Surfaces and solids

CS 4620 Lecture 19

Modeling in 3D

- Representing subsets of 3D space
  - volumes (3D subsets)
  - surfaces (2D subsets)
  - curves (1D subsets)
  - points (0D subsets)

Representing geometry

- In order of dimension…
- Points: trivial case
- Curves
  - normally use parametric representation
  - line—just a point and a vector (like ray in ray tracer)
    - polylines (approximation scheme for drawing)
  - more general curves: usually use splines
    - $\mathbf{p}(t)$ is from R to R^3
    - $\mathbf{p}$ is defined by piecewise polynomial functions

Representing geometry

- Surfaces
  - this case starts to get interesting
  - implicit and parametric representations both useful
  - example: plane
    - implicit: vector from point perpendicular to normal
    - parametric: point plus scaled tangent
  - example: sphere
    - implicit: distance from center equals *r*
    - parametric: write out in spherical coordinates
      - messiness of parametric form not unusual

Representing geometry

- Volumes
  - boundary representations (B-reps)
    - just represent the boundary surface
    - convenient for many applications
    - must be closed (watertight) to be meaningful
      - an important constraint to maintain in many applications

Representing geometry

- Volumes
  - CSG (Constructive Solid Geometry)
    - apply boolean operations on solids
    - simple to define
    - simple to compute in some cases
      - [e.g. ray tracing]
    - difficult to compute stably with B-reps
      - [e.g. coincident surfaces]

Specific surface representations

- Parametric spline surfaces
  - extrusions
  - surfaces of revolution
  - generalized cylinders
  - spline patches

[Hearn & Baker]

From curves to surfaces

- So far have discussed spline curves in 2D
  - it turns out that this already provides of the mathematical machinery for several ways of building curved surfaces
- Building surfaces from 2D curves
  - extrusions and surfaces of revolution
- Building surfaces from 2D and 3D curves
  - generalized swept surfaces
- Building surfaces from spline patches
  - generalizing spline curves to spline patches
- Also to think about: generating triangles

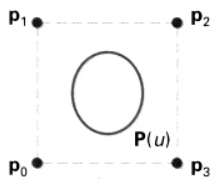
Extrusions

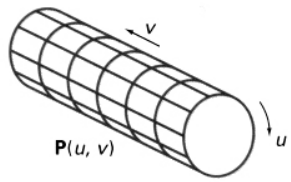
- Given a spline curve $C \in \mathbb{R}^2$, define $S \in \mathbb{R}^3$ by

  $$S = C \times [a, b]$$

- This produces a "tube" with the given cross section
  - Circle: cylinder; "L": shelf bracket; "I": I beam
- Parameterized by the spline's u and $v \in [a, b]$

  $$\mathbf{P}(u, v) = (c_x(u), c_y(u), v)$$



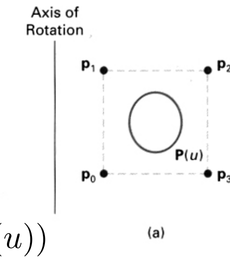


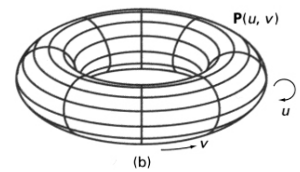
Surfaces of revolution

- Take a 2D curve and spin it around an axis
- Given curve $\mathbf{p}(u)$ in the plane, the surface is defined easily in cylindrical coordinates:

  $$\mathbf{P}(u, v) = (r, \phi, z) = (p_x(u), v, p_y(u))$$

  - the torus is an example in which the curve **p** is a circle

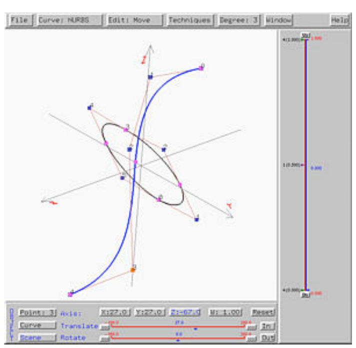


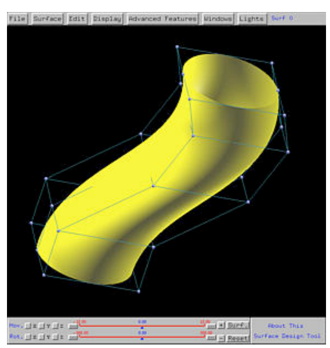


[Hearn & Baker]

