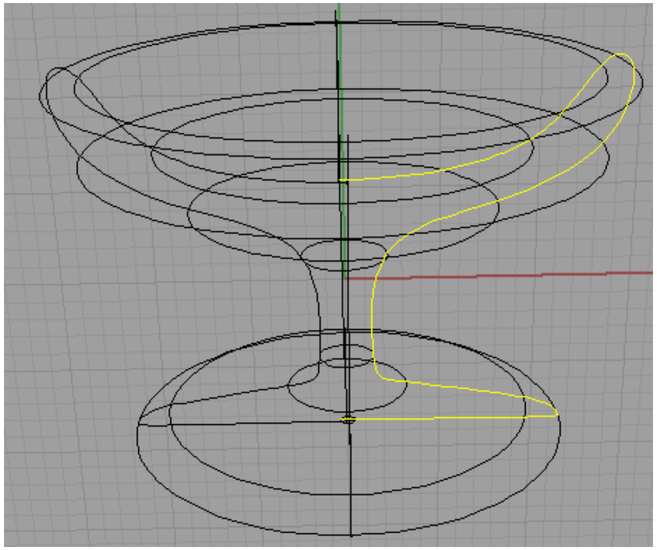
If the two curves are NURBS, then the ruled surface is rational as well, with weights

$$w_{i1} = w_i^{[1]}; \quad w_{i2} = w_i^{[2]};$$

Surfaces of revolution

Given a profile curve **c**(*t*) in the plane, the surface is defined by spinnig it through an arbitrary angle around an axis





Surfaces of revolution

Profile curve in the x-z plane (revolve it about the z axis):

$$c(v) = \sum_{j=1}^{m+K} P_j N_{j,m}(v) \quad Knot\ vector\ V$$

$$P_j = (x_j, 0, z_j)^T$$ **Control points**

For fixed u=u0, s(u0,v) is the isoparametric curve c(v) rotated by a given angle around the z axis

For fixed v=v0, s(u,v0) is a circle in x-y plane, with its center on the z axis

Surfaces of revolution

NURBS surface of revolution s(u,v):

$$s(u,v)=\sum_{i=1}^{9}\sum_{j=1}^{m+K}P_{ij}R_{i,3}(u)R_{j,m}(v)$$

$$U=\{0,0,0,1/4,1/4,1/2,1/2,3/4,3/4,1,1,1\}$$

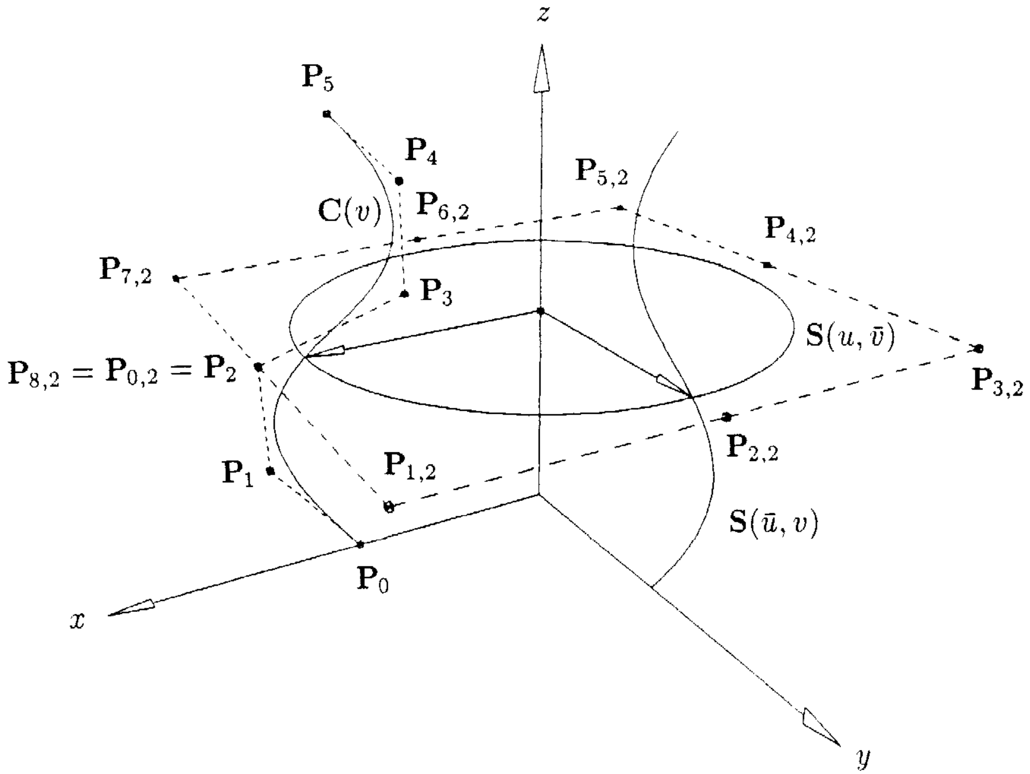
Knot vector for a 9-points circle

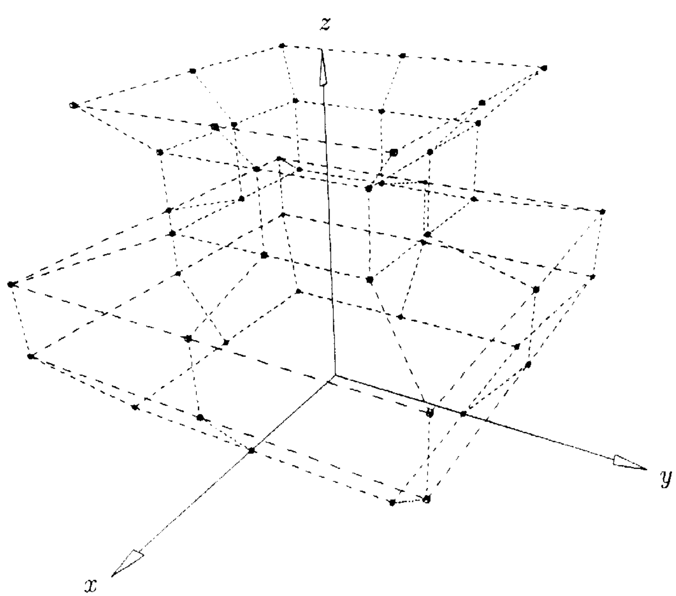
$$P_{ij}=\begin{cases} P_j \quad i=1 \\ \text{rotation } P_j \text{ of } 45° \quad i=2,3,..,9 \\ \text{(for fixed j CP lie on the } z=zj \text{ plane)} \end{cases}$$

