Advanced Graphics 2004

Subdivision curves & surfaces
Lecture 6: 26th October 2004

Beware: some slides contain multi-layer animations, which do not print well.
©2002,2003 Neil Dodgson

Modelling smooth 3D surfaces

Where are smooth 3D surfaces used?
- Computer Aided Design (CAD)
  - First developed for cars & aeroplanes
  - Adopted for other manufactured objects
- Computer animation
- What mechanisms exist?
  - Bézier patches
  - NURBS surfaces
  - Subdivision surfaces

Desirable features

- Need to handle any surface
- Need guaranteed continuity
  - C1-continuity
    - Smooth surfaces
  - C2-continuity
    - Smoothly reflecting surfaces
    - Required for some aerodynamics
- Need to allow discontinuities
  - Edges, creases and holes
- Needs to be easy to use

History of 3D modelling 1/3

- Some mechanism was needed for modelling 3D surfaces
- Hermite interpolation was generalised to bivariate patches
  - ...but proved too difficult to use in practice
- Bézier patches
  - Developed for car design around 1960
    - Bézier (Renault), de Casteljau (Citroën), de Boor (GM)

History of 3D modelling 2/3

- B-spline theory
  - Developed in the 1960s and '70s, led to:
- NURBS (Non-Uniform Rational B-Splines)
  - More general than Bézier patches
    - Béziers are special cases of NURBS
  - NURBS quickly became the industry standard in CAD
    - ...and remain the industry standard today
  - Adopted by the computer animation industry when it began

History of 3D modelling 3/3

- Subdivision surfaces
  - Theory developed in 1970s and early '80s
  - Picked up by computer animation industry in late 1990s
  - Now replaced NURBS in computer animation
    - Solves one of the big problems of NURBS
  - Still under active research for use in CAD
    - Introduces new problems, not present in
      NURBS, which make it unsuitable for CAD in its present form
NURBS curve
- A curve is defined parametrically
- Its shape is determined by:
  - control points, \( P_i \)
  - and the NURBS basis functions, \( N_{i,k} \)

\[
P(t) = \sum_{i=1}^{n+1} N_{i,k}(t) P_i
\]

Basic properties of NURBS 1/3
- \( P(t) = \sum_{i=1}^{n+1} N_{i,k}(t) P_i \)
- The basis functions must sum to 1 to produce a valid new point

\[
\sum_{i=1}^{n+1} N_{i,k}(t) = 1, \quad t_{\min} \leq t \leq t_{\max}
\]

Basic properties of NURBS 2/3
- The basis functions are calculated from a knot vector
- Just a non-decreasing sequence of real numbers
  - e.g. [0,0,0,1,1,1] or [1,2,3,4,5,6]
  - or [1, 2, 3, 4, 5, 6, 7, 2, 15, 6]
- See lecture notes or Rogers & Adams for details

Basic properties of NURBS 3/3
- If the basis functions are \( C_m \)-continuous at \( t_r \), then \( P(t) \) is guaranteed to be \( C_m \)-continuous at \( t_r \)
- So continuity depends only on the basis functions, \( N_{i,k} \)
- Continuity does not depend on the locations of the control points
  - you can sometimes get extra continuity by careful positioning of control points

NURBS surface
- A bivariate generalisation of the univariate NURBS curve

Curve
\[
P(t) = \sum_{i=1}^{n+1} N_{i,k}(t) P_i
\]

Surface
\[
P(s,t) = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} N_{i,k}(s) N_{j,k}(t) P_{i,j}
\]

The big constraint...
- NURBS surfaces require a quadrilateral mesh of \((m+1)\times(n+1)\) points
The first problem
- Very few objects are made up of a single rectangular patch, so we need to join patches together

The second problem
- What do we do at special points where other than four patches meet?
  - Either we cannot get $C^2$
    - Which means that curvature is not continuous
  - Or we get $C^2$ be forcing curvature to be zero
    - Which produces a flat spot
  - Or we get $C^2$ using very high degree patches
    - Which are very hard for a designer to control

Drawing a NURBS curve
- NURBS curves and surfaces are always drawn on a pixelated surface
- NURBS curves can be approximated by straight lines
  - So long as each straight line deviates from the curve by less than half a pixel

Drawing a NURBS surface
- NURBS surfaces are subdivided and drawn as a series of planar polygons
- Each polygon is only one or two pixels in area on the screen
- Shading algorithms are used to ensure that the surfaces appear to be smoothly curved

Subdivision surfaces
- Do away with the explicit parametric representation
- Base a curve or surface solely on its control points and their connectivity
- Provide a simple mechanism which produces a larger, more refined set of control points from the current set
- Iterate refinement until the appropriate level of detail is achieved

History of subdivision schemes
- A univariate (curve) scheme was described by de Rahm in 1947
- Rediscovered by Chaikin in 1974
- Extended to bivariate (surfaces)
  - Doo-Sabin bi-quadratic patches (1978)
  - Catmull-Clark bi-cubic patches (1978)
- Flurry of mathematical work in the early 1980s
  - Dyn & Levin at Tel Aviv University
Use of subdivision schemes
- Pixar picked up the ideas and tested them in Geri’s Game (1997)
- ...then discarded its NURBS based software in favour of subdivision schemes
  - NURBS
    - Toy Story 1995
    - A Bug’s Life 1998
  - Subdivision surfaces
    - Toy Story II 1999
    - Monsters Inc. 2001
    - Finding Nemo 2003

