Interactive Graphics Systems

Non-uniform rational B-spline

concepts and practice

v1.0 20221019

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Alexandre Valle | DEI | FEUP

Pictures: two details of HCI from movie T*he Dark Knight (credits BLIND, Ltd.)*

Concepts

Spline

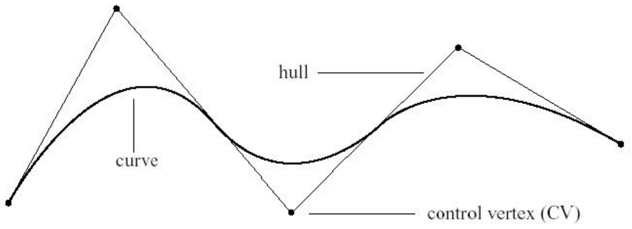
A special function defined piecewise by polynomials.

Linear Splines (1st degree)

- The most straight-forward way of drawing a curve is by connecting a sequence of points.
- The resulting curve is a linear spline and is equivalent to a polygon.
- There are 2 major drawbacks to this method of producing a curve.
    - In order to produce anything that actually appears curved, you would need a large number of points. Storing and computing all those points is not an efficient use of the computer's resources.
    - Manipulating a curve created in this fashion is very cumbersome because, once a point is moved, you lose the smoothness of the shape.

Higher degree splines (2nd, 3rd… degree)

- The way around the jaggedness produced by linear connectivity is through a series of blending functions.
- The blending functions generate smooth connection between the control vertices (CV) of the curve.
- A spline curve generates a smooth transition between its CV through a blending function that operates on these points.
- The set of CVs controlling the curve is referred to as the "hull".



(source https://www.derivative.ca/wiki088/index.php?title=Spline)

Concepts

Nurbs

Stands for:

- **N**on-**U**niform: uniformity controlled by knots values (can be non-uniformly spaced).
- The "**R**" in NURBS stands for rational and indicates that a NURBS curve has the possibility of being rational (later explained).
- **B-S**pline: or basis spline (function that has minimal support with respect to a given degree, smoothness, and domain partition).

The NURBS evaluation is a formula that uses basis spline functions which feed on input parameters: <u>degree</u>, <u>control points</u>, and <u>knots</u>.

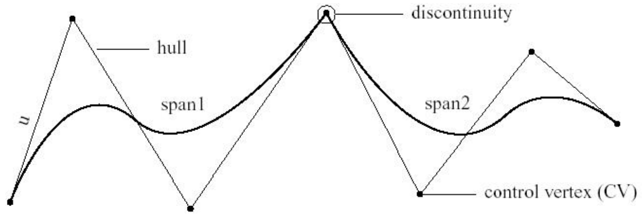
(sources:
https://www.derivative.ca/wiki088/index.php?title=Spline,
http://developer.rhino3d.com/guides/opennurbs/nurbs-geometry-overview/)

Concepts

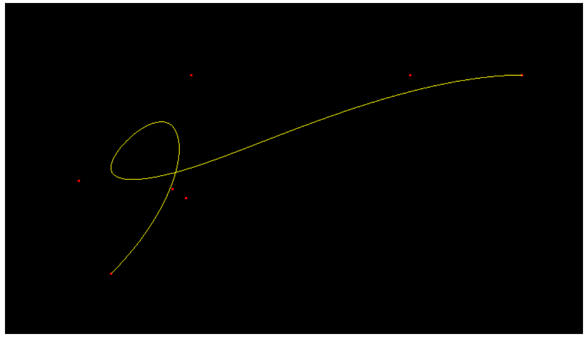
Sources: download and execute nurbs and Bezier examples at https://nccastaff.bournemouth.ac.uk/jmacey/RobTheBloke/www/opengl_programming.html#3

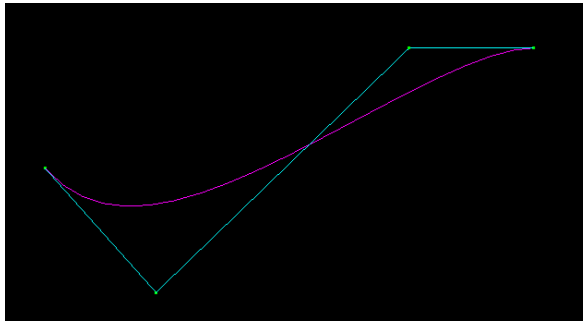
Nurbs and Bezier curves

- Both are piecewise curves made of a number of connected curve segments.
- Differ in the level of continuity at the points where the curve segments touch.
- A NURBS curve will typically be very smooth at these joints (the higher the degree of the blending function, the smoother the connection).
- Bézier curves have a discontinuity every **degree plus one points**.

