CS 283
Advanced Computer Graphics

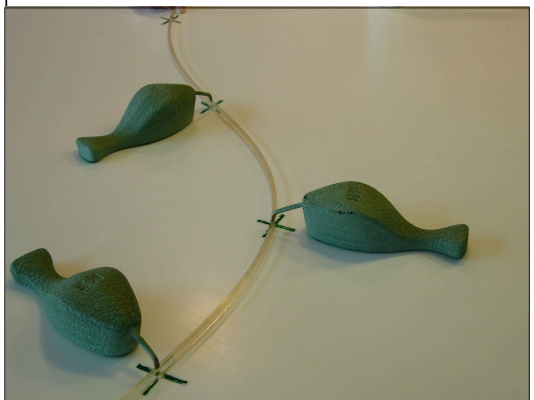
Subdivision Basics

James F. O'Brien

Associate Professor
U.C. Berkeley

Natural Splines

- Draw a "smooth" line through several points



A real draftsman's spline.

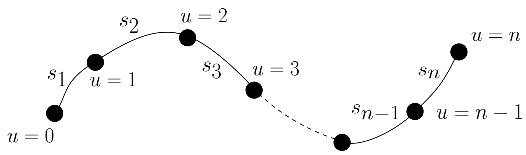
Image from Carl de Boor's webpage.

Natural Cubic Splines

- Given $n+1$ points
  - Generate a curve with n segments
  - Curves passes through points
  - Curve is C^2 continuous

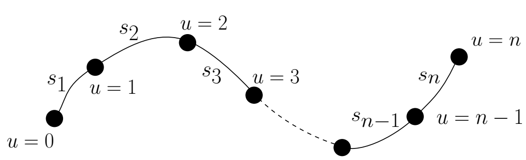
- Use cubics because lower order is better...

Natural Cubic Splines



$$\mathbf{x}(u) = \begin{cases} \mathbf{s}_1(u) & \text{if } 0 \leq u < 1 \\ \mathbf{s}_2(u-1) & \text{if } 1 \leq u < 2 \\ \mathbf{s}_3(u-2) & \text{if } 2 \leq u < 3 \\ & \vdots \\ \mathbf{s}_n(u-(n-1)) & \text{if } n-1 \leq u \leq n \end{cases}$$

Natural Cubic Splines



$s_i(0) = p_{i-1} \quad i = 1 \ldots n$ ← n constraints

$s_i(1) = p_i \quad i = 1 \ldots n$ ← n constraints

$s_i'(1) = s_{i+1}'(0) \quad i = 1 \ldots n-1$ ← n-1 constraints

$s_i''(1) = s_{i+1}''(0) \quad i = 1 \ldots n-1$ ← n-1 constraints

$s_1''(0) = s_n''(1) = 0$ ←2 constraints

Total 4n constraints

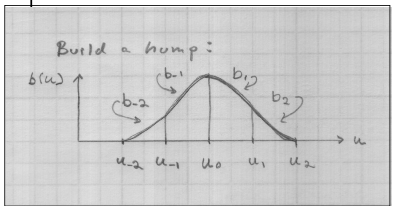
Natural Cubic Splines

- Interpolate data points
- No convex hull property
- Non-local support
  - Consider matrix structure...
- C^2 using cubic polynomials

B-Splines

- Goal: C^2 cubic curves with local support
  - Give up interpolation
  - Get convex hull property
- Build basis by designing "hump" functions

B-Splines



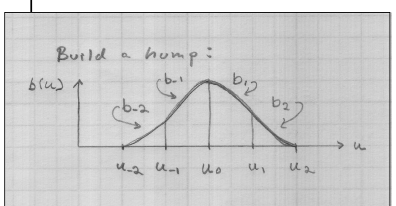
$$\mathbf{b}(u) = \begin{cases} \mathbf{b_{-2}}(u) & \text{if } u_{-2} \le u < u_{-1} \\ \mathbf{b_{-1}}(u) & \text{if } u_{-1} \le u < u_{0} \\ \mathbf{b_{+1}}(u) & \text{if } u_{0} \le u < u_{+1} \\ \mathbf{b_{+2}}(u) & \text{if } u_{+1} \le u \le u_{+2} \end{cases}$$

$b''_{-2}(u_{-2}) = b'_{-2}(u_{-2}) = b_{-2}(u_{-2}) = 0$ ←3 constraints
$b''_{+2}(u_{+2}) = b'_{+2}(u_{+2}) = b_{+2}(u_{+2}) = 0$ ←3 constraints

$b_{-2}(u_{-1}) = b_{-1}(u_{-1})$
$b_{-1}(u_0) = b_{+1}(u_0)$
$b_{+1}(u_{+1}) = b_{+2}(u_{+1})$ ← Repeat for b' and b'' 3×3=9 constraints

Total 15 constraints ...... need one more

B-Splines



$$\mathbf{b}(u) = \begin{cases} \mathbf{b_{-2}}(u) & \text{if } u_{-2} \le u < u_{-1} \\ \mathbf{b_{-1}}(u) & \text{if } u_{-1} \le u < u_{0} \\ \mathbf{b_{+1}}(u) & \text{if } u_{0} \le u < u_{+1} \\ \mathbf{b_{+2}}(u) & \text{if } u_{+1} \le u \le u_{+2} \end{cases}$$

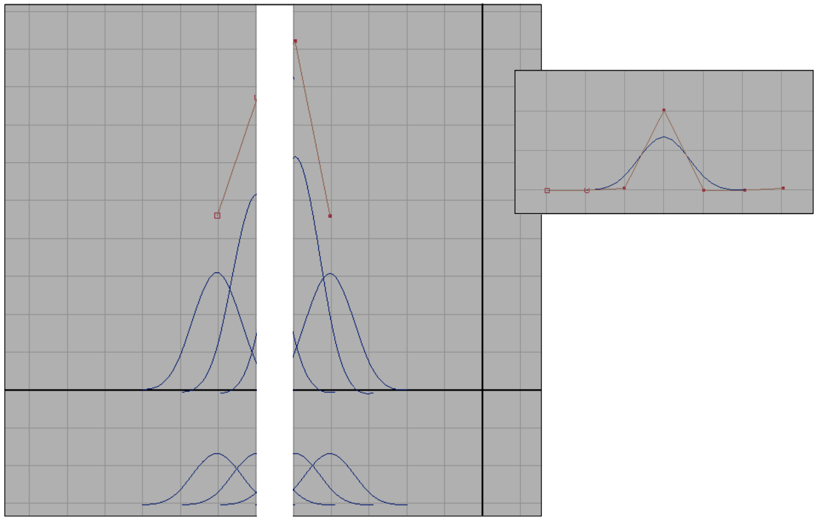
$b''_{-2}(u_{-2}) = b'_{-2}(u_{-2}) = b_{-2}(u_{-2}) = 0$ ←3 constraints
$b''_{+2}(u_{+2}) = b'_{+2}(u_{+2}) = b_{+2}(u_{+2}) = 0$ ←3 constraints

$b_{-2}(u_{-1}) = b_{-1}(u_{-1})$
$b_{-1}(u_0) = b_{+1}(u_0)$
$b_{+1}(u_{+1}) = b_{+2}(u_{+1})$ ← Repeat for b' and b'' 3×3=9 constraints