Swept surfaces

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

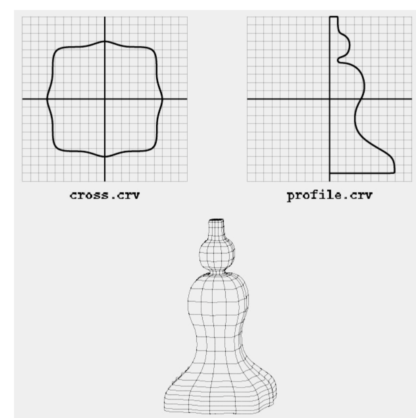




[C

Generalized cylinders

- General swept surfaces
  - varying radius
  - varying cross-section
  - curved axis

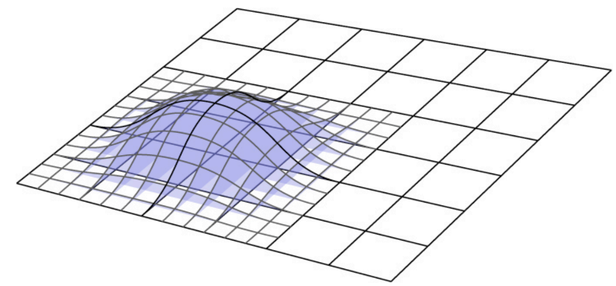


[Snyder 1992]

From curves to surface patches

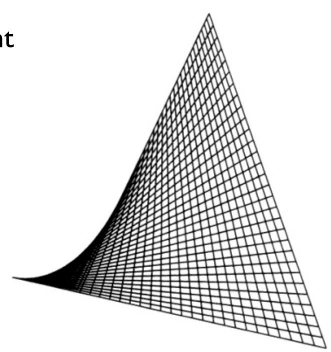
- Curve was sum of weighted 1D basis functions
- Surface is sum of weighted 2D basis functions
  - construct them as separable products of 1D fns.
  - choice of different splines
    - spline type
    - order
    - closed/open (B-spline)

Separable (tensor) product construction



Bilinear patch

- Simplest case: 4 points, tensor product of two linear segments
  - basis function is a 3D tent

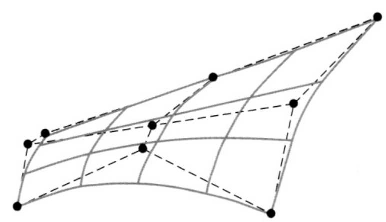


(c)

[Rogers]

Biquadratic Bézier patch

- Tensor product of quadratic Bézier curves

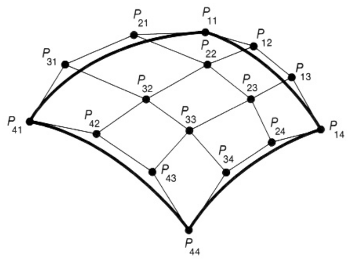


(a)

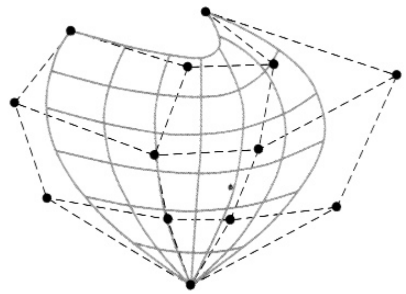
[Hearn & Baker]

Bicubic Bézier patch

- Tensor product of two cubic Bézier segments



[Foley et al.]



(b)

[Hearn & Baker]

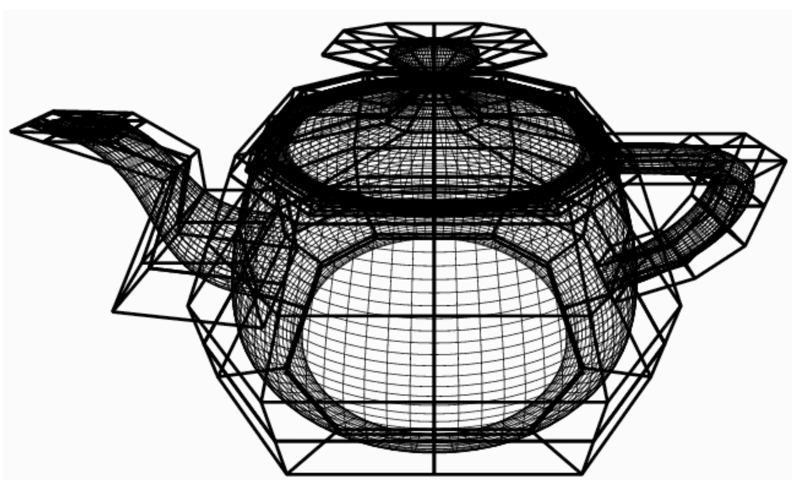
  - properties that carry over
    - interpolation at corners, edges
    - tangency at corners, edges
    - convex hull

