# Lecture 9

## More on Curves and
## Parametric Bicubic Surfaces

**Wednesday, February 16, 2000**

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

Readings:
Sections 11.1-11.3, Foley *et al*
(Reference) Sections 10.1-10.13, Hearn and Baker 2$^e$

---

# Lecture Outline

- Readings
  - Sections 11.1, 11.2.10, 11.3-11.5, Foley *et al*
  - Optional references: Sections 10.1-10.13, Hearn and Baker 2$^e$
- Quick Review: Cubic Curve Representations
  - Polygon meshes and parametric cubic curves
  - Hermite and Bézier curves
- More on Cubic Curves
  - Splines: *B*- (UN, NUN, NUR ≡ NURBS), Beta- (β-), Catmull-Rom, Kochanek-Bartels
  - Interpolation by subdivision
  - Properties of some cubic curves
    - Uniformity, rationality
    - Continuity: $C^0$, $C^1$, $C^2$
    - Interpolation, number, and geometry of control points
- Bicubic Surfaces
  - Types: Hermite, Bézier, *B*-spline
  - Interpolation and other issues (shading, etc.)
- Next Lecture: 3D Graphics Data Structures

---

# Quick Review:
# Surface Representations

- Polygon Meshes (11.1, Foley *et al*; 10.1, Hearn and Baker)
  - Idea: use polygon functions (rendering primitives) for modeling objects
  - Collection of in graphics kernel (e.g., PHIGS, OpenGL)
    - Examples: triangle strip, quadrilateral mesh (see 10.1, Hearn and Baker)
    - Popular in modern 3-D video games (first- / third-person; flying / shooting)
  - Data structures: same as used in computational geometry (see 11.1, FVD)
- Parametric Cubic Curves
  - Idea: define smooth curves *parametrically* using *specification* and *interpolation*
  - Specification: control polygons
  - Interpolation: fitting piecewise polynomial curve segments, $x = x(t)$, $y = y(t)$
  - Alternative functional representations
    - Explicit: $y = f(x)$ (problems - ellipses, closed curves?)
    - Implicit: $f(x, y) = 0$ (problems - multiple solutions, tangent discontinuity)
- Tradeoffs: *When to Use Which?*
  - Polygon meshes: can be computationally less demanding, but “rougher”
  - Parametric cubic curves: “smoother” if computational effort invested

---

# Properties of Cubic Curves:
# Definitions

- Representation: $Q[t] = [x(t)\ y(t)\ z(t)]$
  - Polynomial (here, cubic) system: Equations 11.5-11.6, FVD
  - Matrix of coefficients $C$
- Continuity
  - Two curve segments join together: $G^0$ geometric continuity
  - Directions (not necessarily magnitudes) of tangent vectors equal at join point: $G^1$ geometric continuity
  - Tangent vectors equal at join point: $C^1$ continuity (camera analogy)
  - $n$th derivative of system ($d^n/dt^n\ [Q(t)]$) equal at join point: $C^n$ continuity
  - Exercise: when does $C^1$ continuity not necessarily imply $G^1$? (Figure 11.10, FVD)
- Uniformity
  - Knot: join point between segments of *piecewise cubic* curve
  - Uniform: knots spaced at equal intervals
- Rationality
  - Rational: $x(t)$, $y(t)$, $z(t)$ each defined as ratio of two cubic polynomials
  - Can define in *homogeneous coordinates*: see Section 11.2.5, FVD

---

# Quick Review:
# Hermite Curves

- Definition
  - Curve defined in terms of piecewise cubic segments
  - Basis matrix: $M_H$ (Equation 11.19, FVD)
    - System of (3) cubic polynomials: $Q(t) = [x(t)\ y(t)\ z(t)\ 1]$ (Equation 11.9, FVD)
    - Derivation: Section 11.2.1, FVD (Equations 11.12-11.19)
- Distinguishing Characteristics
  - Direct specification of curves, blended to form target curve; *no control points*
  - Inherently $C^0$ and $G^0$ continuous (Why? First of all, $C^0 \equiv G^0$)
- Pros
  - Easy to get $C^1$ and $G^1$ continuity (How? See constraints: Equation 11.22)
  - Easy to display: evaluate Equation 11.5 FVD (i.e., $Q(t) = T \bullet M \bullet G$) at $n$ successive values of $t$
  - Nice interactive representation: good for graphical front-ends
- Cons
  - Computing blended curve: good but suboptimal subdivision procedure
  - See Section 11.2.7, FVD; Section 10.13, Hearn and Baker

---

# Quick Review:
# Bézier Curves

- Definition
  - Another piecewise cubic curve
    - Defined indirectly
    - Control points: 2 on curve, 2 not on curve
  - Basis matrix
    - $M_B = M_H \bullet M_{HB}$ (Equation 11.28, FVD)
    - Derivation: Section 11.2.2, FVD (Equations 11.25-11.28)
- Distinguishing Characteristics
  - Indirect specification of curves; convex control polygon
  - Inherently $C^0$ and $G^0$ continuous; easy to get $C^1$ and $G^1$ continuity
- Pros
  - Combinatorially simple basis functions (Bernstein polynomials)
  - Easy to convert from Hermite! (11.2.2, FVD; 10.12, Hearn and Baker)
- Cons
  - Not as intuitively manipulable as Hermite (see Figure 11.23, FVD)