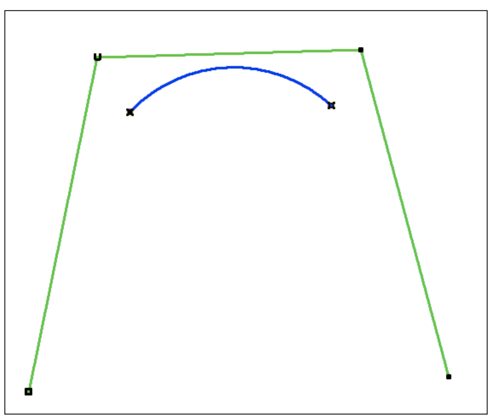
$b_{-2}(u_{-2}) + b_{-1}(u_{-1}) + b_{+1}(u_0) + b_{+2}(u_{+1}) = 1$ ←1 constraint (convex hull)

Total 16 constraints

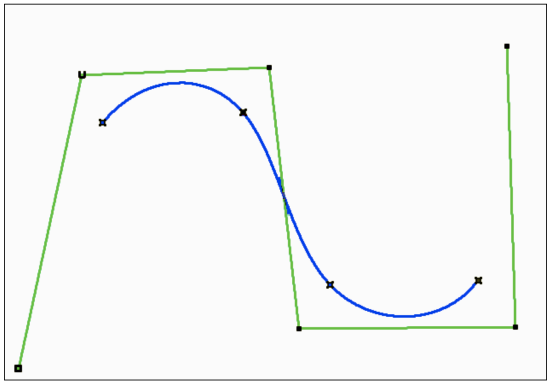
B-Splines



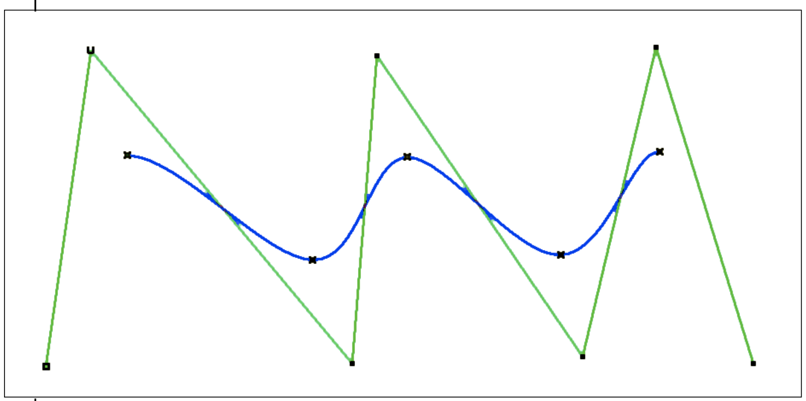
B-Splines



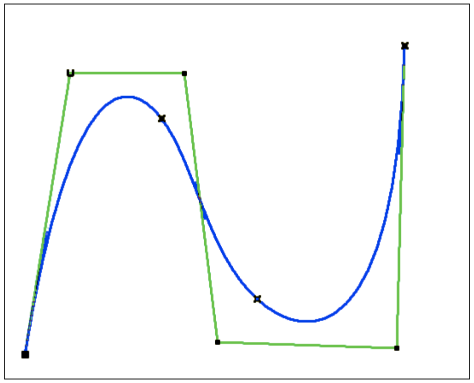
B-Splines



B-Splines



B-Splines



Example with end knots repeated

B-Splines

- Build a curve w/ overlapping bumps
- Continuity
  - Inside bumps C^2
  - Bumps "fade out" with C^2 continuity
- Boundaries
  - Circular
  - Repeat end points
  - Extra end points

B-Splines

- Notation
  - The basis functions are the $b_i(u)$
  - "Hump" functions are the concatenated function
    - Sometimes the humps are called basis... can be confusing
  - The u_i are the knot locations
  - The weights on the hump/basis functions are control points

B-Splines

- Similar construction method can give higher continuity with higher degree polynomials
- Repeating knots drops continuity
  - Limit as knots approach each other
- Still cubics, so conversion to other cubic basis is just a matrix multiplication

B-Splines

- Geometric construction
  - Due to Cox and de Boor
  - My own notation, beware if you compare w/ text
- Let hump centered on u_i be $N_{i,4}(u)$

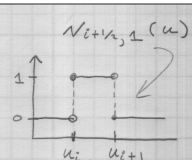
Cubic is order 4

$N_{i,k}(u)$ Is order k hump, centered at u_i

Note: i is integer if k is even
else $(i + 1/2)$ is integer

B-Splines

$$N_{i,1}(u) = \begin{cases} 1 & \text{if } u_{i-1/2} \le u < u_{i+1/2} \\ 0 & \text{else} \end{cases}$$



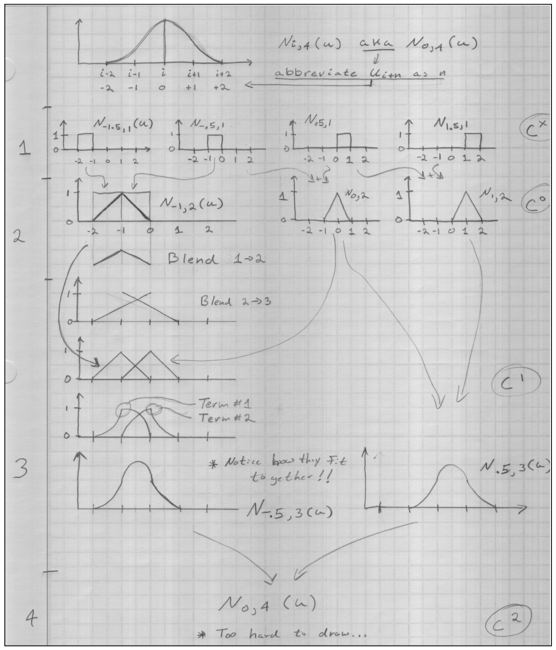
$$N_{i,k}(u) = \frac{(u - u_{i-k/2})\, N_{i-1/2,k-1}(u)}{u_{i+k/2-1} - u_{i-k/2}} + \frac{(u_{i+k/2} - u)\, N_{i+1/2,k-1}(u)}{u_{i+k/2} - u_{i-k/2+1}}$$

$k \geq 2$

"Term #1"

"Term #2"

$\boxed{\text{recursive defn.}}$



NURBS

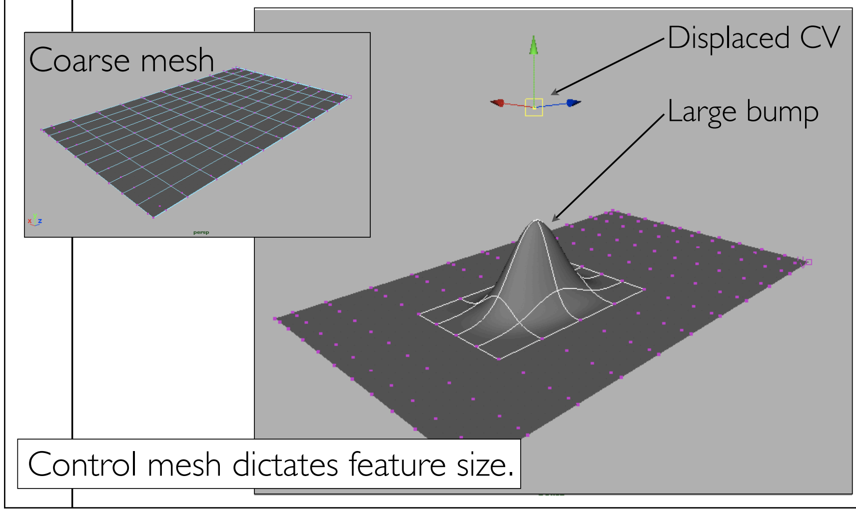
- **N**on**u**niform **R**ational **B**-**S**plines
  - Basically B-Splines using homogeneous coordinates
  - Transform under perspective projection
  - A bit of extra control