Subdivision basics
- An example: Catmull-Clark subdivision
  - Introduce new points
    - At face-centres
    - At mid-edges
  - Adjust positions of original points
  - Repeat until sufficiently detailed

Chaikin curve subdivision
- Underlies Doo-Sabin surface subdivision
- C1-continuous in the limit
- Essentially just a ¼-¾ rule

The maths of Chaikin
\[
P_{2i}^{n+1} = \frac{1}{3} P_i^n + \frac{2}{3} P_{i+1}^n
\]
\[
P_{2i+1}^{n+1} = \frac{1}{3} P_i^n + \frac{2}{3} P_{i+1}^n
\]
\[
h = [..., 0, 0, \frac{1}{3}, \frac{2}{3}, 1, 0, 0, ...]
\]
\[
P^n = [..., P_0^n, P_1^n, P_2^n, ...]
\]
\[
P^{n+1} = h \ast P^n
\]

The limit curve
- It can be shown that the limit curve of the Chaikin scheme is the uniform quadratic B-spline, which is guaranteed to be C1
- When drawing curves in computer graphics, we draw a set of straight lines, so only need to subdivide until each segment is about a pixel long and we have a good enough approximation to the curve

C2 approximating scheme
- Underlies Catmull-Clark surface subdivision
- Can be described as: “Insert a midpoint and adjust the old control points”
The maths of the C2 scheme

\[ P_{2n+1}^{n+1} = \frac{1}{2} P_{2n+1}^n + \frac{1}{2} P_{2n}^n + \frac{1}{2} P_{2n-1}^n \]

\[ P_{2n+1}^{n+1} = \frac{1}{2} P_{2n}^n + \frac{1}{2} P_{2n+1}^n \]

\[ P_{2n+1}^{n+1} = h \ast P^n \]

\[ h = [\ldots, 0, 0, \frac{1}{2}, \frac{1}{2}, 0, 0, \ldots] \]

\[ P^n = [\ldots, P_0^n, P_1^n, P_2^n, \ldots] \]

\[ \overrightarrow{P^n} = [\ldots, P_1^n, 0, P_2^n, 0, P_3^n, 0, \ldots] \]

\[ P_{2n+1}^{n+1} = h \ast P^n \]

Why this notation?

- Easy to analyse
- Allows use of the z-transform

\[ h = [\ldots, h_0, h_1, h_2, \ldots] \]

\[ h(z) = \ldots + h_0 z^0 + h_1 z^1 + h_2 z^2 + \ldots \]

\[ P_{2n+1}^{n+1} = h \ast P^n \]

\[ P_{2n+1}^{n+1}(z) = h(z) \times P^n(z^2) \]

The analysis tools

- Generating function formalism
  - Use the z-transform on the kernel, \( h \)
  - Provides sufficient conditions for continuity
    - Essentially checks that the differences between adjacent points decrease fast enough at each refinement step to produce a smooth curve
- There is also a matrix formalism
  - Analyse stationary points
  - Provides necessary conditions for continuity
- For details see our research papers 😊

Useful subdivision kernels

\[ h = \frac{1}{4} [1, 3, 3, 1] \]

- C1, approximating, limit curve is quadratic B-spline

\[ h = \frac{1}{8} [1, 4, 6, 4, 1] \]

- C2, approximating, limit curve is cubic B-spline

\[ h = \frac{1}{16} [1, 0, 9, 16, 9, 0, -1] \]

- C1, interpolating, “four-point scheme”
- There is also a C2 interpolating six-point scheme

From Chaikin to Doo-Sabin

- Doo-Sabin scheme is bivariate generalisation of Chaikin \( \frac{1}{4} - \frac{3}{4} \) scheme

Extraordinary polygons

- Need special co-efficients for these

\[ \alpha_0 = \frac{1}{4} + \frac{5}{4K} \]

\[ \alpha_1 = \frac{1}{4K} (3 + 2 \cos \frac{3\pi}{K}) \]

(Doo-Sabin)
Catmull-Clark subdivision

- Catmull-Clark is based on the \( \frac{1}{8}[1,4,6,4,1] \) univariate scheme

Catmull-Clark rules

- This is easy: the rules are simply the tensor product of the univariate \( \frac{1}{8}[1,4,6,4,1] \) rules.

Catmull-Clark special cases

- This is more difficult: we need to find coefficients which maintain continuity
- It is only possible to get C1 continuity at these extraordinary points.

\[
\alpha = \frac{1}{3n^2} \\
\beta = \frac{1}{2n^2} \\
\gamma = 1 - \frac{2}{3n^2}
\]

Extraordinary polygons: disappear after one step
Extraordinary vertices: remain in the mesh

Subdivision vs NURBS

- Extraordinary points
  - Subdivision handles them easily
  - NURBS requires the use of other types of surface to fill in the holes
- Memory requirements
  - Subdivision needs a lot (many MB)
  - NURBS is very compact
- Artifacts
  - Some artifacts present in both
  - Subdivision has extra artifacts

The future

- Computers now have enough memory to handle subdivision easily
- Subdivision now standard for computer animation
- NURBS still standard for CAD
- Subdivision will eventually replace NURBS for CAD if we can sort out the artifact problems

Our work at Cambridge

- Univariate schemes that are not binary
  - Ternary (\( \times 3 \)) schemes
  - Sesquinary (\( \times 1\frac{1}{2} \)) schemes
- Towards a bestiary of bivariate schemes
  - Classification & analysis of all schemes
  - Identification & analysis of new schemes (especially ternary)
- Geometrically-sensitive subdivision
  - Modifying existing schemes to take account of geometric relationships