Utah Teapot: Bicubic Bézier patches

```
Rim:
{ 102, 103, 104, 105,   4,   5,   6,   7,
    8,   9,  10,  11,  12,  13,  14,  15 }
Body:
{  12,  13,  14,  15,  16,  17,  18,  19,
   20,  21,  22,  23,  24,  25,  26,  27 }
{  24,  25,  26,  27,  29,  30,  31,  32,
   33,  34,  35,  36,  37,  38,  39,  40 }
Lid:
{  96,  96,  96,  96,  97,  98,  99, 100,
  101, 101, 101, 101,   0,   1,   2,   3 }
{   0,   1,   2,   3, 106, 107, 108, 109,
  110, 111, 112, 113, 114, 115, 116, 117 }
Handle:
{  41,  42,  43,  44,  45,  46,  47,  48,
   49,  50,  51,  52,  53,  54,  55,  56 }
{  53,  54,  55,  56,  57,  58,  59,  60,
   61,  62,  63,  64,  28,  65,  66,  67 }
Spout:
{  68,  69,  70,  71,  72,  73,  74,  75,
   76,  77,  78,  79,  80,  81,  82,  83 }
{  80,  81,  82,  83,  84,  85,  86,  87,
   88,  89,  90,  91,  92,  93,  94,  95 }
```

