Lecture 10

Curves and Surfaces Concluded
3-D Graphics Data Structures

Friday, February 18, 2000

William H. Hsu
Department of Computing and Information Sciences, KSU
http://www.cis.ksu.edu/~bhsu

Readings:
Sections 11.3, 12.1-12.5, Foley *et al*
(Reference) 9, 10.1, 10.6-10.13, 10.16-10.17 Hearn and Baker 2[e]
Slide Set 5, VanDam (8b, 11/09/1999)

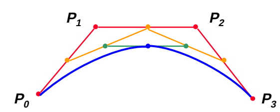
Lecture Outline

- Readings
  - Sections 11.3, 12.1-12.5, Foley *et al*
  - Optional reference: Chapter 9, 10.1, 10.6-10.13, 10.16-10.17, Hearn and Baker 2[e]
- Quick Review: Properties of Cubic Curves and Splines
  - Splines: *B*- (UN, NUN, NUR ≡ NURBS), Beta- (β-), Catmull-Rom, Kochanek-Bartels
  - Uniformity, rationality, continuity, control point and polygon properties
- Interpolating Cubic Curves and Surfaces
  - deCasteljau's algorithm for curves
  - Bicubic surface interpolation (concluded)
- 3D Graphics Data Structures
  - This time: boundary representations (*aka* B-reps)
  - Next time: spatial partitioning representations
- Visible Surface Determination (VSD): Introduction
  - Role of graphics data structures in VSD (more later)
  - Graphics data structures in computational geometry
- Next Lecture: Basics of Constructive Solid Geometry (Survey)

Comparison of Cubic Curves

- Hermite
  - Blend 4 functions; no CP; full interpolation; C^1 and G^1 with constraints; fast
- Bézier
  - Convex CP; interpolate 2 of 4 control points; C^1 and G^1 with constraints; fastest
- *B*-splines
  - Uniform, nonrational
    - Convex CP, 4 points each, no interpolation; C^2 and G^2; medium
  - Nonuniform, nonrational
    - Convex CP, 5 points each, "no interpolation"; "up to" C^2 and G^2; slow
  - Nonuniform, rational
    - Convex CP, 5 points each, "no interpolation"; rational; "up to" C^2 and G^2; slow
- Beta Splines (β-Splines)
  - Convex CP; 6 points to control curve (4 local points, 2 global); C^1 and G^2; medium
- Catmull-Rom Splines
  - General CP; interpolate or approximate 4 points per CP; C^1 and G^1; medium
- Kochanek-Bartels Splines
  - General CP; interpolate 7 points per CP; C^1 and G^1; medium

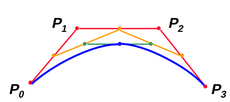
Interpolating Curves [1]: Recursive Subdivision



- Intuitive Idea
  - Given
    - Curve (Bézier or uniform B-spline) defined using control polygons (CPs)
    - 4 control points P_0, P_1, P_2, P_3
  - Problem: can't get quite the right curve shape (not enough control points)
  - Solutions: *increase degree of polynomial segments* OR *add CPs*
  - Technique: recursive subdivision algorithm
    - Add control points by splitting existing CP up recursively
    - Compute CPs for left curve L_0, L_1, L_2, L_3, right curve R_0, R_1, R_2, R_3
    - Stop when variation (curve-to-control point distance) is low enough
    - Purpose: *display curve* OR *allow new control points to be manipulated*

Interpolating Curves [2]: deCasteljau's Algorithm

- Recursive Subdivision Algorithm for Interpolation [deCasteljau, 1959]
  - Purpose: *display curve* OR *allow new control points to be manipulated*
  - Display: fast and cheap (see below)
- Properties
  - Cheap: can implement using subdivision matrices (Equations 11.52, 11.53, FVD)
  - Fast: rapid convergence due to…
  - Variation-diminishing property
    - Monotonic convergence to curve
    - Holds for all splines with convex-hull CPs
- When Does It Work?
  - Uniform splines (uniformly-spaced knots)
    - Q: Can we subdivide NURBS?
    - A: Yes, by adding knots (expensive) [Böhm, 1980; Cohen *et al*, 1980]
  - Alternative approach: *hierarchical B-splines* [Forsey and Bartels, 1988]