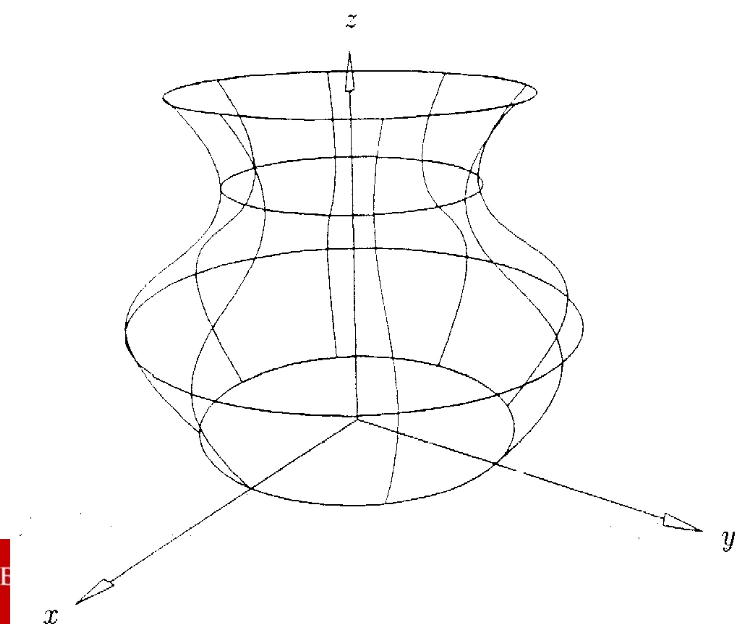
$$w_i=\{1,\sqrt{2}/2,1,\sqrt{2}/2,1,\sqrt{2}/2,1,\sqrt{2}/2,1\}$$

Weights for the NURBS circle curve

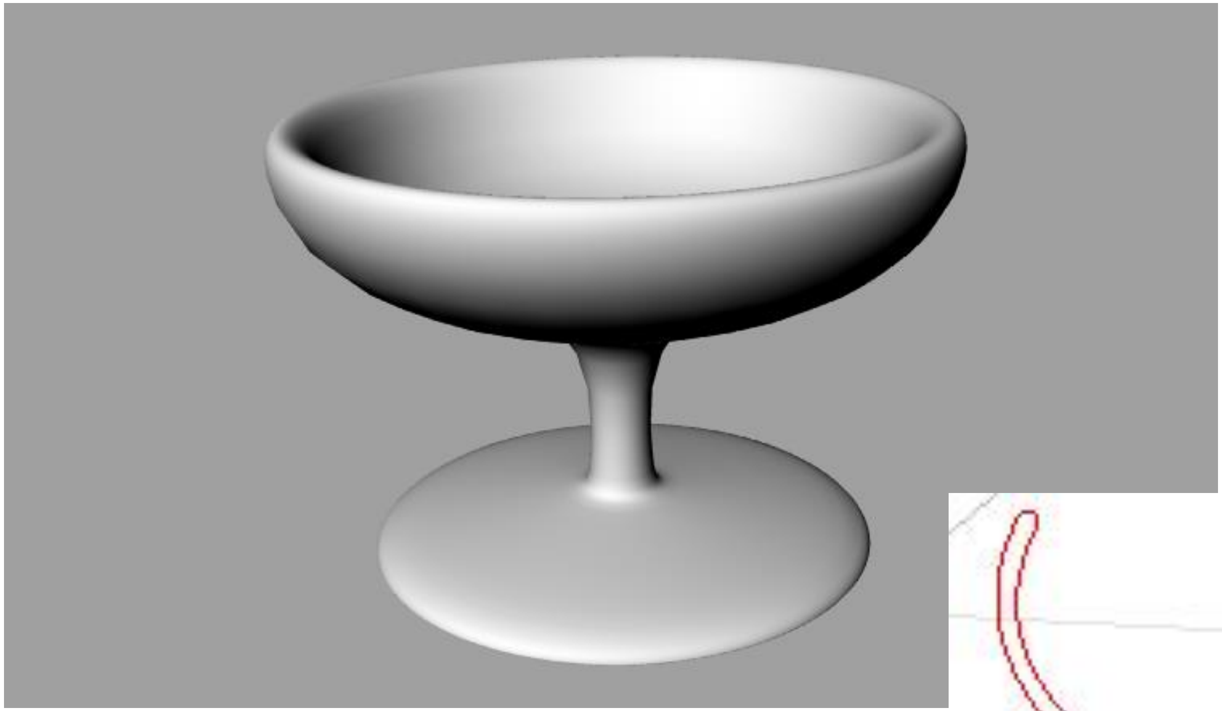
$$w_{ij}=w_i w_j$$

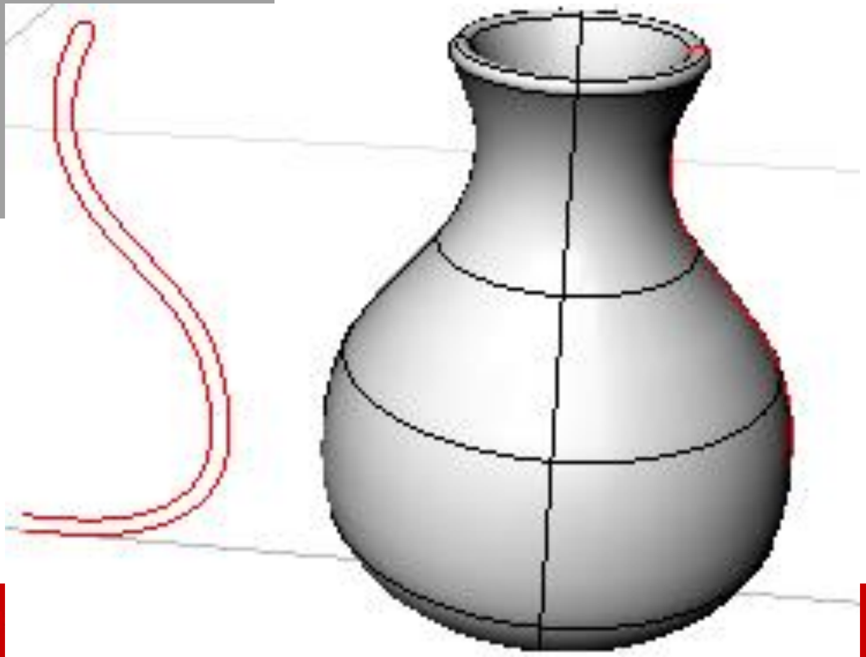

z
$\mathbf{P}_5$
$\mathbf{P}_4$
$\mathbf{P}_{5,2}$
$\mathbf{C}(v)$
$\mathbf{P}_{6,2}$
$\mathbf{P}_{4,2}$
$\mathbf{P}_{7,2}$
$\mathbf{P}_3$
$\mathbf{S}(u, \bar{v})$
$\mathbf{P}_{8,2} = \mathbf{P}_{0,2} = \mathbf{P}_2$
$\mathbf{P}_{3,2}$
$\mathbf{P}_{2,2}$
$\mathbf{P}_1$
$\mathbf{P}_{1,2}$
$\mathbf{S}(\bar{u}, v)$
$\mathbf{P}_0$
x
y