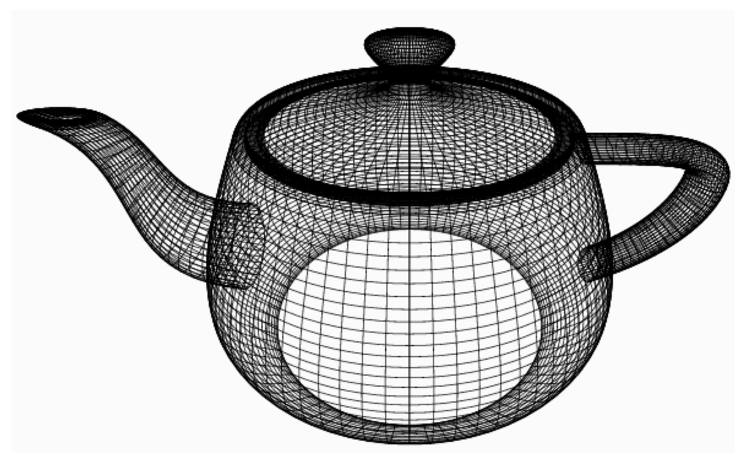
```
Vertices:

{  0.2000,  0.0000, 2.70000 }, {  0.2000, -0.1120, 2.70000 },
{  0.1120, -0.2000, 2.70000 }, {  0.0000, -0.2000, 2.70000 },
{  1.3375,  0.0000, 2.53125 }, {  1.3375, -0.7490, 2.53125 },
{  0.7490, -1.3375, 2.53125 }, {  0.0000, -1.3375, 2.53125 },
{  1.4375,  0.0000, 2.53125 }, {  1.4375, -0.8050, 2.53125 },
{  0.8050, -1.4375, 2.53125 }, {  0.0000, -1.4375, 2.53125 },
{  1.5000,  0.0000, 2.40000 }, {  1.5000, -0.8400, 2.40000 },
{  0.8400, -1.5000, 2.40000 }, {  0.0000, -1.5000, 2.40000 },
{  1.7500,  0.0000, 1.87500 }, {  1.7500, -0.9800, 1.87500 },
{  0.9800, -1.7500, 1.87500 }, {  0.0000, -1.7500, 1.87500 },
{  2.0000,  0.0000, 1.35000 }, {  2.0000, -1.1200, 1.35000 },
{  1.1200, -2.0000, 1.35000 }, {  0.0000, -2.0000, 1.35000 },
{  2.0000,  0.0000, 0.90000 }, {  2.0000, -1.1200, 0.90000 },
{  1.1200, -2.0000, 0.90000 }, {  0.0000, -2.0000, 0.90000 },
{ -2.0000,  0.0000, 0.90000 }, {  2.0000,  0.0000, 0.45000 },
{  2.0000, -1.1200, 0.45000 }, {  1.1200, -2.0000, 0.45000 },
{  0.0000, -2.0000, 0.45000 }, {  1.5000,  0.0000, 0.22500 },
{  1.5000, -0.8400, 0.22500 }, {  0.8400, -1.5000, 0.22500 },
{  0.0000, -1.5000, 0.22500 }, {  1.5000,  0.0000, 0.15000 },
{  1.5000, -0.8400, 0.15000 }, {  0.8400, -1.5000, 0.15000 },
{  0.0000, -1.5000, 0.15000 }, { -1.6000,  0.0000, 2.02500 },
{ -1.6000, -0.3000, 2.02500 }, { -1.5000, -0.3000, 2.25000 },
{ -1.5000,  0.0000, 2.25000 }, { -2.3000,  0.0000, 2.02500 },
{ -2.3000, -0.3000, 2.02500 }, { -2.5000, -0.3000, 2.25000 },
{ -2.5000,  0.0000, 2.25000 }, { -2.7000,  0.0000, 2.02500 },
{ -2.7000, -0.3000, 2.02500 }, { -3.0000, -0.3000, 2.25000 },
{ -3.0000,  0.0000, 2.25000 }, { -2.7000,  0.0000, 1.80000 },
{ -2.7000, -0.3000, 1.80000 }, { -3.0000, -0.3000, 1.80000 },
{ -3.0000,  0.0000, 1.80000 }, { -2.7000,  0.0000, 1.57500 },
{ -2.7000, -0.3000, 1.57500 }, { -3.0000, -0.3000, 1.35000 },
{ -3.0000,  0.0000, 1.35000 }, { -2.5000,  0.0000, 1.12500 },
{ -2.5000, -0.3000, 1.12500 }, { -2.6500, -0.3000, 0.93750 },
{ -2.6500,  0.0000, 0.93750 }, { -2.0000, -0.3000, 0.90000 },
{ -1.9000, -0.3000, 0.60000 }, { -1.9000,  0.0000, 0.60000 },
{  1.7000,  0.0000, 1.42500 }, {  1.7000, -0.6600, 1.42500 },
{  1.7000, -0.6600, 0.60000 }, {  1.7000,  0.0000, 0.60000 },
```