NURBS

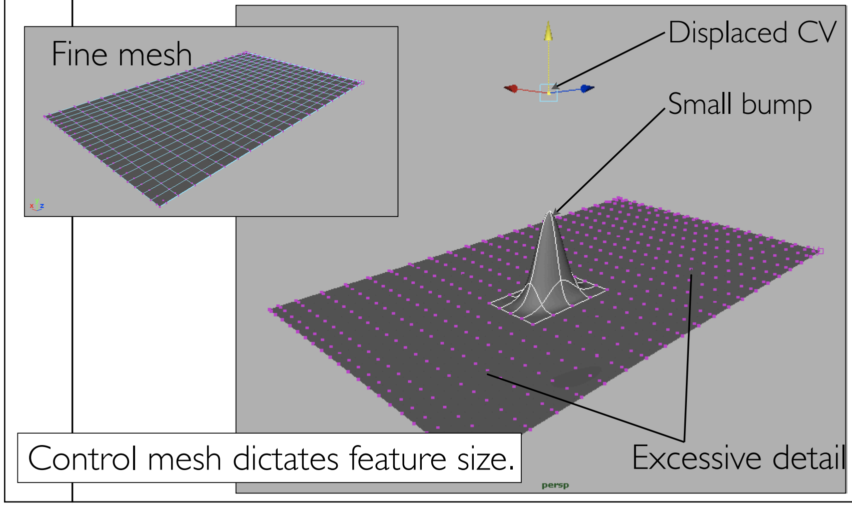
$$\mathbf{p}_i = \begin{bmatrix} p_{ix} \\ p_{iy} \\ p_{iz} \\ p_{iw} \end{bmatrix} \qquad \mathbf{x}(u) = \frac{\sum_i \begin{bmatrix} p_{ix} \\ p_{iy} \\ p_{iz} \end{bmatrix} N_i(u)}{\sum_i p_{iw} N_i(u)}$$

- Non-linear in the control points
- The p_{iw} are sometimes called "weights"

Consider NURBS Surface

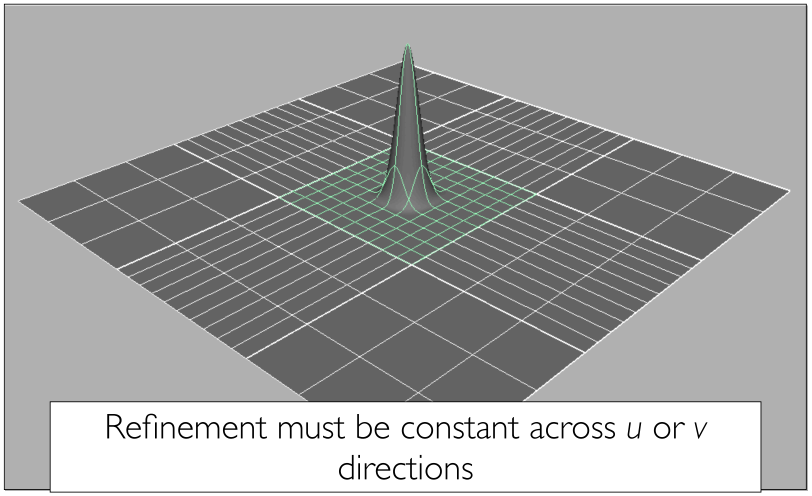


Consider NURBS Surface



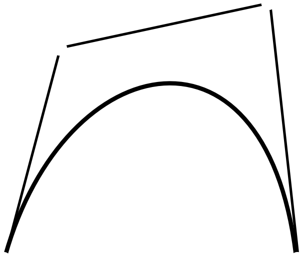
Control mesh dictates feature size.

Tensor Product Surface Refinement



Refinement must be constant across *u* or *v* directions

Bézier Subdivision



$u \in [0..1]$

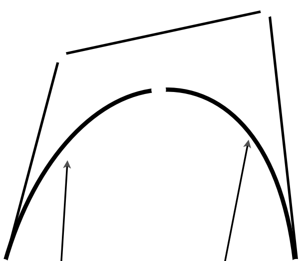
$$\mathbf{x}(u) = \sum_i b_i(u)\,\mathbf{p}_i$$

$$\mathbf{x}(u) = [1, u, u^2, u^3]\,\beta_Z\,\mathbf{P}$$

Vector of control points

$$\beta_Z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}$$

Bézier Subdivision



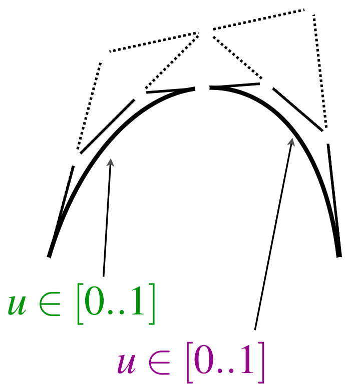
$u \in [0..1/2]$

$u \in [1/2..1]$

$$\mathbf{x}(u) = [1, u, u^2, u^3]\,\beta_Z\,\mathbf{P}$$

$$\beta_Z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}$$

Bézier Subdivision

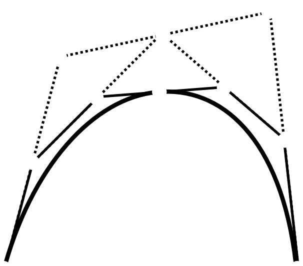


$$\mathbf{x}(u) = [1, u, u^2, u^3]\,\beta_Z\,\mathbf{P}$$

Can't change these...

$$\beta_Z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}$$

Bézier Subdivision



$$\mathbf{x}(u) = [1, u, u^2, u^3]\,\beta_Z\,\mathbf{P} \qquad u \in [0..\tfrac{1}{2}]$$

$$\mathbf{x}(u) = [1, \frac{u}{2}, \frac{u^2}{4}, \frac{u^3}{8}]\beta_Z\mathbf{P} \qquad u \in [0..1]$$

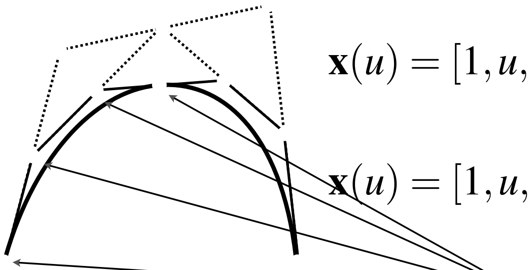
$$\mathbf{x}(u) = [1, u, u^2, u^3]\mathbf{S}_1\beta_Z\mathbf{P}$$

$$\mathbf{x}(u) = [1, u, u^2, u^3]\beta_Z\beta_Z^{-1}\mathbf{S}_1\beta_Z\mathbf{P}$$

$$\mathbf{x}(u) = [1, u, u^2, u^3]\beta_Z\mathbf{H}_{Z1}\mathbf{P}$$

$$\mathbf{S}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 \\ 0 & 0 & 1/4 & 0 \\ 0 & 0 & 0 & 1/8 \end{bmatrix}$$