z
y
x


z
y
x

Example: profile curve and Surfaces of revolution



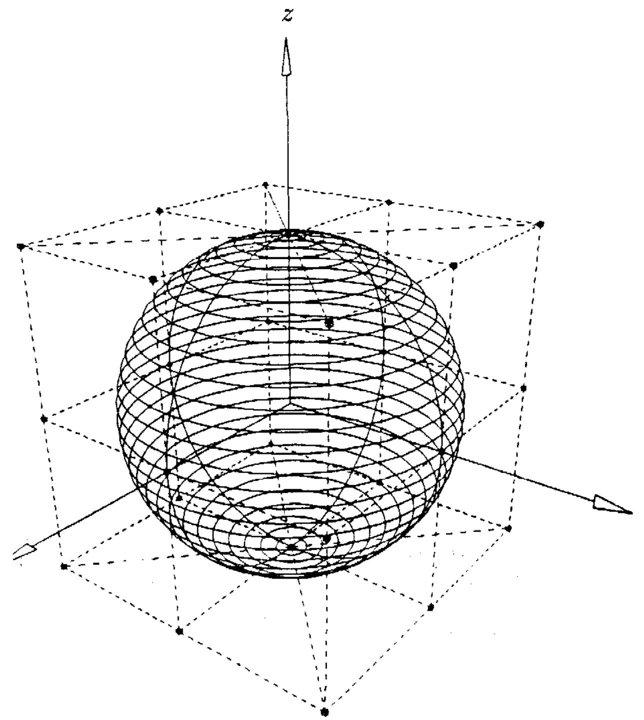


Sphere as a revolution NURBS surface

Revolving about the z-axis a half-circle (unitary ray, centered at the origin) c(u) of order 3

$$c(u) = \frac{\sum_{i=1}^{5} w_i P_i N_{i,3}(u)}{\sum_{j=1}^{5} w_j N_{j,3}(u)}$$

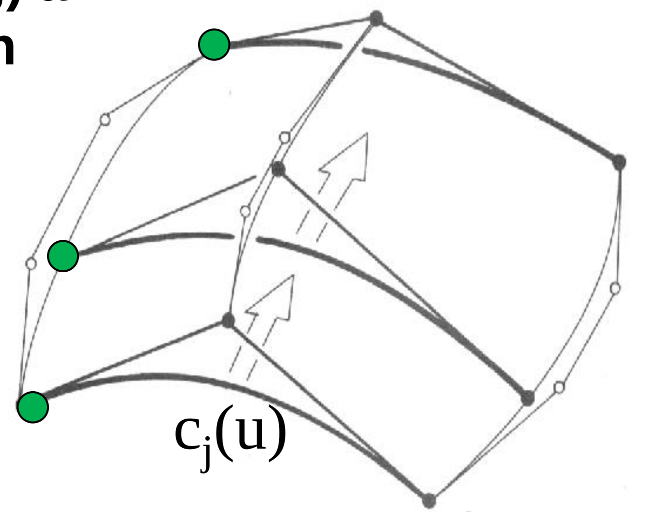
$$U = \left\{0,0,0,\frac{1}{2},\frac{1}{2},1,1,1\right\}, \quad W = \left\{1,\frac{\sqrt{2}}{2},1,\frac{\sqrt{2}}{2},1\right\}$$



CP at the North and South poles are repeated 9 times,

Skinned Surface

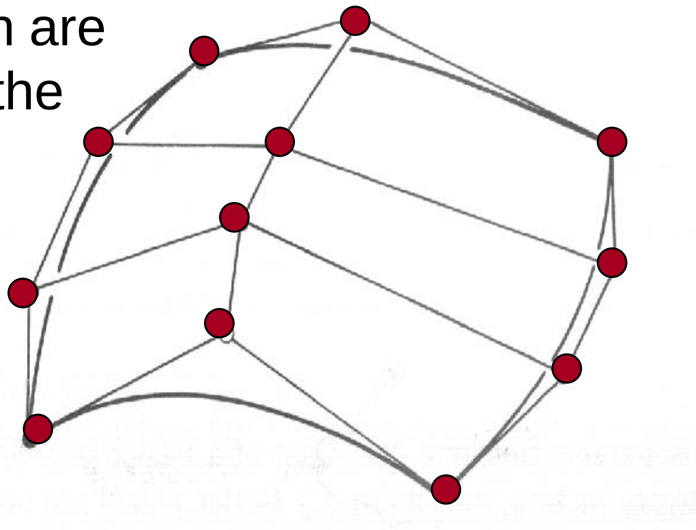
Skinning is the process of interpolating (blending) a given set of NURBS curves (section curves with common degree and number of CP) to form a surface.



Section curves of degree n-1 and knot vector U:

$$c_j^w(u) = \sum_{i=1}^{n+K} Q_{ij}^w N_{i,n}(u), \quad j = 1,..,L+m$$

For each index i, the CP Q_{ij}^w (points ●) in v direction are interpolated in the homogeneous space, obtaining the curves



$$c_i^w(v) = \sum_{j=1}^{m+L} P_{ij}^w N_{j,m}(v), \quad i = 1,..,K+n$$

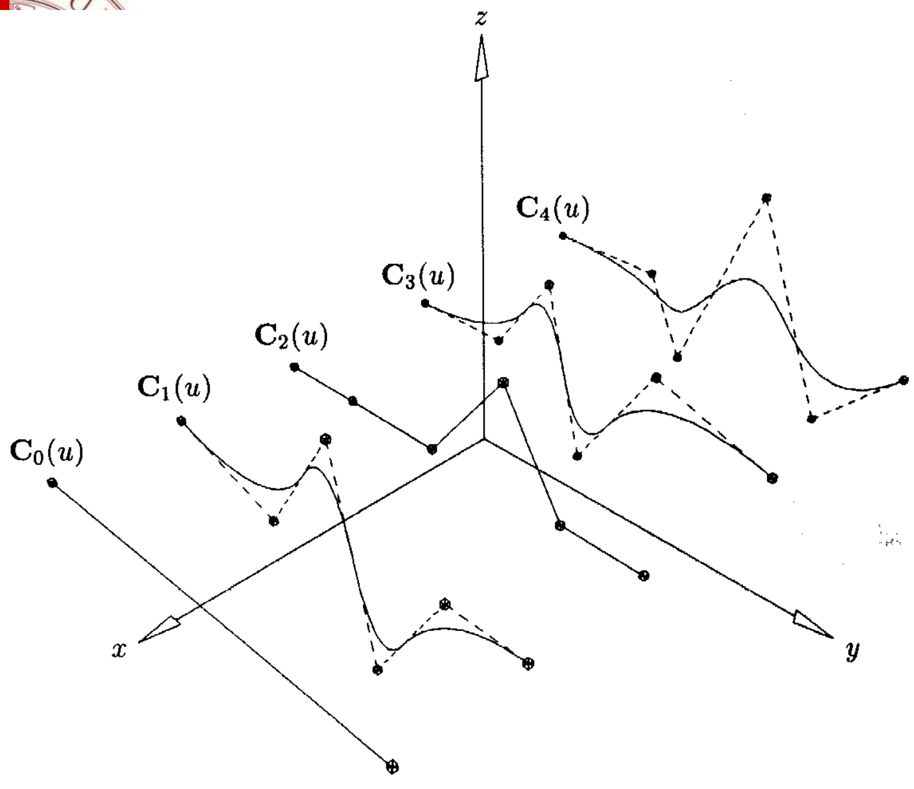
Skinned Surface

The skinned surface is defined by the computed CP P_{ij}^{w}

$$s^{w}(u,v) = \sum_{i=1}^{n+K} \sum_{j=1}^{m+L} P^{w}{}_{ij} N_{i,n}(u) N_{j,m}(v)$$