Utah Teapot: Bicubic Bézier patches



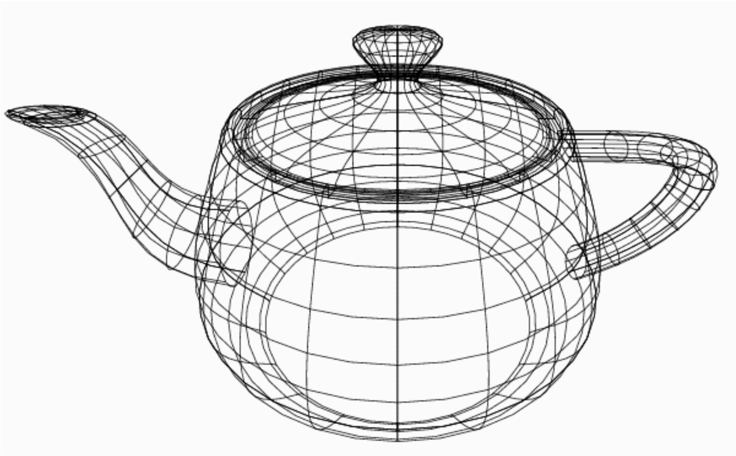
http://www.holmes3d.net/graphics/teapot

Utah Teapot: Bicubic Bézier patches



http://www.holmes3d.net/graphics/teapot

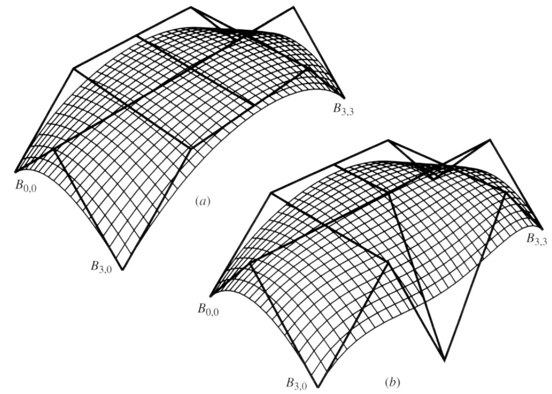
Utah Teapot: Bicubic Bézier patches



http://www.holmes3d.net/graphics/teapot

3x5 Bézier patch

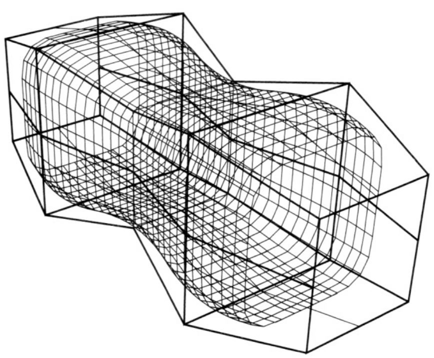
- Tensor product of quadratic and quartic Béziers



[Rogers]

Cylindrical B-spline surfaces

- Tensor product of closed and open cubic B-splines



(b)

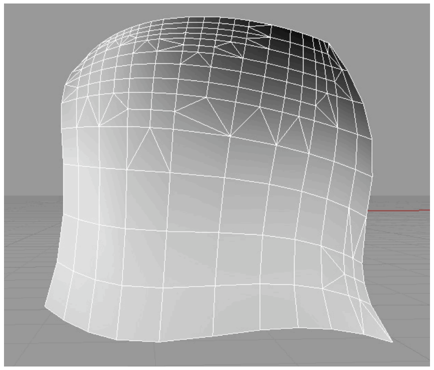
[Rogers]

Approximating spline surfaces

- Similar to curves, approximate with simple primitives
  - in surface case, triangles or quads
  - quads widely used because they fit in parameter space
    - generally eventually rendered as pairs of triangles
- Adaptive subdivision
  - basic approach: recursively test flatness
    - if the patch is not flat enough, subdivide into four using curve subdivision twice, and recursively process each piece
    - See HW8
  - as with curves, convex hull property is useful for termination testing (and is inherited from the curves)

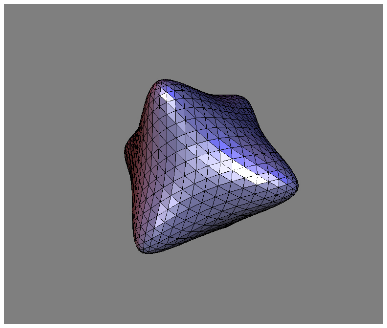
Approximating spline surfaces

- With adaptive subdivision, must take care with cracks
  - at the boundaries between degrees of subdivision
  - "T vertices"



Subdivision Surfaces

- Based on polygon meshes (quads or triangles)
- Rules for subdividing surface by adding new vertices
- Converges to continuous limit surface



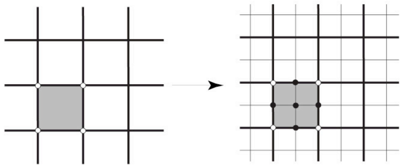
Subdivision Surfaces: Key Properties

(Subdivision Course Notes [Zorin et al. 2000])

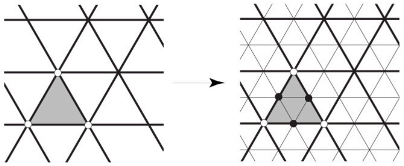
- **Efficiency:** the location of new points should be computed with a small number of floating point operations;
- **Compact support:** the region over which a point influences the shape of the final curve or surface should be small and finite;
- **Local definition:** the rules used to determine where new points go should not depend on "far away" places;
- **Affine invariance:** if the original set of points is transformed, e.g., translated, scaled, or rotated, the resulting shape should undergo the same transformation;
- **Simplicity:** determining the rules themselves should preferably be an offline process and there should only be a small number of rules;
- **Continuity:** what kind of properties can we prove about the resulting curves and surfaces, for example, are they differentiable?