Bézier Subdivision



$$\mathbf{x}(u) = [1, u, u^2, u^3]\beta_Z \mathbf{H}_{Z1}\mathbf{P}$$

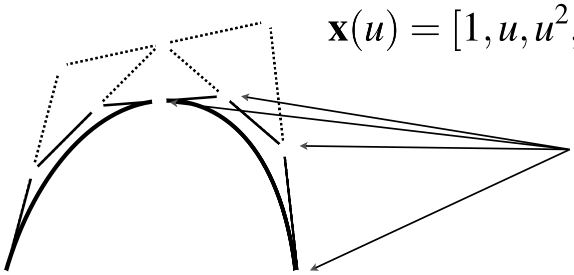
$$\mathbf{x}(u) = [1, u, u^2, u^3]\beta_Z \mathbf{P}_1$$

$$\mathbf{P}_1 = \mathbf{H}_{Z1}\mathbf{P}$$

$$\mathbf{H}_{Z1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} & 0 \\ \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{bmatrix}$$

Bézier Subdivision

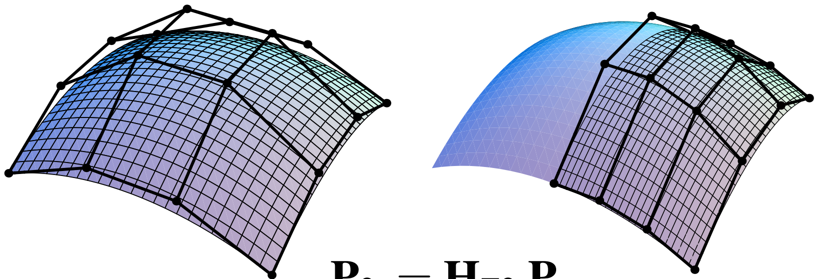
$$\mathbf{x}(u) = [1, u, u^2, u^3]\beta_Z \mathbf{P}_2$$



$$\mathbf{P}_2 = \mathbf{H}_{Z2}\mathbf{P}$$

$$\mathbf{H}_{Z2} = \begin{bmatrix} \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \\ 0 & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \mathbf{S}_2 = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{4} & \frac{1}{8} \\ 0 & \frac{1}{2} & \frac{1}{2} & \frac{3}{8} \\ 0 & 0 & \frac{1}{4} & \frac{3}{8} \\ 0 & 0 & 0 & \frac{1}{8} \end{bmatrix}$$

Bézier Subdivision

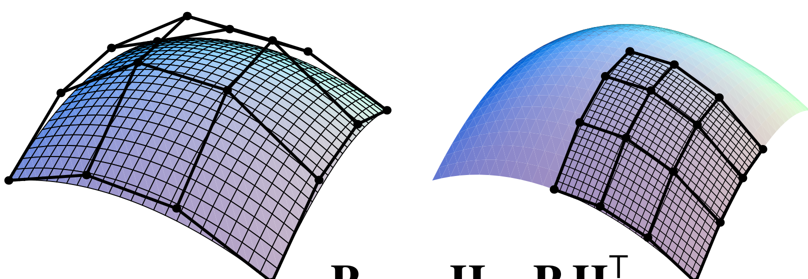


$$\mathbf{P}_{2\cdot} = \mathbf{H}_{Z2}\,\mathbf{P}$$

$$\mathbf{x}(u,v) = [1,u,u^2,u^3]\beta_Z \mathbf{P} \beta_Z^T [1,v,v^2,v^3]^T$$

4 x 4 matrix of control points

Bézier Subdivision

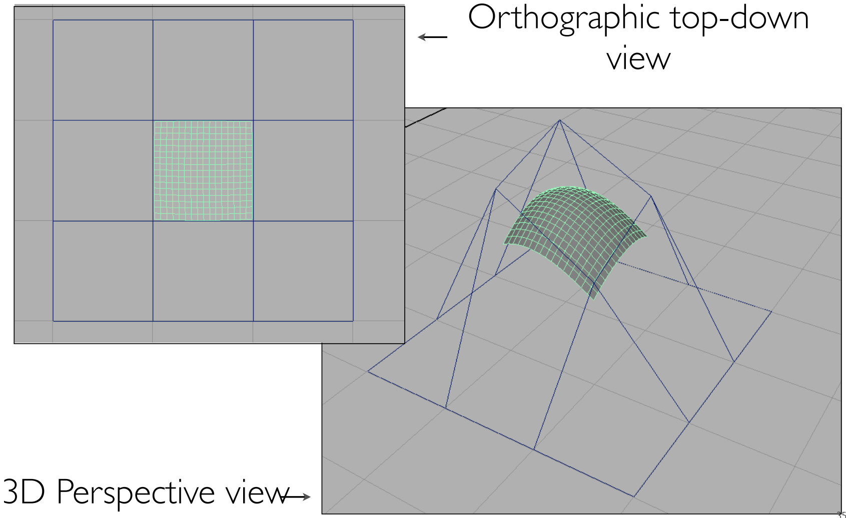


$$\mathbf{P}_{21} = \mathbf{H}_{Z2}\,\mathbf{P}\,\mathbf{H}_{Z1}^T$$

$$\mathbf{x}(u,v) = [1,u,u^2,u^3]\beta_Z \mathbf{P} \beta_Z^T [1,v,v^2,v^3]^T$$

4 x 4 matrix of control points

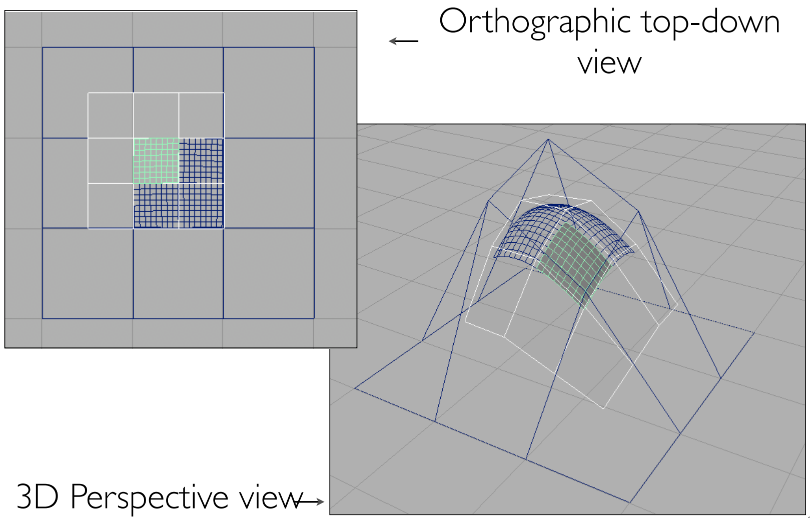
Regular B-Spline Subdivision



Orthographic top-down view

3D Perspective view

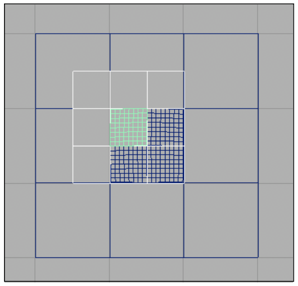
Regular B-Spline Subdivision

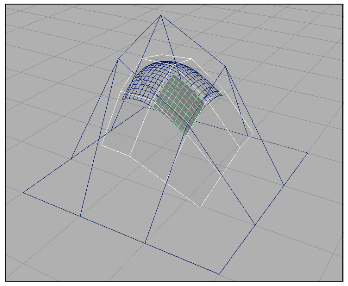


Orthographic top-down view

3D Perspective view

Regular B-Spline Subdivision

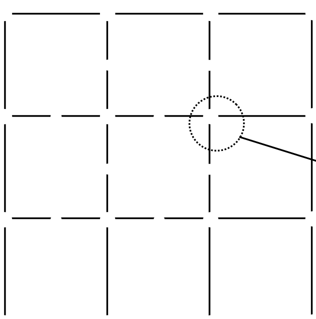




$$\mathbf{x}(u,v) = [1,u,u^2,u^3]\beta_B\,\mathbf{P}\,\beta_B^{\mathsf{T}}[1,v,v^2,v^3]^{\mathsf{T}}$$