Quick Review: Bicubic Surfaces

Quick Review: Interpolating Bicubic Surfaces

Paper Reviews [1]: General Information

- 3 of 4 (Assigned) Reviews Required
  - All reviews worth 15% of course grade
  - Choose 3 of 4 (*may* have > 1 choice on some) or write all 4
  - Lowest dropped (each of remaining 3 worth 50 of 1000 points)
- General Objectives
  - Compare, evaluate CG techniques (synthesis, processing, visualization)
  - Guidelines: next (suggested topics, tools to appear on CIS 736 course web page)
- Review Topics
  - Modeling, rendering, animation, information visualization
  - Selection criteria: target length 10 pages; no more than 15 pages
- Logistics
  - Papers will be available online (and at 17 Seaton Hall) *next week*
  - Send to CIS 736 GTA (Songwei Zhou) at cis736ta@ringil.cis.ksu.edu
  - Turn in by midnight of due date (no late reviews)
  - Get back commented reviews in electronic form

Paper Reviews [2]: Specific Objectives

- Modeling
  - "The right *representation* is half the battle"
  - "Graphics database formats + rendering / animation algorithms = CG programs"
- Rendering
  - Image synthesis: aspects of realism
  - "The right tool for the right job"
- Animation
  - What's beneficial, what's overkill?
  - What's easy, what's hard?
- Information Visualization
  - How to avoid "saying nothing" and "telling lies" with graphs
  - How to maximize information, not "ink" (screen / disk usage, etc.)
- Overall: Be Able To
  - Justify using CG technique *X* in scenario *S*
  - Select and develop appropriate (practical) CG techniques

Paper Reviews [3]: Do's and Don'ts

- Do
  - Use typical
    - Font (Times, Arial, etc.), type size (10-12 point), spacing (single), margins
    - Length (1-2 pages)
  - Cite your sources
  - Use spelling and grammar checkers (and check carefully by hand)
  - Write in complete sentences and your own words
  - Discuss paper
    - Significance, audience
    - Pros, cons (*Does CG method meet objectives? Why or why not?*)
    - Applications you would like to see in future work
    - Open (unanswered) questions! (Read carefully…)
- Don't
  - Merely
    - Quote paper, authors, bibliographic references, or other reviews
    - Summarize content of paper without evaluation and discussion
  - Critique without justification ("*This paper was {bad | vague | great}.*")

Terminology

- Interpolation versus Approximation (Section 10.6, Hearn and Baker)
  - Interpolation: fit curve *through* specified points
  - Approximation: fit curve *to* control path (without necessarily passing through)
- deCasteljau's Algorithm: Recursive Subdivision Algorithm for Interpolation
- Bicubic Surfaces
  - Types: Hermite (11.3.1 FVD), Bézier (11.3.2 FVD), B-splines (11.3.3 FVD)
  - Coons patch: generalization of Hermite patch form to arbitrary boundary curves
- 3D Graphics Data Structures
  - Regularized Boolean set operations: $\cup^*$, $\cap^*$, $-^*$ (12.2 FVD)
  - Primitive instancing: parameterized object-like 3D solid representation (12.3 FVD)
  - Sweep representations: objects moved along trajectory define others (12.4 FVD)
  - Boundary representations *aka* B-reps: vertex, edge, face descriptions (12.5 FVD)
    - Polyhedra: solid bounded by poygons, satisfying Euler's formula (12.5.1 FVD)
    - Winged edge: vertex-edge-face data structure (12.5.2 FVD)
    - Composition of B-reps using Boolean set operations (12.5.3 FVD)

Summary Points

- Quick Review: Properties of Cubic Curves and Splines
- Interpolating Cubic Curves and Surfaces
  - deCasteljau's algorithm for curves – 11.2.7 FVD
  - Bicubic surface interpolation (concluded) – 11.3.5 FVD
- 3D Graphics Data Structures (Chapter 12, FVD)
  - Representing solids – 12.1 FVD
  - Regularized Boolean set operations, primitive instancing, sweep representations
  - Boundary representations (*aka* B-reps) – 12.5 FVD
    - Polyhedra, winged edge (Mantyla)
    - Composition of B-reps using Boolean set operations
- Role of Graphics Data Structures in Visible Surface Determination (VSD)
- Next Lecture
  - Spatial partitioning representations – 12.6 FVD
    - Cell decomposition
    - Quadtrees and octrees
  - Basics of Constructive Solid Geometry (CSG)