Analytical Geometry and Linear Algebra I, Visiting Lecture

Parametric form of segment line, polyline

Splines: Cubic, Bezier

Surfaces: made by extruding, shearing, revolving

Innopolis University

Disclaimer

Goal

The goal of this lecture is to get acquainted with splines, surfaces, their applications. Obtain some basic intuition.

Constraints

1. Only really necessary proofs (others can be found in reference material).
2. I show today topics only from practical perspective (how to use it as a user and a programmer, not as a creator of new algorithms).
3. I have to make a small recap of some topics due to the reason of your misunderstanding of some concepts.

Lecture Objectives

- To get the main benefit of parametric form.
- To have an intuition where and how splines can be used.
- To understand a relationship between splines and conic sections.
- How to make a surfaces using curves.


THE ADIDASAURUS REX WAS BORN

Computer Aided Design

Form types

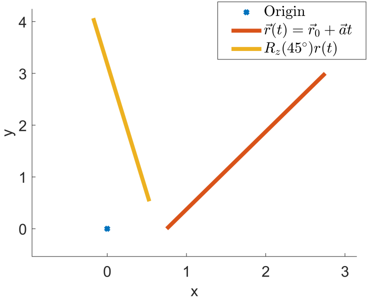
Type	Form	Example	Description
Explicit	$y = f(x)$	$y = mx + b$	Line
Implicit	$f(x, y) = 0$	$(x - p_x)^2 + \left(y - p_y\right)^2 = r^2$	Circle
Parametric	$x = g(t); y = h(t)$	$x = p_{0x} + a_x t;\ y = p_{0y} + a_y t$	Line
		$x = p_{0x} + r\cos t;\ y = p_{0y} + r\sin t$	Circle

Benefits of parametric form

Definition

A parametric description of a curve is called such if the coordinates of the curve point are **continuous** and **unambiguous** functions of the parameter t.

- The result is a point cloud, which can be easily discretized.
- It can be easily controlled.
- We can work with our parametric curves as with coordinates (change basis, apply affine transformation).



Segment line in parametric form

AGLA (6th lab) — $\vec{r}(t) = \vec{p}_0 + a(t)$ or

$$\begin{cases} x = p_{0x} + a_x t \\ y = p_{0y} + a_y t \\ z = p_{0z} + a_z t \end{cases}$$

Not easy to make a segment line (we have only one clear point and a direction)

$\vec{r}(t) = \vec{p}_0(1 - t) + \vec{p}_1 t$ or

$$\begin{cases} x = p_{0x}(1 - t) + p_{1x} t \\ y = p_{0y}(1 - t) + p_{1y} t \\ z = p_{0z}(1 - t) + p_{1z} t \end{cases}$$

It is really convenient, if you know 2 points and want to make a segment line. We will meet this form a lof of times today

Polyline (Polygonal chain)

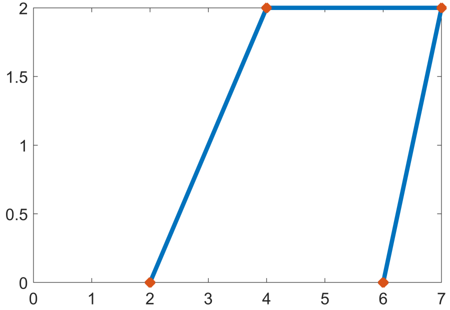
$\vec{r}(t) = \vec{p}_i(1 - w) + \vec{p}_{i+1}w$, where w is a local parameter $w = \dfrac{t - t_i}{t_{t+1} - t_i}$, $t_i \leq t \leq t_{i+1}$

Polyline passes through given control points $\vec{p}_0, \ \vec{p}_1, \ldots, \vec{p}_n$. $t_i \leq t_{i+1}$