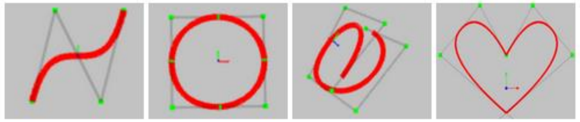


Source: https://www.derivative.ca/wiki088/index.php?title=Spline





https://nurbscalculator.in



Concepts

Degree

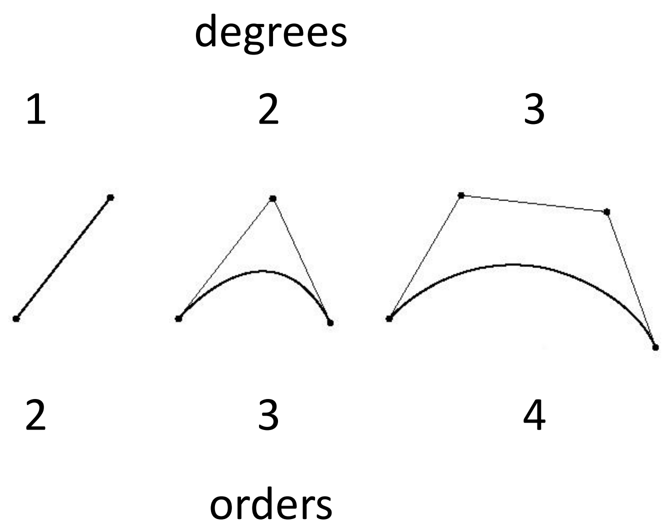
(http://developer.rhino3d.com/guides/opennurbs/nurbs-geometry-overview/)

- The degree of the spline is given by the degree of the underlying blending functions. It is a positive whole number.
- This number is usually 1, 2, 3 or 5, but can be any positive whole number.
- NURBS lines are usually degree 1,
- NURBS circles are degree 2, and most free-form curves are degree 3 or 5.
- Sometimes the terms linear, quadratic, cubic, and quintic are used.
- Linear means degree 1, quadratic means degree 2, cubic means degree 3, and quintic means degree 5.
- Cubic splines are usually sufficiently smooth and well behaved for most applications.

Order

(https://www.derivative.ca/wiki088/index.php?title=Spline)

- The "degree plus one" formulation is often referred to as the <u>order of the curve</u>.
- A cubic curve, for example, has a degree of three and, therefore, an order of four.



(adapted from source
https://www.derivative.ca/wiki088/index.php?title=Spline)

Concepts

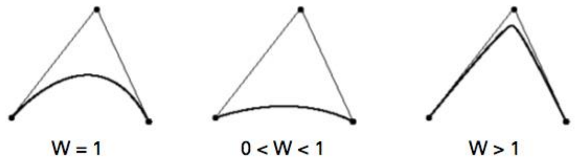
Control points (CP)

(http://developer.rhino3d.com/guides/opennurbs/nurbs-geometry-overview/, https://www.derivative.ca/wiki088/index.php?title=Spline)

- Each control point of the curve has X, Y, and Z coordinates that determine its position in world space.
- The control points are a list of at least degree + 1 points.
- The control points have an associated number called a weight (next section).

Rational / non-rational Spline

- Besides X,Y,Z coordinates, each control point has an additional fourth component, W.
- The W component determines a CP's weight. The weight determines the "pull" (like a magnet) of a CP on the spline curve.
- The value of the W component makes a spline rational or non-rational. A non-rational spline has only equal weights (typically, W=1), while a rational spline contains at least one different weight.
- With a few exceptions, weights are positive numbers. When a curve's control points all have the same weight (usually 1), the curve is called non-rational, otherwise the curve is called rational.



Practice
(requires WebCGF 3.0.0)

WebCGF: Parametric surface support

```
import {CGFnurbsSurface} from './lib/CGF/CGFnurbsSurface.js'
```

public class | source

CGFnurbsSurface

Defines a NURBS surface to be rendered using a CGFnurbsObject.

Constructor Summary

Public Constructor	
public	`constructor(degree1: Number, degree2: Number, controlPoints: Array)` Constructs a surface with the provided parameters.

Member Summary

Public Members	
public	`controlPoints: Array` List of control points, divided by U and V.
public	`degree1: Number` Degree in U.
public	`degree2: Number` Degree in V.