Subdivision of meshes

- Quadrilaterals
  - Catmull-Clark 1978
- Triangles
  - Loop 1987



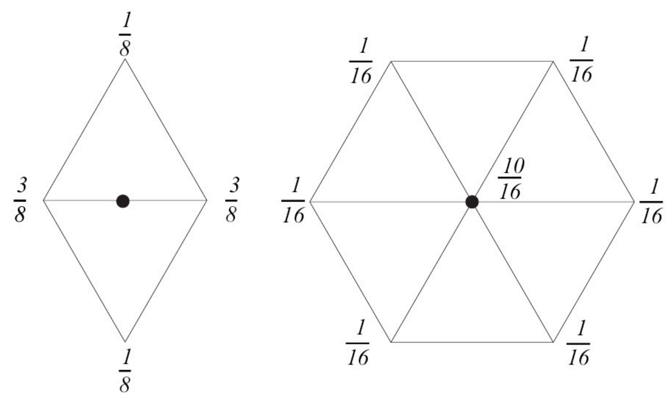
Face split for quads



Face split for triangles

[Schröder & Zorin SIGGRAPH 2000 course 23]

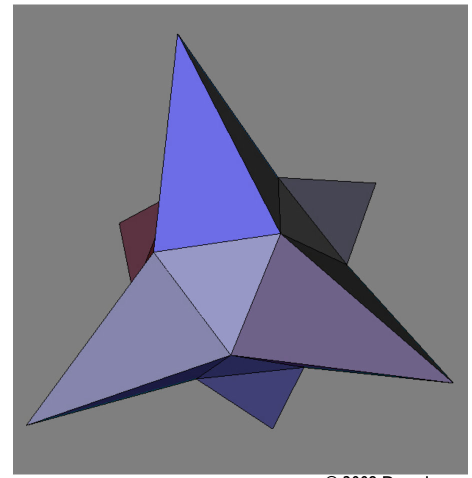
Loop regular rules



[Schröder & Zorin SIGGRAPH 2000 course 23]