Example

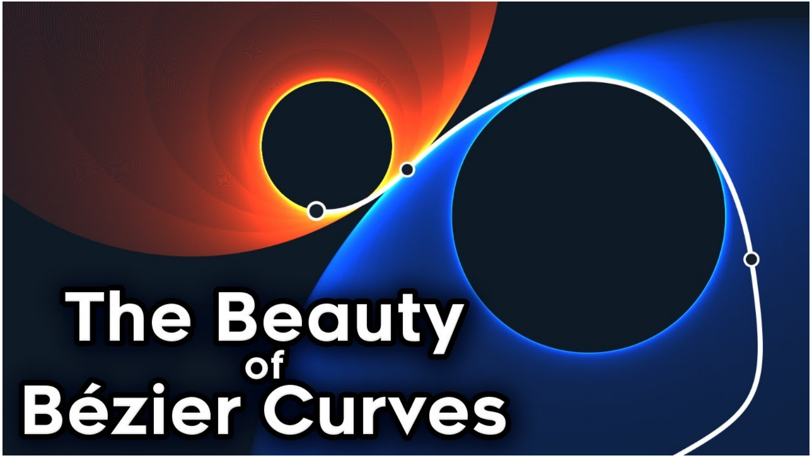
$$\vec{p}_0 = \begin{bmatrix} 2 \\ 0 \end{bmatrix}, \ \vec{p}_1 = \begin{bmatrix} 4 \\ 2 \end{bmatrix}, \ \vec{p}_2 = \begin{bmatrix} 7 \\ 2 \end{bmatrix}, \ \vec{p}_3 = \begin{bmatrix} 6 \\ 0 \end{bmatrix}$$

If each knot will be the same and equal to 1, then $\vec{r}(t) = \vec{p}_i(1 - t) + \vec{p}_{i+1}t$, $i = 0 \ldots n - 1$



Intro to splines

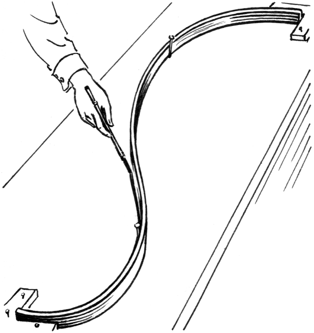
Video



Splines

Informal Definition

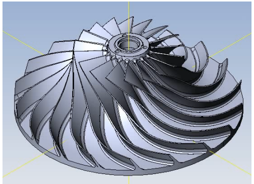
Splines (*piecewise polynomial* functions) are awesome tool to construct *smooth* and controllable shapes in computer graphics.



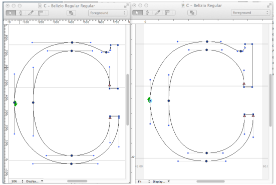
Starting 15th century, ship hull designers used splines for making a smooth shape

French curve (Лекало)

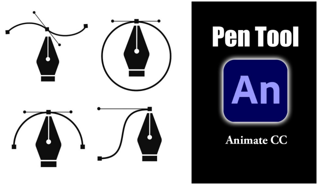
Splines: Applications



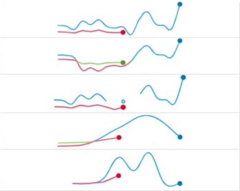
User: Car shape design, aircrafting



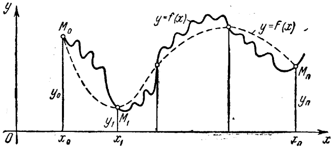
User: Make fonts



User: Pen tool in PhotoShop



Math: Interpolation — advanced data analysis



Math: Approximation — signal post processing (reduce noise)

Math: Extrapolation — revenue prediction during the covid

Cubic spline (proof in (1) p. 33-35)