$$\mathbf{P}_{11} = \mathbf{H}_{B1}\,\mathbf{P}\,\mathbf{H}_{B1}^{\mathsf{T}}$$

Regular B-Spline Subdivision

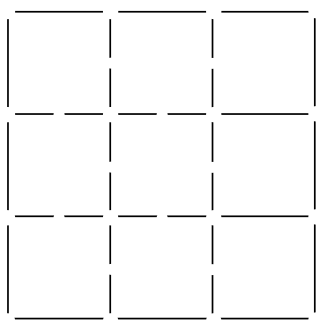


$$\mathbf{P}_{11} = \mathbf{H}_{B1}\,\mathbf{P}\,\mathbf{H}_{B1}^{\mathsf{T}}$$

In this parametric view these knot points are collocated.

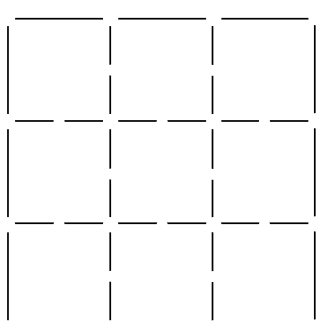
The 3D control points are not.

Regular B-Spline Subdivision



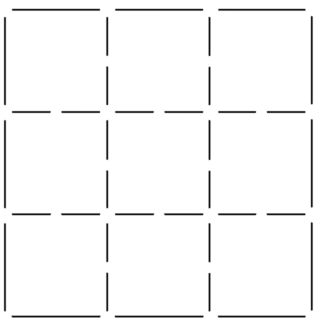
$$\mathbf{P}_{11} = \mathbf{H}_{\mathrm{B}1}\,\mathbf{P}\,\mathbf{H}_{\mathrm{B}1}^{\mathsf{T}}$$
$$\mathbf{P}_{12} = \mathbf{H}_{\mathrm{B}1}\,\mathbf{P}\,\mathbf{H}_{\mathrm{B}2}^{\mathsf{T}}$$

Regular B-Spline Subdivision



$$\mathbf{P}_{11} = \mathbf{H}_{\mathrm{B}1}\,\mathbf{P}\,\mathbf{H}_{\mathrm{B}1}^{\mathsf{T}}$$
$$\mathbf{P}_{12} = \mathbf{H}_{\mathrm{B}1}\,\mathbf{P}\,\mathbf{H}_{\mathrm{B}2}^{\mathsf{T}}$$
$$\mathbf{P}_{22} = \mathbf{H}_{\mathrm{B}2}\,\mathbf{P}\,\mathbf{H}_{\mathrm{B}2}^{\mathsf{T}}$$

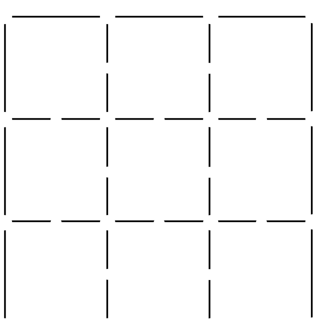
Regular B-Spline Subdivision



$$\mathbf{P}_{11} = \mathbf{H}_{B1}\,\mathbf{P}\,\mathbf{H}_{B1}^{\top}$$
$$\mathbf{P}_{12} = \mathbf{H}_{B1}\,\mathbf{P}\,\mathbf{H}_{B2}^{\top}$$
$$\mathbf{P}_{22} = \mathbf{H}_{B2}\,\mathbf{P}\,\mathbf{H}_{B2}^{\top}$$
$$\mathbf{P}_{21} = \mathbf{H}_{B2}\,\mathbf{P}\,\mathbf{H}_{B1}^{\top}$$

$$\mathbf{H}_{B1} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{8} & \frac{3}{4} & \frac{1}{8} & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{8} & \frac{3}{4} & \frac{1}{8} \end{bmatrix} \quad \mathbf{H}_{B2} = \begin{bmatrix} \frac{1}{8} & \frac{3}{4} & \frac{1}{8} & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{8} & \frac{3}{4} & \frac{1}{8} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

Regular B-Spline Subdivision



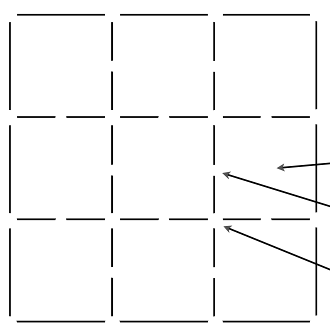
$$\mathbf{P}^{i+1} = \mathbf{H}\,\mathbf{P}^{i}$$

Length 25 vector of fine CPs

25 × 16 subdivision matrix

Length 16 vector of coarse CP

Regular B-Spline Subdivision



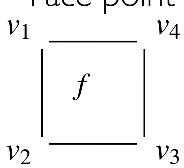
$$\mathbf{P}^{i+1} = \mathbf{H}\,\mathbf{P}^{i}$$

Inspection would reveal a pattern

- Face points
- Edge points
- Vertex points

Regular B-Spline Subdivision

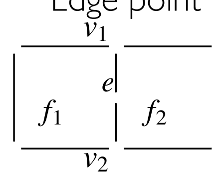
Face point

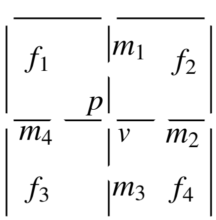


$$f = \frac{v_1 + v_2 + v_3 + v_4}{4}$$

$$e = \frac{v_1 + v_2 + f_1 + f_2}{4}$$

Edge point





Vertex point

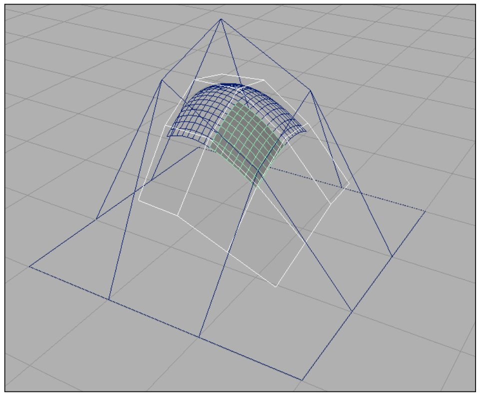
$$v = \frac{f_1 + f_2 + f_3 + f_4 + 2(m_1 + m_2 + m_3 + m_4) + 4p}{16}$$

m midpoint of edge, not "edge point"

p old "vertex point"