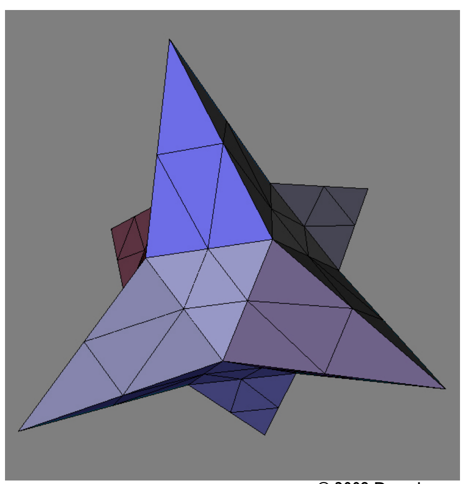
Loop Subdivision Example

control polyhedron



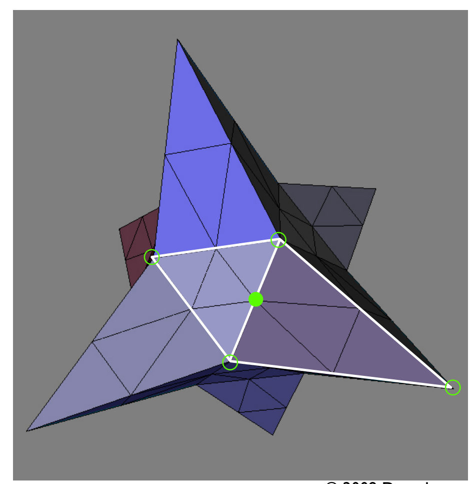
Loop Subdivision Example

refined
control polyhedron

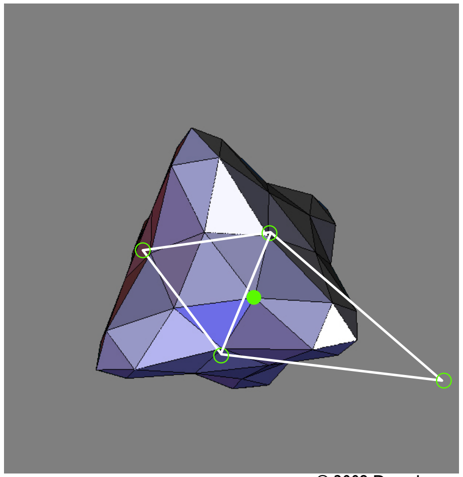


Loop Subdivision Example

odd
subdivision mask



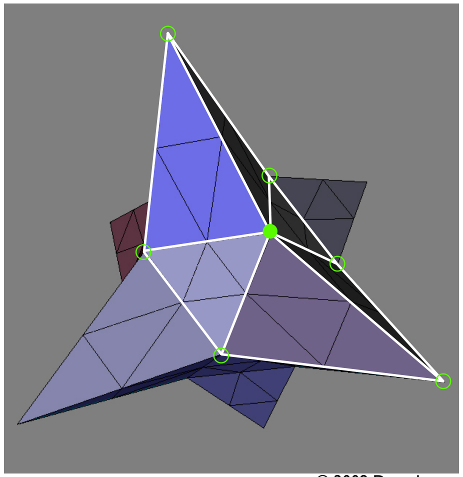
Loop Subdivision Example



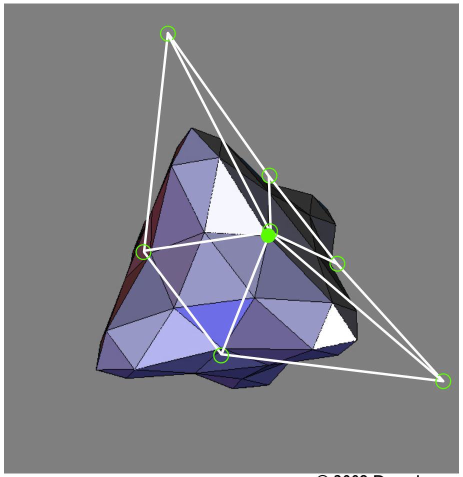
subdivision level 1

Loop Subdivision Example

even
subdivision mask
(ordinary vertex)