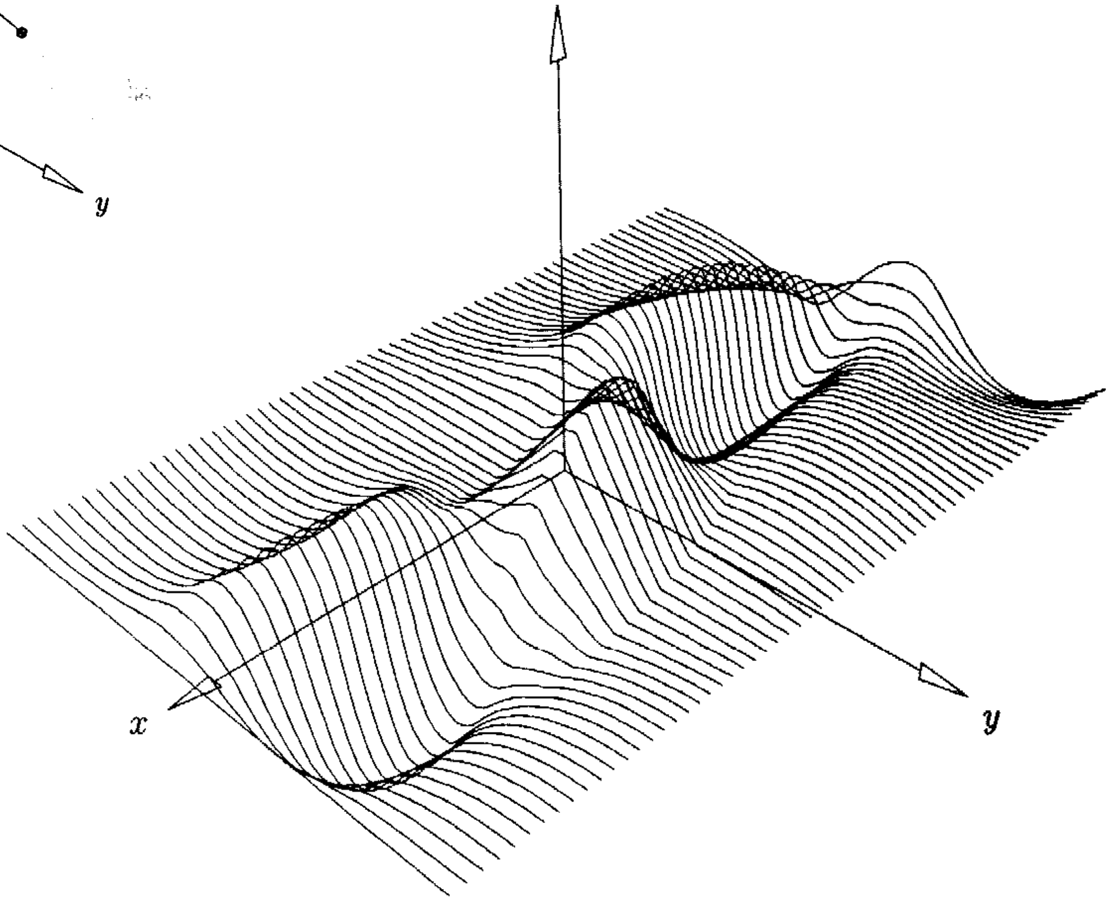
Example

Section curves

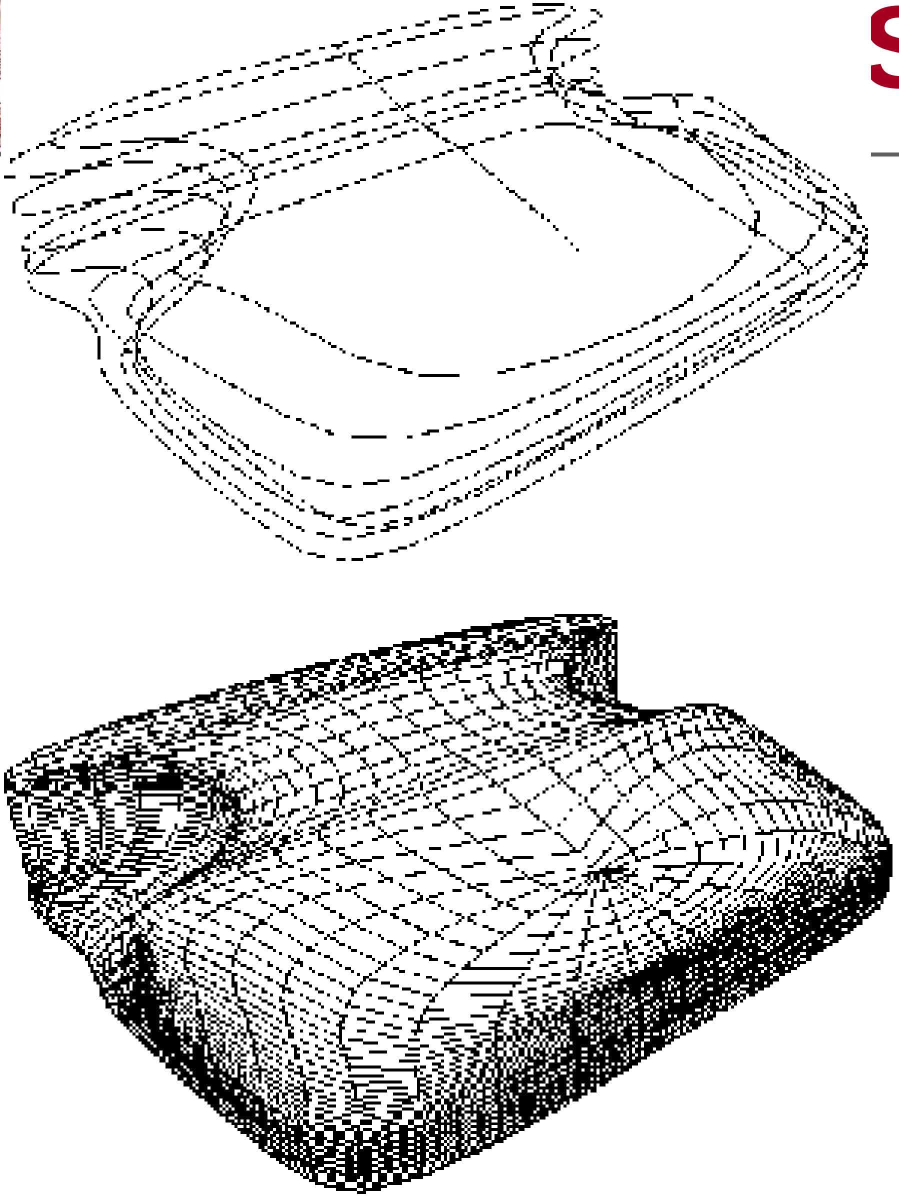


(a)