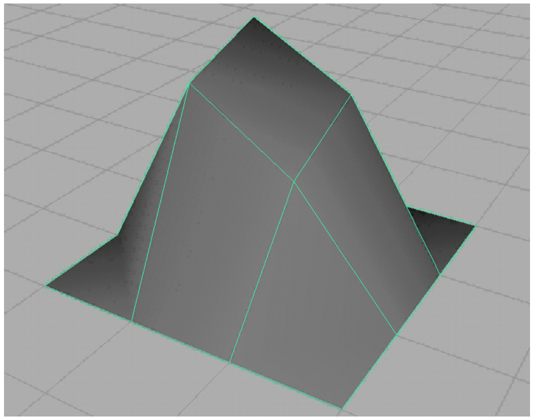
Regular B-Spline Subdivision

- Recall that control mesh approaches surface



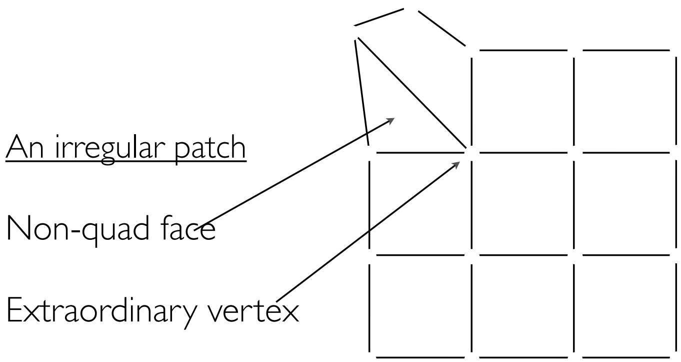
Regular B-Spline Subdivision

- Limit of subdivision is the surface



Irregular B-Spline Subdivision

- Catmull-Clark Subdivision
  - Generalizes regular B-Spine subdivision



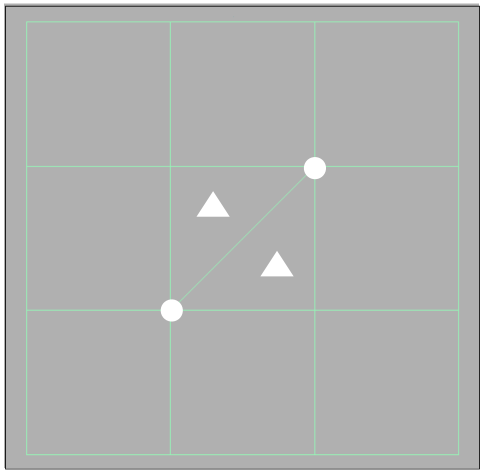
Irregular B-Spline Subdivision

- Catmull-Clark Subdivision
  - Generalizes regular B-Spine subdivision
  - Rules reduce to regular for ordinary vertices/faces

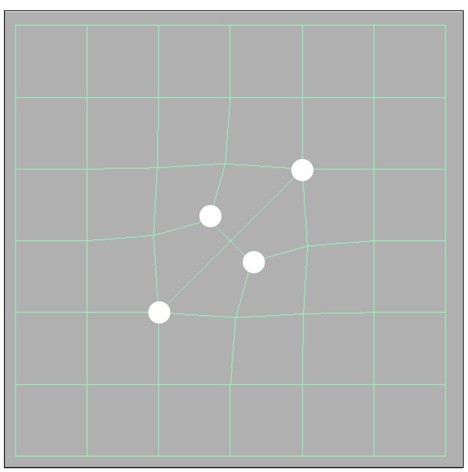
$$f = \text{average of surrounding vertices}$$
$$e = \frac{f_1 + f_2 + v_1 + v_2}{4}$$
$$v = \frac{\bar{f}}{n} + \frac{2\bar{m}}{n} + \frac{p(n-3)}{n}$$

$$\bar{m} = \text{average of adjacent midpoints}$$
$$\bar{f} = \text{average of adjacent face points}$$
$$n = \text{valence of vertex}$$

