# Interactive NURBS Tutorial in Virtual Reality

**Eng 477 Project Fair**
October 10, 2006

Edmond Nadler
enadler@umich.edu
(734) 913-6970

# NURBS Curves and Surfaces

= **N**on-**U**niform **R**ational **B**-**S**plines

- Fundamental
- Pervasive, Standard
- Enable creation of wide variety of free-form curves and surfaces

*but…*

Often not well understood.

Interactive Virtual Reality tutorial illustrating features and properties of NURBS curves and surfaces would be interesting and useful.

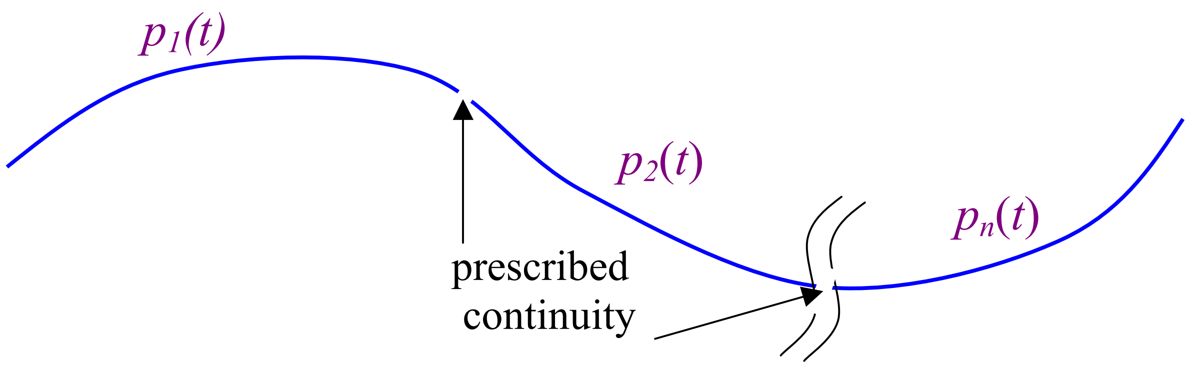
# What are NURBS?

A convenient, efficient and geometric representation for piecewise polynomial and rational curves and surfaces.

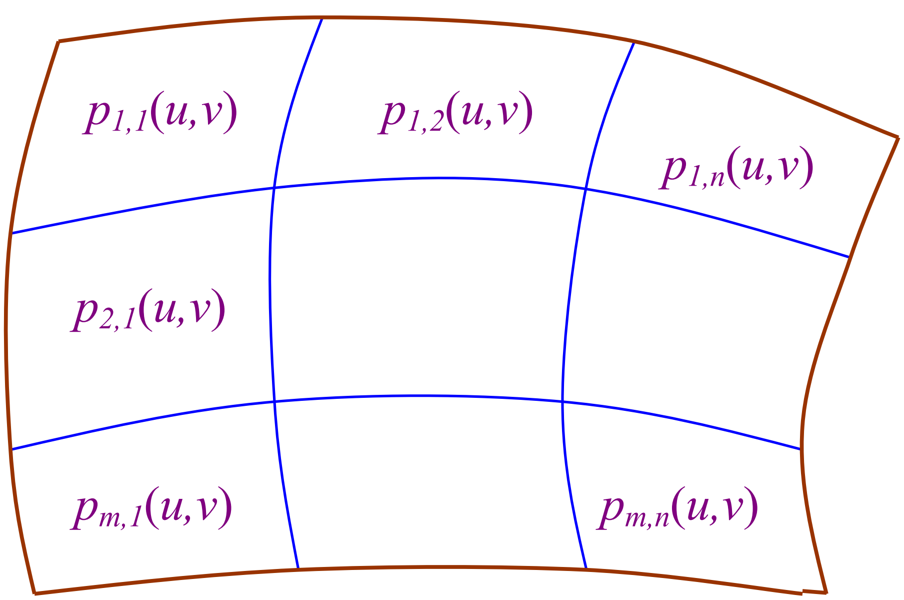
rational = polynomial / polynomial

needed, for example, for exact representation of conic sections

Curve:




Surface:



# NURBS Curve

## Polynomial

$$\vec{c}(t) = \sum_i \vec{a}_i B_i(t)$$

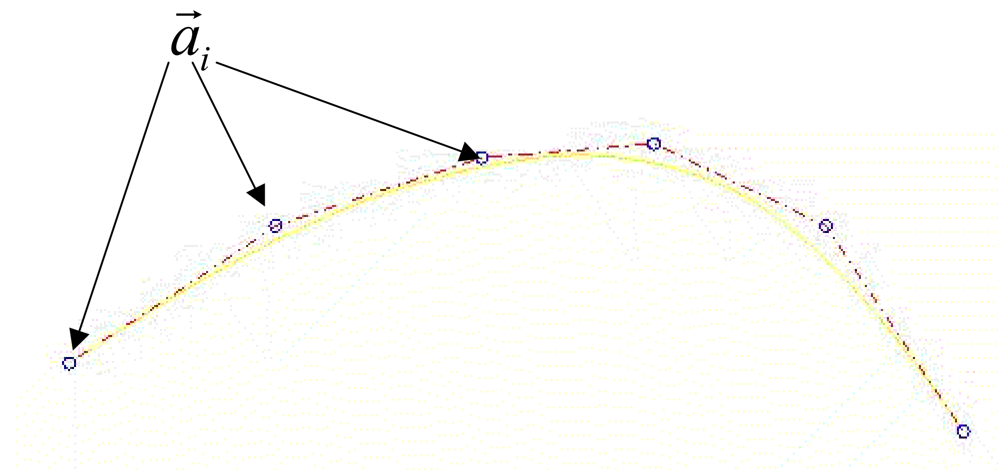
*with* $\vec{a}_i$ *Control Points*, vertices of the *Control Polygon*