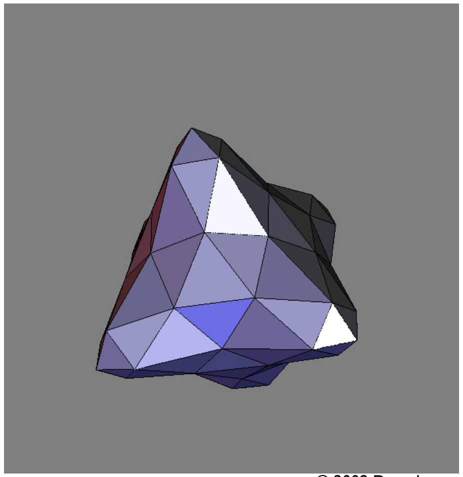


Loop Subdivision Example



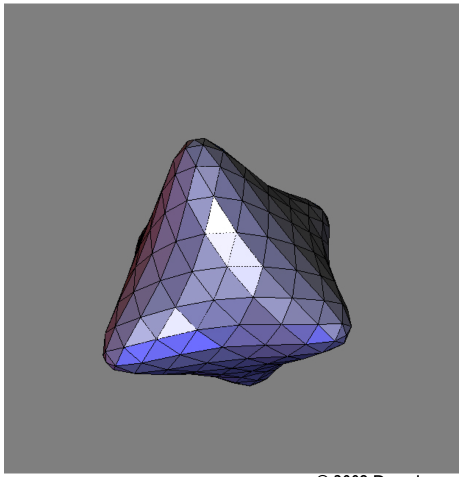
subdivision level 1

Loop Subdivision Example



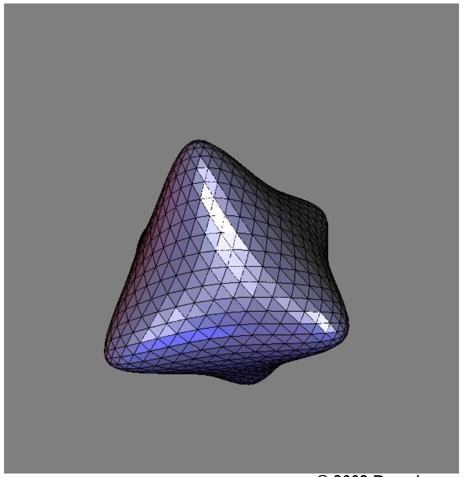
subdivision level 1

Loop Subdivision Example



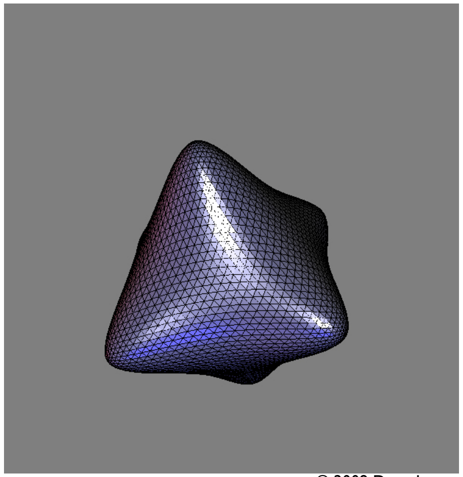
subdivision level 2

Loop Subdivision Example



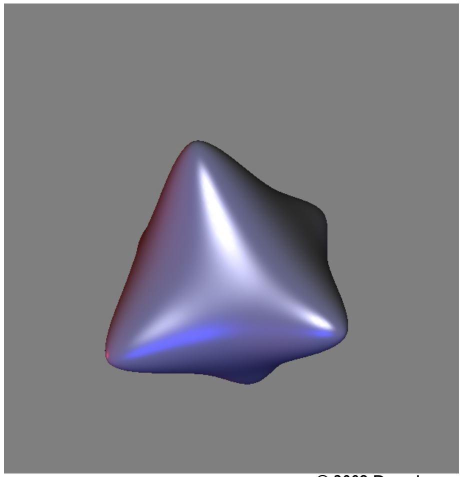
subdivision level 3

Loop Subdivision Example



subdivision level 4

Loop Subdivision Example



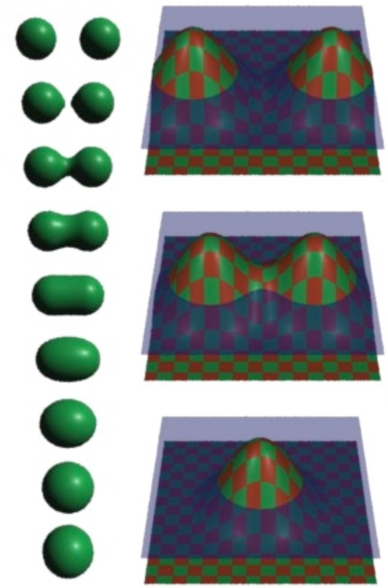
limit surface

Curve Subdivision

- Caltech subdivision applets
  - Interpolating: 4-point scheme
  - Approximating: Chaikin's algorithm
- Subdivision of B-spline control points
  - Reference: Subdivision course notes, Section 2.2

Specific surface reps.

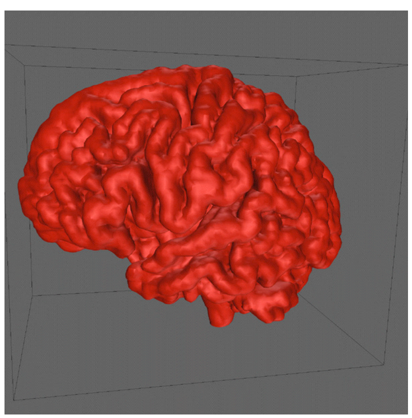
- Algebraic implicit surfaces
  - defined as zero sets of fairly arbitrary functions
  - good news: CSG is easy using min/max
  - bad news: rendering is tough
    - ray tracing: intersect arbitrary zero sets w/ray
    - pipeline: need to convert to triangles
  - e.g. "blobby" modeling



[Pixar | RenderMan Artist tools]

Specific surface representations

- Isosurface of volume data
  - implicit representation
  - function defined by regular samples on a 3D grid
    - (like an image but in 3D)
  - example uses:
    - medical imaging
    - numerical simulation



[source unknown]

Voxel Modeling Demo

3D COAT

http://www.3d-coat.com/gallery2/main.php?g2_itemId=16

Specific surface representations

- Triangle or polygon meshes
  - parametric (per face)
  - very widely used
    - final representation for pipeline rendering
    - these days restricting to triangles is common
  - rather unstructured
    - need to be careful to enforce necessary constraints
    - to bound a volume want a watertight *manifold* mesh

[Foley et al.]