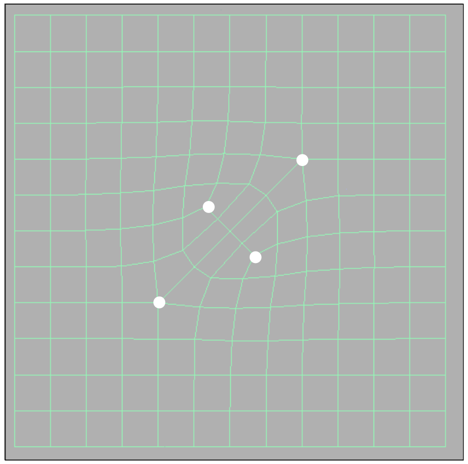
Catmull-Clark Subdivision



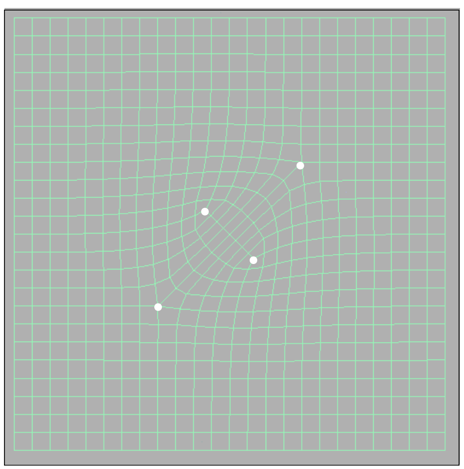
Catmull-Clark Subdivision



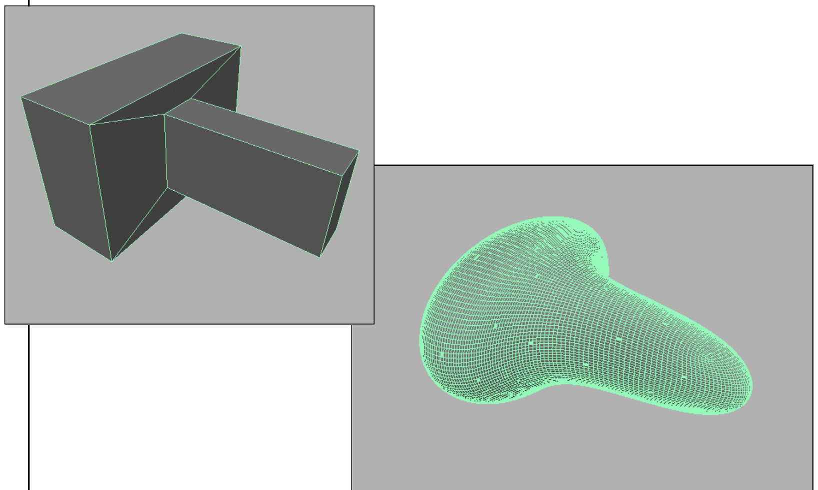
Catmull-Clark Subdivision



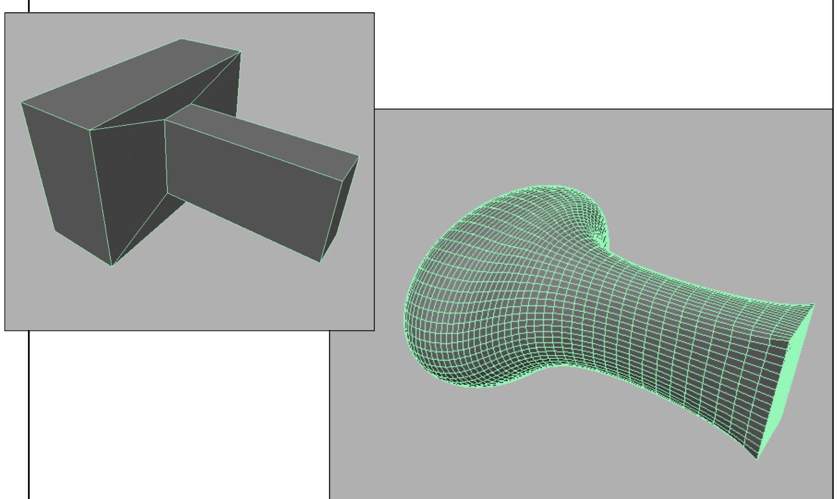
Catmull-Clark Subdivision



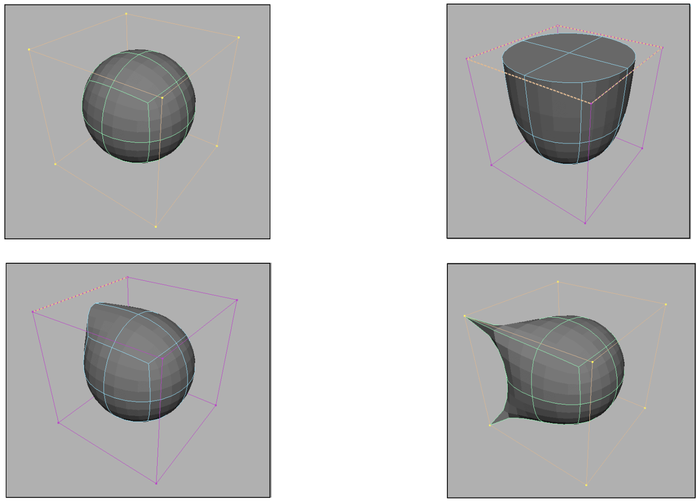
Catmull-Clark Subdivision



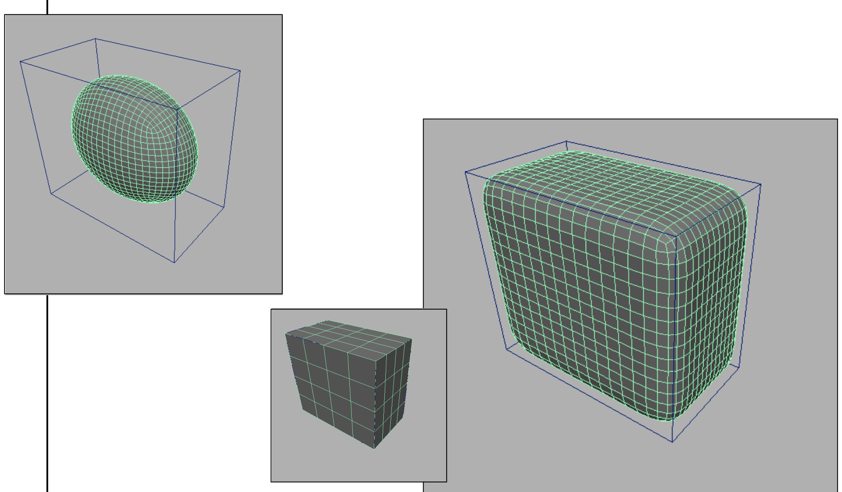
Catmull-Clark Subdivision



Catmull-Clark Subdivision



Catmull-Clark Subdivision



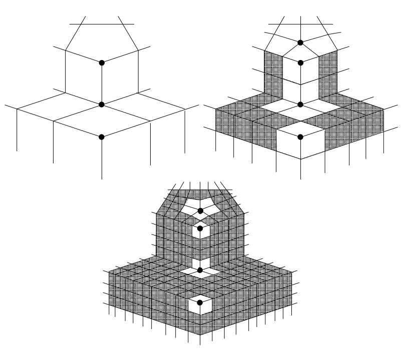
Practical Evaluation

- Uniform subdivision
  - Exponential growth of control mesh
  - Need several subdivisions before error is small
  - Ok if you are "drawing and forgetting", otherwise impractical
- Evaluation at arbitrary points
- Tangent and other derivative evaluation needed

- Paper by Jos Stam SIGGRAPH 98 outlines efficient method

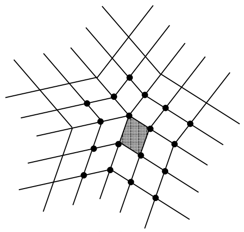
Isolated Extraordinary Points

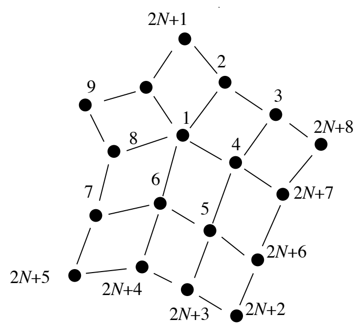
- Where surface is regular, treat as b-splines
- We can assume extraordinary points are isolated



Isolated Extraordinary Points

- Neighborhood around an extraordinary point





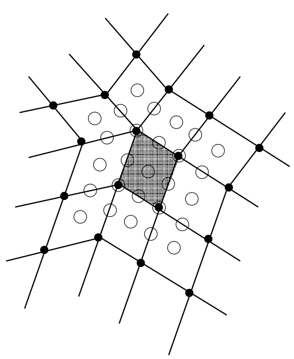
Subdivision Matrix

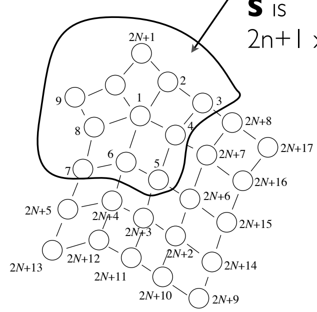
$$\mathbf{C}_1 = \mathbf{A}\mathbf{C}_0.$$

$$\mathbf{C}_n = \mathbf{A}\mathbf{C}_{n-1} = \mathbf{A}^n\mathbf{C}_0$$

$$\mathbf{A} = \begin{pmatrix} \mathbf{S} & \mathbf{0} \\ \mathbf{S}_{11} & \mathbf{S}_{12} \end{pmatrix}$$

S is
2n+1 x 2n+1





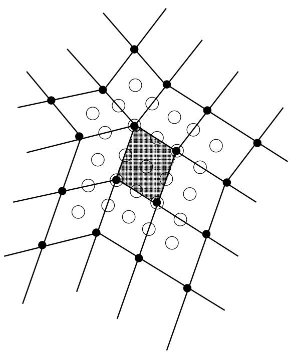
Subdivision Matrix

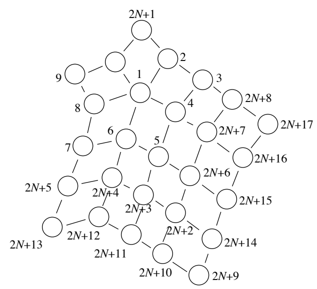
$$\mathbf{C}_1 = \mathbf{A}\mathbf{C}_0.$$

$$\mathbf{C}_n \quad = \quad \mathbf{A}\mathbf{C}_{n-1} = \mathbf{A}^n\mathbf{C}_0$$

$$\bar{\mathbf{C}}_n \quad = \quad \bar{\mathbf{A}}\mathbf{C}_{n-1} = \bar{\mathbf{A}}\mathbf{A}^{n-1}\mathbf{C}_0, \quad n \geq 1$$

$$\bar{\mathbf{A}} = \begin{pmatrix} \mathbf{S} & \mathbf{0} \\ \mathbf{S}_{11} & \mathbf{S}_{12} \\ \mathbf{S}_{21} & \mathbf{S}_{22} \end{pmatrix}$$