Skinned surface

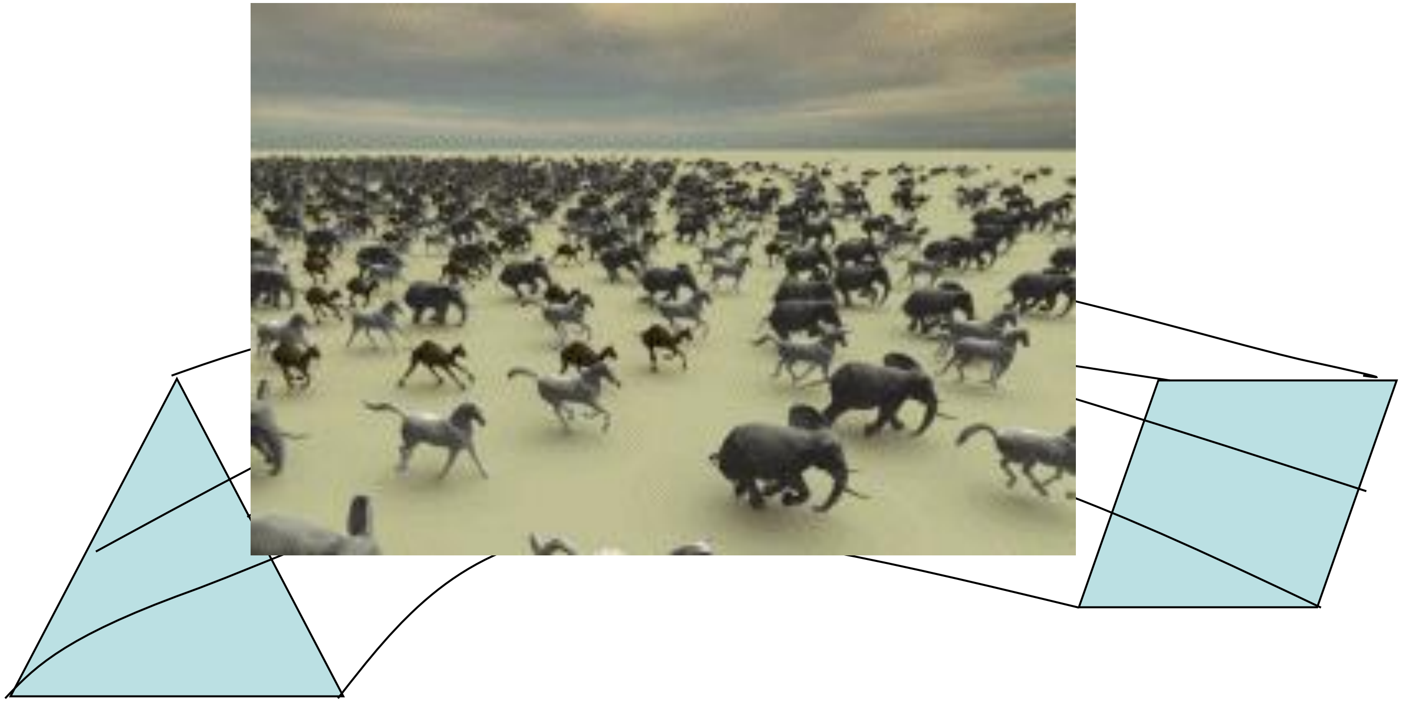


Skinning example



Skinning for animation (morphing)

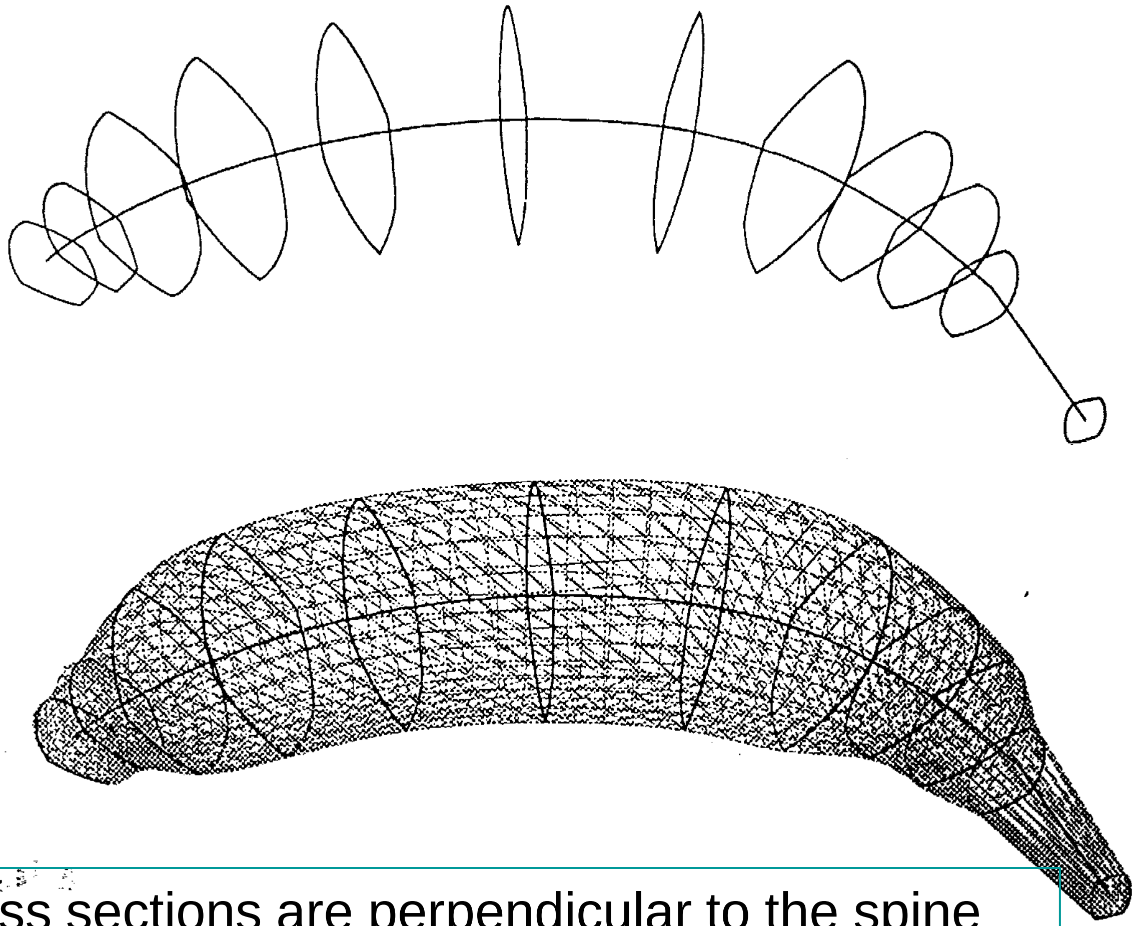
- Do 5), 24(3),
  Au



Swept Surfaces

Surface defined by a *cross sectional curve* moving along a *spine.* Simple version: a single 3D curve for spine and a single 2D curve for the cross section

Sweeping example: several cross sections rather than just one.



The planes containing the cross sections are perpendicular to the spine

Swung surfaces

It's a generalization of a surface of revolution where the trajectory curve is not necessary circular.