## Uniform, Nonrational B-Splines

- **Definition**
  - Locally controlled model (true of all *B*-splines *aka* basis splines)
    - Definition: polynomial coefficients depend on *few control points*
    - Result: *very smooth* but (one hopes) not too slow
  - Basis matrix ("B" stands for "basis")
    - $M_{Bs}$ (Equation 11.34, FVD)
    - Derivation: Section 11.2.3, FVD (Equations 11.32-11.34)
- **Distinguishing Characteristics**
  - Uniform (spacing of knots), nonrational (not expressed as ratio of equations)
  - *No interpolation* (true of *B*-splines in general except for specific cases)
- **Pros**
  - *Flexible, most smooth*: inherently $C^2$ and $G^2$ continuous
  - Speed through uniformity
  - Easy to convert to Hermite, Bézier for *display* vs. design (10.12, Hearn and Baker)
- **Cons**
  - Curve "must" be smooth (can't *reduce* continuity)

---

## Nonuniform, Nonrational B-Splines

- **Definition**
  - Another locally controlled model
  - *NB*: not described using single set of blending functions
    - Easier to write down recursive definition
    - Recurrence: Equation 11.44, FVD
- **Distinguishing Characteristics**
  - Nonuniform (spacing of knots): purpose - to increase flexibility of model
  - Nonrational (not expressed as ratio of equations)
  - *No interpolation* (except with reduced continuity); 5 control points
- **Pros**
  - *Most smooth*: inherently $C^2$ and $G^2$ continuous
  - Even more flexible *in design*: able to reduce to $C^1$ to $C^0$ (CP interpolated) to none
- **Cons**
  - Can have repeated knot values (multiple knots)
  - *Very slow to converge* with enough segments (true for all nonuniform)

---

## Nonuniform, Rational B-Splines (NURBS)

- **Definition**
  - Yet another locally controlled model
  - Nonuniform, rational polynomial curve segments
    - Generalizes over arbitrary piecewise polynomial curves
    - Segment = *B*-spline $\Rightarrow$ NURBS
  - Rational form in homogenous coordinates (HC): Equation 11.45, FVD
- **Distinguishing Characteristics**
  - Nonuniform (spacing of knots)
  - Rational
    - Trivial conversion: add $W(t) = 1$ to get HC representation
    - Compare: NUR Hermite, Bézier
  - *No interpolation*; 5 control points (see Figures 11.28 and 11.29, FVD)
- **Pros**
  - *Most smooth*: inherently $C^2$ and $G^2$ continuous
  - Very flexible, popular (despite computational complexity)
- **Cons**
  - *Very slow to converge* with enough segments (true for all nonuniform)

---

## Other Splines

- **Beta ($\beta$-Splines)**
  - Purpose: surface design (CAD)
  - Distinguishing characteristics: 2 parameters ($\beta_1$, $\beta_2$), geometry as for *B*-spline, convex control polygon; partly local (4 points per CP, 2 global)
  - Pros: further control over shape (see basis matrix: $M_\beta$ - Equation 11.48, FVD)
  - Cons: can be somewhat computationally intensive (uniform but $M_\beta$ more complex)
- **Catmull-Rom (*aka* Overhauser)**
  - Purpose: for animating motion - mouse trajectory, camera in 3D, etc. (*Coming soon to a homework near you!*)
  - Distinguishing characteristics: local control, interpolation / approximation
  - Pros: smooth transitioning (see basis matrix: $M_{CR}$)
  - Cons: another tradeoff (need speed); not fastest, but much faster than NURBS
- **Kochanek-Bartels**
  - Purpose: controlling animation
  - Distinguishing characteristic: similar to Hermite form
  - Pros: fast (but not fastest)
  - Cons: another tradeoff; good interface

---

## 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 ($\beta$-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

---

## 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 New Roman, Arial, etc.), type size (10-12 point), spacing (single)
    - 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

- Cubic Curve Representations
  - Polygon meshes: using many polygons to represent 3D surface
  - Parametric cubic curves: Hermite, Bézier, splines
  - Curve properties
    - Uniformity: knots (*aka* join points) spaced at even intervals
    - Rationality: segment expressible as ratio of polynomial parametric functions
    - Continuity: geometric ($G^0$, $G^1$, $G^2$); differentiability ($C^0$, $C^1$, $C^2$)
  - Splines: smooth parametric cubic curves
    - *B*- (UN, NUN, NUR ≡ NURBS): locally controlled, non-interpolative
    - Beta- (β-): semi-locally controlled, non-interpolative
    - Catmull-Rom: for smooth, fast camera animation
    - Kochanek-Bartels: for smooth, fast object animation
  - Control polygon: “closed” curve consisting of control points
  - Convex hull: smallest convex region defined by set of points
- Interpolation by Subdivision
- Bicubic Surfaces: Expressed as Patches (4 Cubic Curves)

---

# Summary Points

- Cubic Curve Representations (Concluded)
  - Polygon meshes and parametric cubic curves
  - Hermite and Bézier curves
  - Splines: *B*- (UN, NUN, NUR ≡ NURBS), Beta- (β-), Catmull-Rom, Kochanek-Bartels
- Interpolation by Subdivision
- Properties
  - Uniformity, rationality
  - Continuity: $C^0$, $C^1$, $C^2$
  - Interpolation, number, and geometry of control points
- Implementing Bicubic Surfaces using Parametric Curves
- Next Class: 3D Graphics Data Structures
  - Read or review polygon meshes (11.1 FVD)
  - Chapter 12 FVD: lead-in to (basics of) solid modeling for CAD / CAM
  - Read about boundary representations (B-reps), spatial partitioning