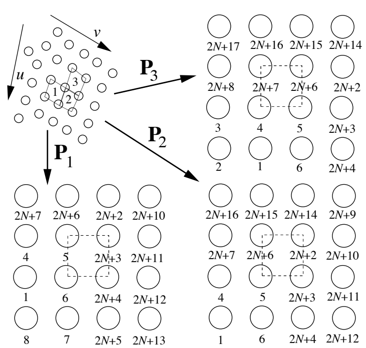
Subdivision Matrix

$$\mathbf{C}_1 = \mathbf{A}\mathbf{C}_0.$$

$$\mathbf{C}_n \quad = \quad \mathbf{A}\mathbf{C}_{n-1} = \mathbf{A}^n\mathbf{C}_0$$

$$\bar{\mathbf{C}}_n \quad = \quad \bar{\mathbf{A}}\mathbf{C}_{n-1} = \bar{\mathbf{A}}\mathbf{A}^{n-1}\mathbf{C}_0, \quad n \geq 1$$

$$\bar{\mathbf{A}} = \begin{pmatrix} \mathbf{S} & \mathbf{0} \\ \mathbf{S}_{11} & \mathbf{S}_{12} \\ \mathbf{S}_{21} & \mathbf{S}_{22} \end{pmatrix}$$



Eigen Space

$$\mathbf{C}_1 = \mathbf{A}\mathbf{C}_0.$$
$$\mathbf{C}_n = \mathbf{A}\mathbf{C}_{n-1} = \mathbf{A}^n\mathbf{C}_0$$

$$\mathbf{A} = \begin{pmatrix} \mathbf{S} & \mathbf{0} \\ \mathbf{S}_{11} & \mathbf{S}_{12} \end{pmatrix}$$

$$\mathbf{AV} = \mathbf{V\Lambda} \longrightarrow \mathbf{A} = \mathbf{V\Lambda V}^{-1}$$

$$\bar{\mathbf{C}}_n = \bar{\mathbf{A}}\mathbf{A}^{n-1}\mathbf{C}_0 = \bar{\mathbf{A}}\mathbf{V}\mathbf{\Lambda}^{n-1}\mathbf{V}^{-1}\mathbf{C}_0$$

$$\mathbf{s}_{k,n}(u,v) = \hat{\mathbf{C}}_0^T \underline{\mathbf{\Lambda}^{n-1}\left(\mathbf{P}_k\bar{\mathbf{A}}\mathbf{V}\right)^T \mathbf{b}(u,v)}$$

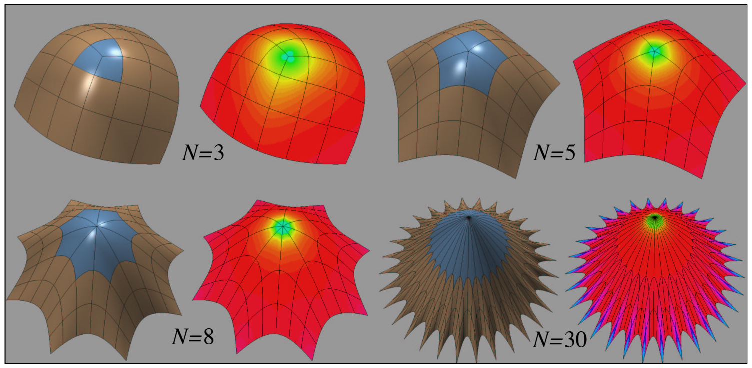
$$\hat{\mathbf{C}}_0 = \mathbf{V}^{-1}\mathbf{C}_0$$

Only depends on valence of extraordinary vertex.

Comments

- Computing Eigen Vectors is tricky
  - See Jos's paper for details
  - He includes the solutions up to valence 500
- All eigenvalues are (abs) less than one
  - except for lead value which is exactly one
  - well defined limit behavior
- Exact evaluation allows "pushing to limit surface"

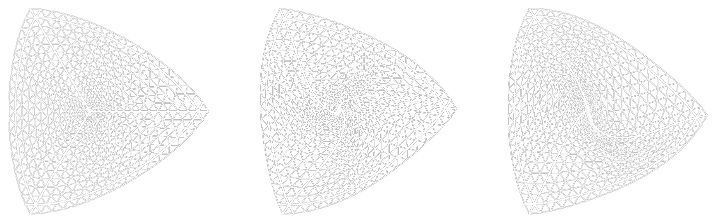
Curvature Plots



Behavior of Limit Surface

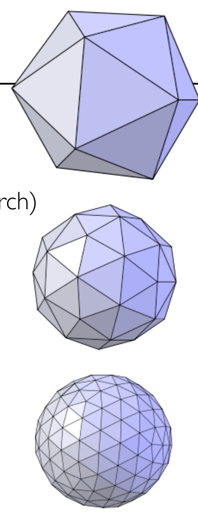
- Away from extraordinary points simple B-Splines
- Existence at extraordinary points
  - Limit of $\mathbf{A}^n \mathbf{C}_0$ must be finite so $|\lambda_i| \leq 1$
  - Not useful if surface collapses to a point $\lambda_0 = 1$
  - Invariant w.r.t. translation $\mathbf{v}_0 = \mathbf{1}$
    - $\mathbf{A}^n \mathbf{C}_0 + \mathbf{t} = \mathbf{A}^n(\mathbf{C}_0 + \mathbf{t})$
- Smoothness at extraordinary points
  - Curve $1 > |\lambda_1| > |\lambda_2|$
  - Surface $1 > |\lambda_1| \geq |\lambda_2| > |\lambda_3|$

Eigenvalues Control Behavior

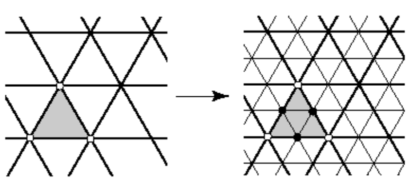


Real Complex Degenerate

Loop Subdivision

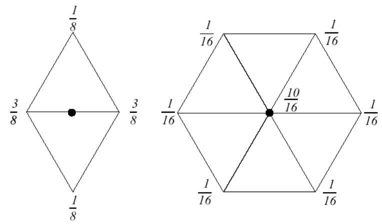


- "Loop" is a name
  - Charles Loop (currently working at Microsoft Research)
  - Smooth Subdivision Surfaces Based on Triangles, M.S. Mathematics thesis , University of Utah, 1987
- Triangle-based subdivision scheme
  - Each triangle split into four new triangles



Loop Subdivision

- Regular vertices



- Extraordinary vertices
  - Original Loop
    $$\beta = \frac{1}{n}\left(\frac{5}{b} - \left(\frac{3}{8} + \frac{1}{4}\cos\frac{2\pi}{n}\right)^2\right)$$
  - Warren
    $$\beta = \begin{cases} \frac{3}{8n} & n > 3 \\ \frac{3}{16} & n = 3 \end{cases}$$