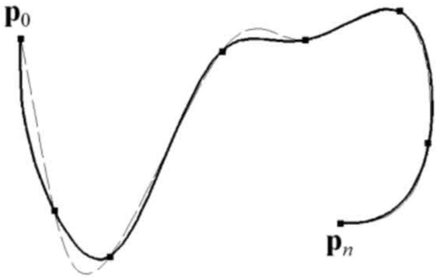
Definition

$$\vec{r}(t) = (1-w)\vec{p}_0 + w\vec{p}_{i+1} + \left((-2w+3w^2-w^3)\vec{s}_i + (-w+w^3)\vec{s}_{i+1}\right)\frac{(t_{i+1}-t_i)^2}{6},$$

$$w = \frac{t-t_i}{t_{t+1}-t_i},\ t_i \le t \le t_{i+1}$$

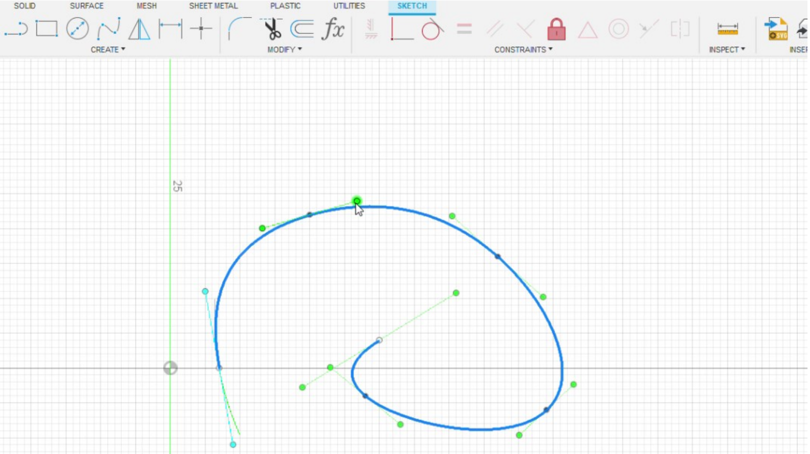
s_i — second derivative in control points

$$s_0 = s_n = 0$$



Cubic spline in Fusion 360

Video



Cubic spline (proof in (1) p. 33-35)

Proof concept

1. We have several points, which we want to connect using cubic spline, which has a continuous first and second derivative radius-vectors. Second derivative equation for each segment $t_i \leq t \leq t_{i+1}$:

$$\frac{d^2\vec{r}}{dt^2} = \vec{s}_i(1 - w) + \vec{s}_{i+1}w, w = \frac{t - t_i}{t_{t+1} - t_i}$$

Cubic spline (proof in (1) p. 33-35)

Proof concept

1. We have several points, which we want to connect using cubic spline, which has a continuous first and second derivative radius-vectors. Second derivative equation for each segment $t_i \leq t \leq t_{i+1}$:
$$\frac{d^2\vec{r}}{dt^2} = \vec{s}_i(1-w) + \vec{s}_{i+1}w,\ w = \frac{t-t_i}{t_{t+1}-t_i}$$
2. We are integrating it twice: $\vec{r}(t) = \vec{s}_i\frac{(t_{i+1}-t)^3}{6(t_{t+1}-t_i)} + \vec{s}_{i+1}\frac{(t-t_i)^3}{6(t_{t+1}-t_i)} + \vec{c}_1 t + \vec{c}_2$. Integration constants can be found putting $\vec{r}(t_i) = \vec{p}_i,\ r(t_{i+1}) = \vec{p}_{i+1}$.

Cubic spline (proof in (1) p. 33-35)

Proof concept

1. We have several points, which we want to connect using cubic spline, which has a continuous first and second derivative radius-vectors. Second derivative equation for each segment $t_i \leq t \leq t_{i+1}$:
$$\frac{d^2\vec{r}}{dt^2} = \vec{s}_i(1-w) + \vec{s}_{i+1}w,\ w = \frac{t-t_i}{t_{t+1}-t_i}$$
2. We are integrating it twice: $\vec{r}(t) = \vec{s}_i\frac{(t_{i+1}-t)^3}{6(t_{t+1}-t_i)} + \vec{s}_{i+1}\frac{(t-t_i)^3}{6(t_{t+1}-t_i)} + \vec{c}_1 t + \vec{c}_2$. Integration constants can be found putting $\vec{r}(t_i) = \vec{p}_i,\ r(t_{i+1}) = \vec{p}_{i+1}$.
3. We have two unknowns $\vec{s}_i,\ \vec{s}_{i+1}$. We equal the first derivative of the spline at the right end of the segment $t_{i-1} \leq t \leq t_i$ to the first derivative of the spline at the left end of the segment $t_i \leq t \leq t_{i+1}$ and substitute $t = t_i$ the answer is the equation from definition slide.