Practice
(requires WebCGF 3.0.0)

WebCGF: Curve rendering support

```
import {CGFnurbsObject} from './lib/CGF/CGFnurbsObject.js'
```

public class | source

CGFnurbsObject

Extends:
CGFobject → CGFnurbsObject

Defines a NURBS object that will be used to render a CGFnurbsSurface.

This class is based on the Parametric Surfaces Geometry class from THREE.JS by https://github.com/zz85 and http://prideout.net/blog/?p=44

Constructor Summary

Public Constructor	
public	`constructor(scene: CGFscene, uDivs: Number, vDivs: Number, evalObj: CGFnurbsSurface)` Creates the NURBS object.

Method Summary

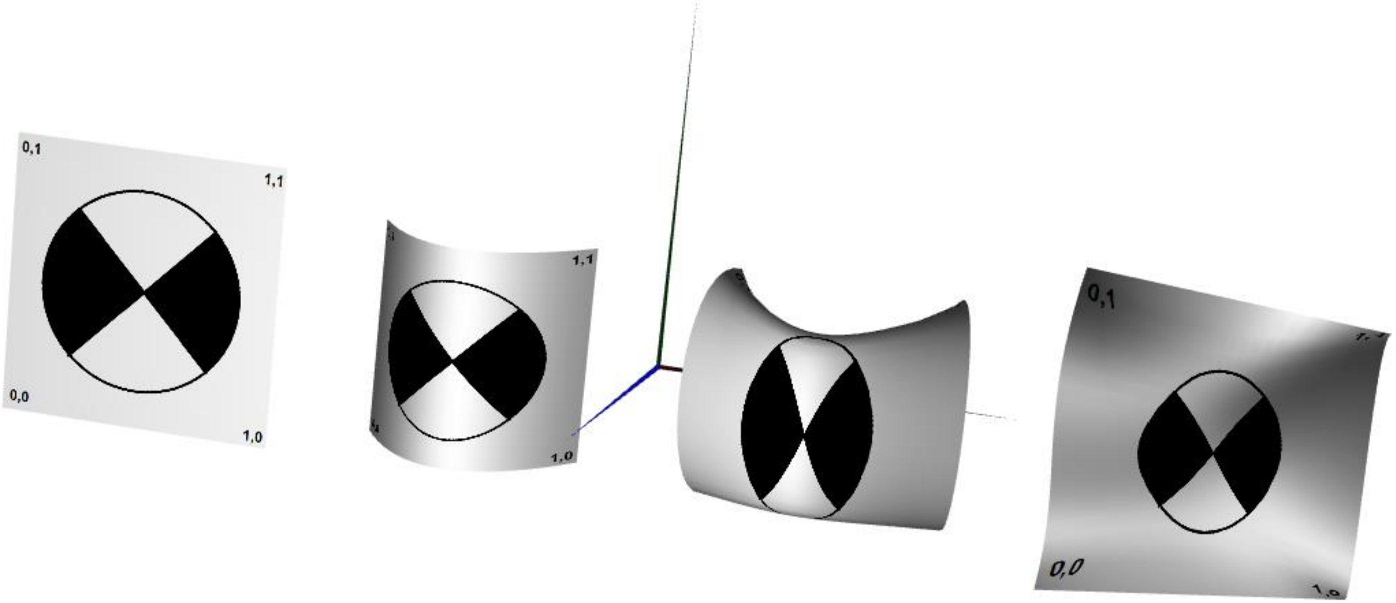
Public Methods	
public	`display()` This method should be called in the display function of the scene to render this object.
public	`initBuffers()` Initializes the buffer.

Practice: putting altogether

(requires WebCGF 3.0.0)

```
// degree on U: 2 control vertexes U
// degree on V: 2 control vertexes on V

var vertexes = [ // U = 0
                   [ // V = 0..1;
                     [-2.0, -2.0, 0.0, 1 ],
                     [-2.0,  2.0, 0.0, 1 ]
                   ],
                   // U = 1
                   [ // V = 0..1
                     [ 2.0, -2.0, 0.0, 1 ],
                     [ 2.0,  2.0, 0.0, 1 ]
                   ]
                 ]

// an object holding the surface representation and having a function getPoint(u, v)
var nurbsSurface = new CGFnurbsSurface(1, 1, vertexes);

// generate a 3D object with 20x20 vertexes based on a surface representation
var obj = new CGFnurbsObject(this, 20, 20, nurbsSurface );


…
obj.display()
```

Demonstration (live)



Source code available in NURBS demo at Moodle website