Profile Curve P(u) defined in the xz plane:

$$P(u)=\sum_{i=1}^{n+K} P_i R_{i,n}(u), \quad P_i=(P_{x_i},0,P_{z_i})^T$$

Trajectory Curve T(v) defined in the xy plane:

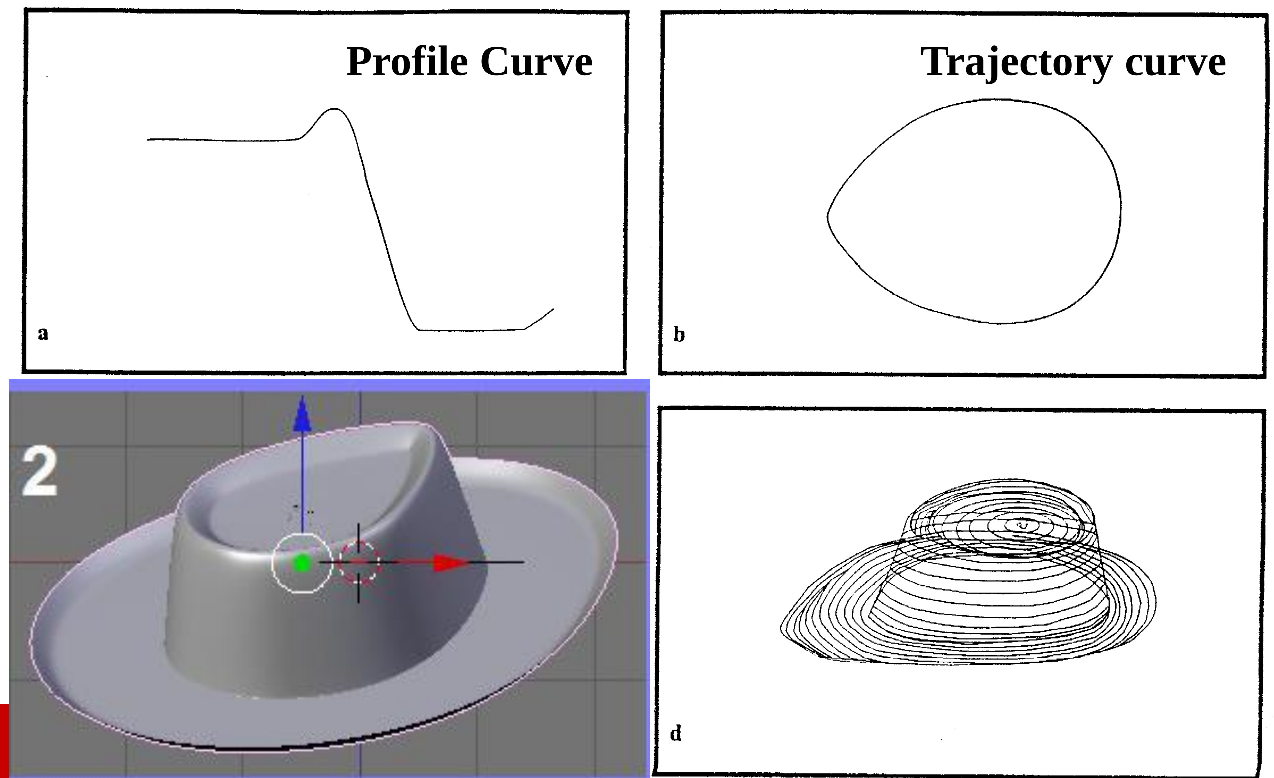
$$T(v)=\sum_{j=1}^{m+L} T_j R_{j,m}(v), \quad T_j=(T_{x_j},T_{y_j},0)^T$$

Trace out surface s(u,v) by moving a profile curve P(u) along a trajectory curve T(v)

Swung surface

Swinging P(u) about the z axis and simultaneously scaling it according to T(v), s is an arbitrary scaling factor. S(u,v) has a NURBS representation given by:

$$s(u,v) = \left(sP_x(u)T_x(v), sP_x(u)T_y(v), P_z(u)\right)^T$$



Swung surface

Fixing u yields curves having the shape of T(v) but scaled in the x and y directions.

Fixing v=$\underline{v}$ the isoparametric curve $C_{\underline{v}}(u)$ are obtained by rotating P(u) into the plane containing the vector ($T_x(\underline{v})$, $T_y(\underline{v})$,0), and scaling the x and y coords. of the rotated curve with the factor s|T(v)|.

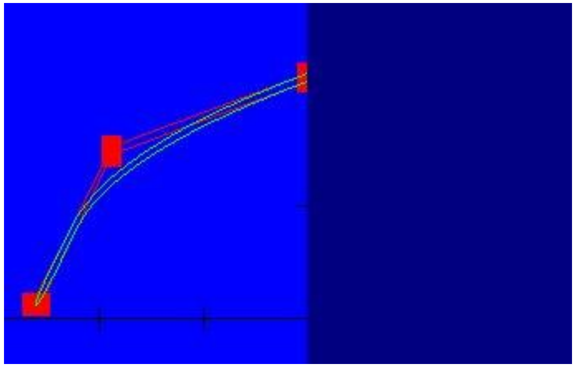
Control points of the swung surface:

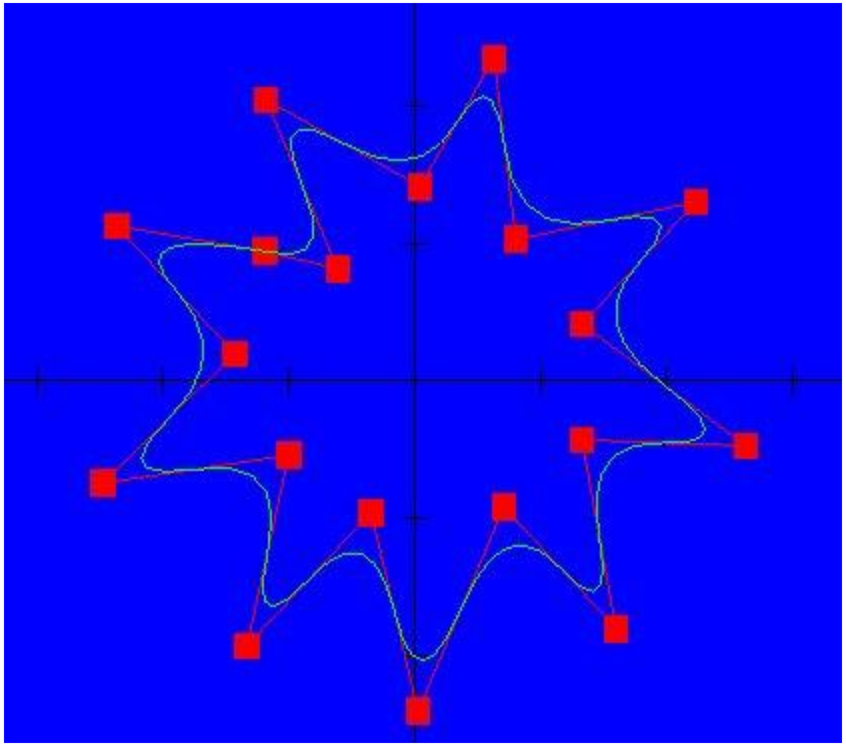
$$Q_{ij} = \left(sP_{x_i} T_{x_j}, sP_{x_i} T_{y_j}, P_{z_i} \right)^T$$