In such case there are $n-1$ equations for n points. We have $n+1$ $\vec{s}_i$ unknowns. Other 2 equations can be taken from the 1st and last points of the curve. We can say that they are free, hence $\vec{s}_0 = \vec{s}_n = 0$

Bezier spline (proof in (1) p. 38-42)

1-st order curve

$$\vec{r}(t) = (1-t)\vec{p}_0 + t\vec{p}_1$$

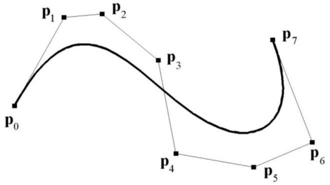
2-nd order curve

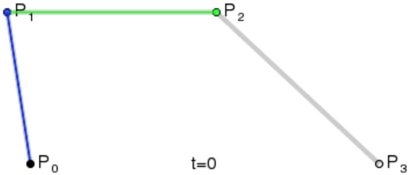
$$\vec{r}(t) = (1-t)^2\vec{p}_0 + 2t(1-t)\vec{p}_1 + t^2\vec{p}_2$$

General form

$$\vec{r}(t) = \sum_{i=0}^{n} B_i^n(t)\vec{p}_i$$

$$B_i^n(t) = \frac{n!}{i!(n-i)!}t^i(1-t)^{n-i}$$ Bernstein polynomials

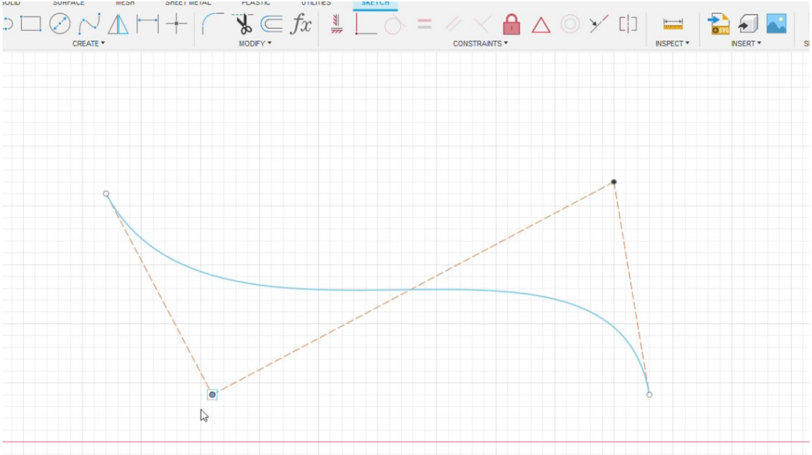




Click on the figure for a video

Bezier spline in Fusion 360

Video



Bezier splines & conic sections (proof in (1) p. 42-48) (1)

c : Ellipse(A, B, C)

→ $3x^2 + 4y^2 - 6x = 9$

D = Point(c)

→ (2, 1.5)

Tangent(D, c)

→ g: y = -0.5x + 2.5

G = Point(g)

→ (3.8, 0.6)

Tangent(G, c)

→ i: y = -0.5x + 2.5

→ j: y = 1.375x - 4.625

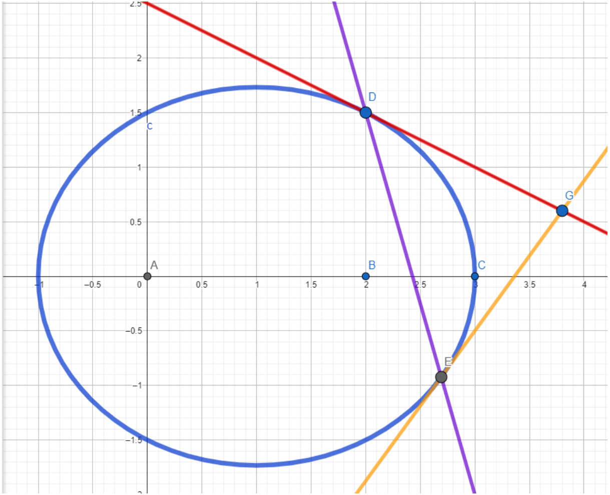
E = Intersect(c, j)

→ (2.6923076923077, -0.9230769230769)

f : Line(E, D)

→ y = -3.5x + 8.5

Input...