$B_i(t)$ *B-Spline Basis Functions*

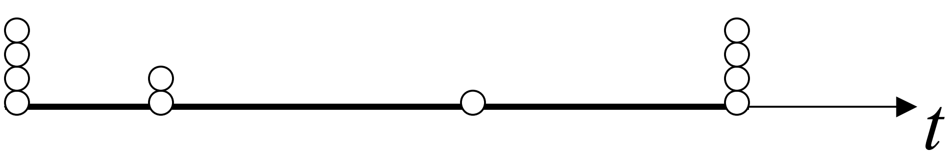
**Rational:** Add scalar *weights* $w_i$ for each Control Point:

$$\vec{c}(t) = \frac{\sum_i w_i \vec{a}_i B_i(t)}{\sum_i w_i B_i(t)}$$

Reduces to polynomial case if all $w_i$ equal.




$B_i(t)$ defined by *Polynomial Degree* and *Knot Sequence*:




*Multiplicity* of knots controls *continuity*

# NURBS Surface

Now, 2 sets of B-Spline Basis Functions

$$\{ B_i(u) \} , \ \{ C_j(v) \}$$

each governed by its own Polynomial Degree & Knot Sequence

## Polynomial

$$\vec{s}(u,v) = \sum_{i,j} \vec{a}_{i,j} B_i(u) C_j(v)$$

*with* $\vec{a}_{i,j}$ a 2-D array of Control Points, which are the vertices of the *Control Net*

**Rational:** As for curves, add scalar *weights* $w_{i,j}$ for each Control Point:

$$\vec{s}(u,v) = \frac{\sum_{i,j} w_{i,j} \vec{a}_{i,j} B_i(u) C_j(v)}{\sum_{i,j} w_{i,j} B_i(u) C_j(v)}$$

Again, reduces to polynomial case if all $w_{i,j}$ equal.

# Interactive NURBS Tutorial in VR:
## Some things it could include…

- Visual creation and/or manipulation of NURBS curves and surfaces in VR
    - Control Points $\vec{a}_i, \vec{a}_{i,j}$
    - Weights $w_i, w_{i,j}$
    - Basis Functions $B_i(\ )$ as determined by *Degree* and *Knot Sequence* – show effect on continuity when knots coalesce

- Evaluation of NURBS
    - Curve: Move $t$ on $[0,1]$ – See point $\vec{c}(t)$ on curve
    - Surface: Move $(u,v)$ on $[0,1]\times[0,1]$ – See pt $\vec{s}(u,v)$ on surface

      Move $u$ on $[0,1]$ – See curve $\vec{s}(u,\bullet)$ on surface

      Move $v$ on $[0,1]$ – See curve $\vec{s}(\bullet,v)$ on surface
    - Visualize geometric algorithm (de Boor): iterated convex combinations of Control Points

- Demonstrate NURBS Properties
    - Locality – region of influence of each Control Point
    - Convex Hull – point in C.H. of relevant Control Points
    - Variation Diminishing for Curves

- Other ideas…

# References and comments

I would be happy to help interested students to learn about NURBS.

I recommend "The NURBS Book" by Les Piegl and Wayne Tiller, Springer 1995. The first four chapters provide good introduction. Tiller is a leading expert on NURBS, and has developed a NURBS software library (Nlib), for which many of the algorithms are described in the later chapters of the book. I have used this book and related software library in developing geometry software in industry.

Other possible books are:
"Fundamentals of Computer Aided Geometric Design", by J. Hoschek and D. Lasser, A.K. Peters, 1993
"Curves and Surfaces for Computer Aided Geometric Design", by Gerald Farin, Academic Press, 1990 (there are more recent editions)
Also, many numerical analysis textbooks, such as "N.A.: Math. of Scientific Computing" by D. Kincaid and W. Cheney, Brooks/Cole 2002 will have material on splines in their sections on interpolation and approximation.
Note: Each book has a slightly different approach and notation.

For MATLAB users, the Spline Toolbox for MATLAB written by spline pioneer Carl de Boor (winner of the prestigious National Medal of Science) is a good hands-on way to learn and play with these things. Note: the Spline Toolbox does not use the terms NURBS, nor does it have rationals or surfaces (at least directly), at least in an older version.

* * *

The project should include at least the visual creation and/or manipulation of NURBS curves and surfaces, with the ability to drag control points and see the effect on the curve or surface. Adjustment of weights would be good. Time permitting, it would also be nice to be able to adjust the knots and see the effect. (In the case of surfaces there would of course be two such sets.) For this, perhaps the set of B-Spline Basis Functions corresponding to the current knot sequence could be shown on plots.

Additional items of interest would relate to *evaluation* of NURBS and demonstration NURBS *properties*, as outlined on the previous page. The Variation Diminishing Property mentioned there states that any plane will intersect a NURBS curve no more times than it does its Control Polygon.

–Ed Nadler