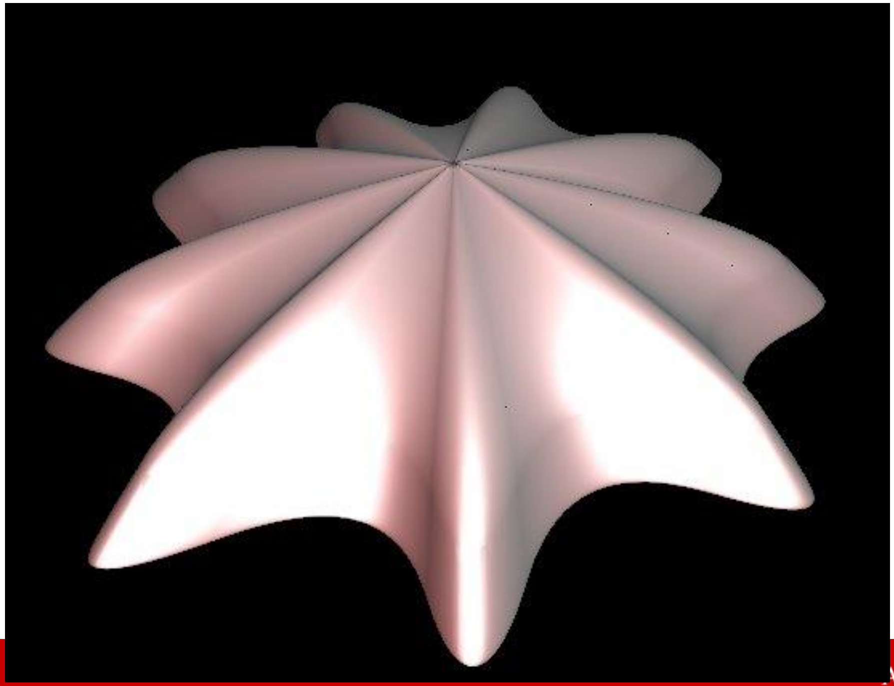
and weights: $w_{ij} = w_i \cdot w_j$

The U and V knot vectors for s(u,v) are those defining P(u) and T(v).

Swinging Example







Tessellation

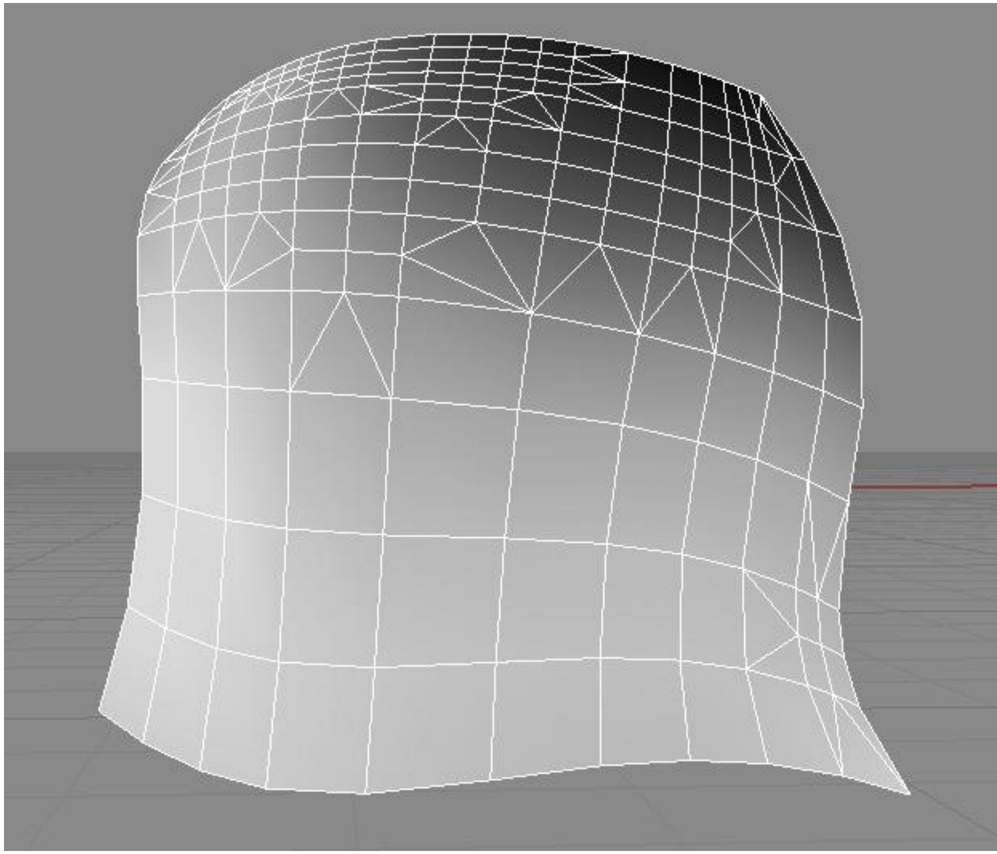
- ***Tessellation*** is the process of taking a complex surface and approximating it with a set of simpler surfaces (like triangles)

- The most straightforward way to tessellate a parametric surface is ***uniform tessellation***
- With this method, we simply choose some resolution in *u* and *v* and uniformly divide up the surface like a grid
- This method is very efficient to compute, as the cost of evaluating the surface reduces to approximately the same cost as evaluating a curve
- However, as the generated mesh is uniform, it may have more triangles than it needs in flatter areas and fewer than it needs in highly curved areas

Adaptive Tessellation

- The goal of a tessellation is to provide the fewest triangles necessary to accurately represent the original surface

- For a curved surface, this means that we want more triangles in areas where the curvature is high, and fewer triangles in areas where the curvature is low

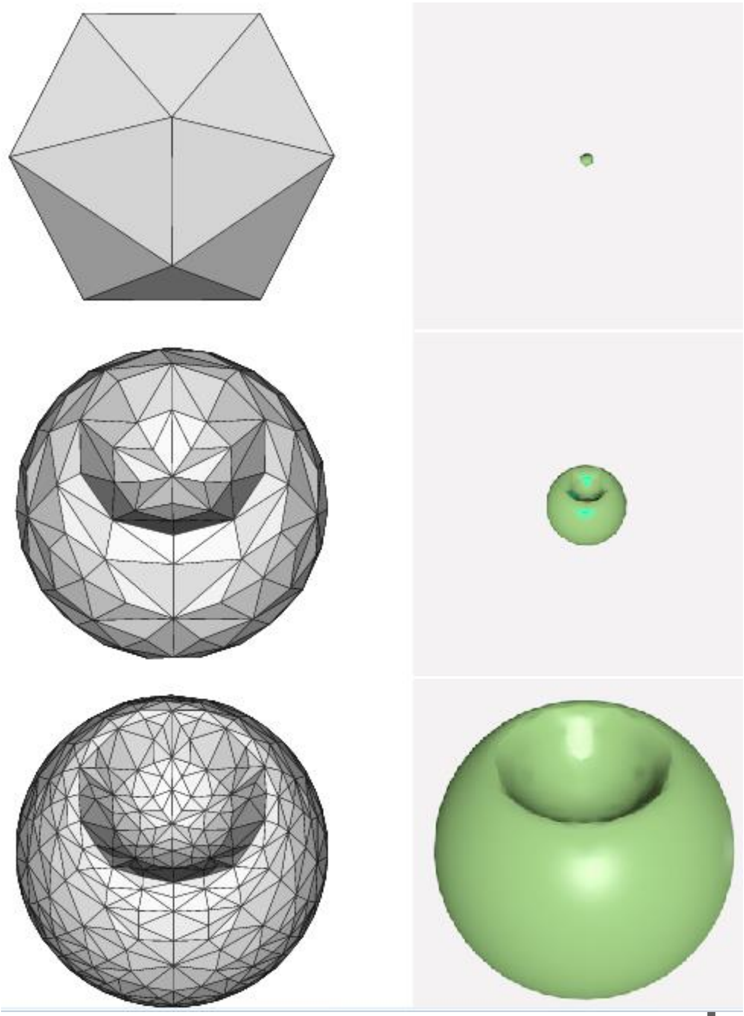


Adaptive Tessellation

- We may also want more triangles in areas that are closer to the camera, and fewer farther away

  Level Of Details

- *Adaptive tessellation* schemes are designed to address these requirements



Draw a Bézier patch: adaptive subdivision method

– basic approach: recursively test flatness
  - if the patch s(u,v) is not flat enough,
  - subdivide into four using curve subdivision twice in v=1/2 and u=1/2, and
  - recursively process each subpatch

– as with curves, convex hull property is useful for termination testing (is inherited from the curves)

Flat test: (convex hull flatness test)

Construct a plane interpolating 3 noncollinear CP

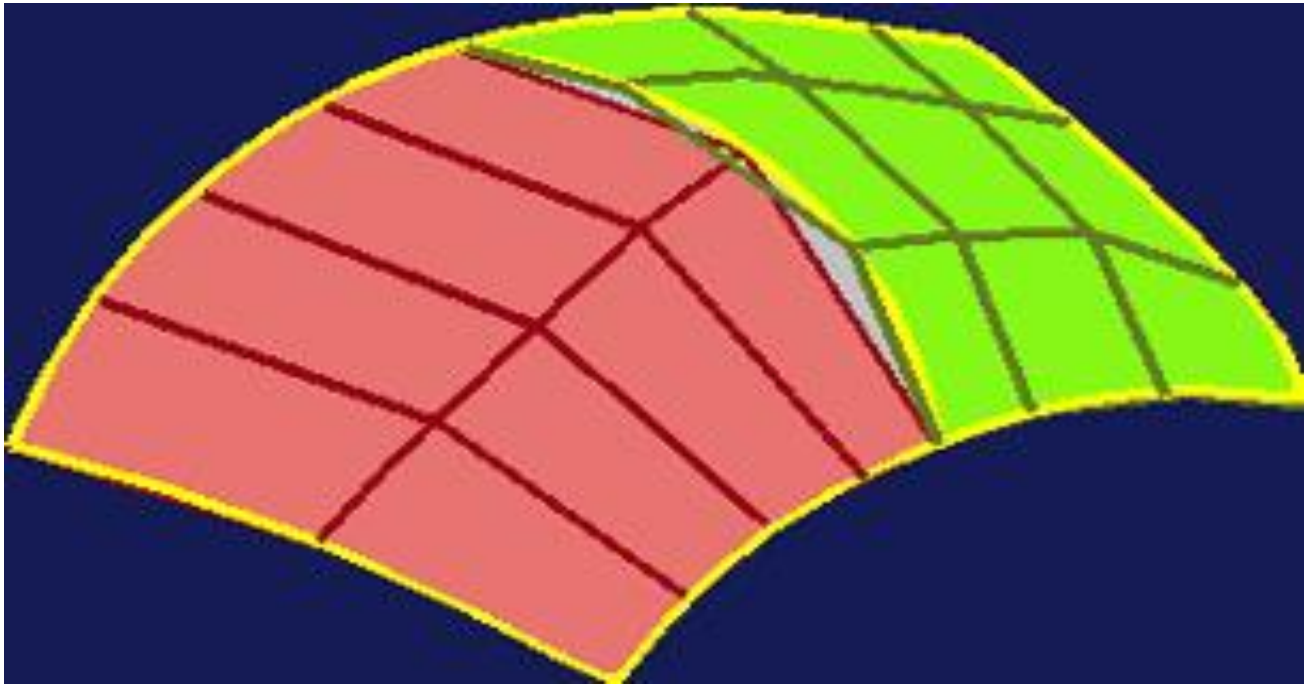
Compute the distances d_i from the remaining CP from this plane. $D = max\ |d_i|$

If (D < Tolerance Tol) then the patch is considered flat and is approximated as a flat quadrilateral.

Crack Problem

With adaptive subdivision, must take care with cracks

- at the boundaries between degrees of subdivision

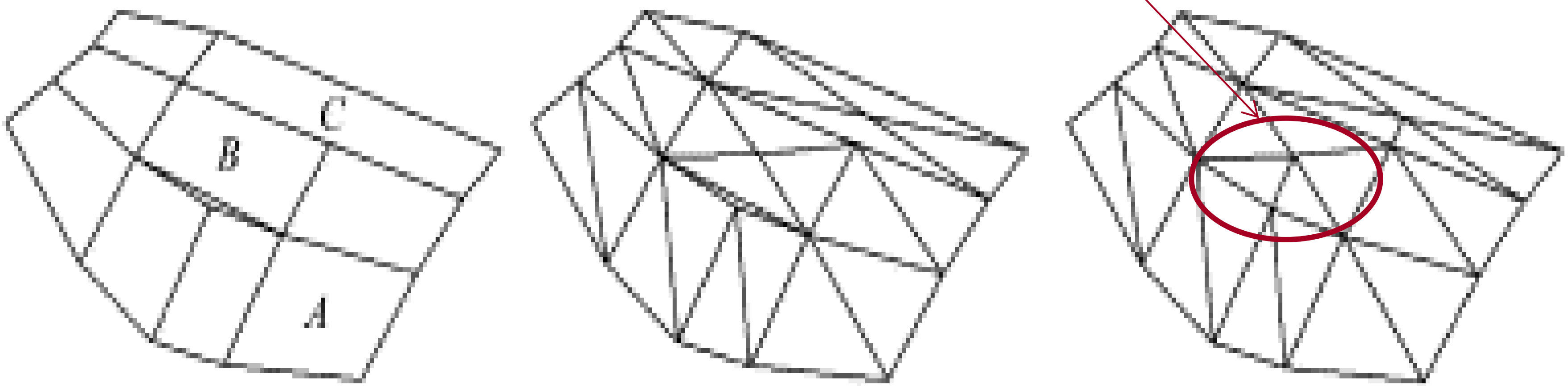


A surface is subdivided and its neighbor is not :
small gaps or small overlaps can appear in the surface

Crack Problem and solutions

Solution:

Replace the patch B with two coplanar patches to allow the common boundary to have the same points and normals



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Serena Morigi

Dipartimento di Matematica

serena.morigi@unibo.it

http://www.dm.unibo.it/~morigi