Bezier splines & conic sections (proof in (1) p. 42-48)

Proof concept

1. Firstly, we need to represent conic section somehow, knowing some points. We will do it making an equation, which connects two pair of straight lines. $l_i = a_i x + b_i y + c_i = 0$. For example $l_1 l_2 = 0$ is the pair of straight line equation. For simplicity, let's take $(1 - \lambda) l_1 l_2 + \lambda l_3^2 = 0$. In such case l_1, and l_2 are tangent to conic section.

Bezier splines & conic sections (proof in (1) p. 42-48)

Proof concept

1. Firstly, we need to represent conic section somehow, knowing some points. We will do it making an equation, which connects two pair of straight lines. $l_i = a_i x + b_i y + c_i = 0$. For example $l_1 l_2 = 0$ is the pair of straight line equation. For simplicity, let's take $(1-\lambda)l_1 l_2 + \lambda l_3^2 = 0$. In such case l_1, and l_2 are tangent to conic section.

2. Point G is $\vec{p}_1$, $D - \vec{p}_0$, $E - \vec{p}_2$. Let's represent the ark DE in new CS, where $\vec{u} = \vec{p}_0 - \vec{p}_1$, $\vec{v} = \vec{p}_2 - \vec{p}_1$. Hence our curve will have such equation $\vec{r}(u,v) = \vec{p}_1 + (\vec{p}_0 - \vec{p}_1)u + (\vec{p}_2 - \vec{p}_1)v$. In such case our conics equation will be represented as $(1-\lambda)uv + \lambda(u+v-1)^2 = 0$ ($l_1 \to v = 0$, $l_2 \to u = 0$, $l_3 \to u + v - 1 = 0$).

Bezier splines & conic sections (proof in (1) p. 42-48)

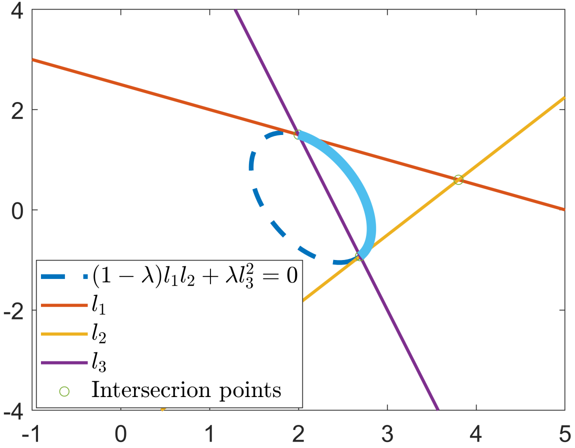
Proof concept

1. Firstly, we need to represent conic section somehow, knowing some points. We will do it making an equation, which connects two pair of straight lines. $l_i = a_i x + b_i y + c_i = 0$. For example $l_1 l_2 = 0$ is the pair of straight line equation. For simplicity, let's take $(1-\lambda)l_1 l_2 + \lambda l_3^2 = 0$. In such case l_1, and l_2 are tangent to conic section.
2. Point G is $\vec{p}_1$, $D - \vec{p}_0$, $E - \vec{p}_2$. Let's represent the ark DE in new CS, where $\vec{u} = \vec{p}_0 - \vec{p}_1$, $\vec{v} = \vec{p}_2 - \vec{p}_1$. Hence our curve will have such equation $\vec{r}(u, v) = \vec{p}_1 + (\vec{p}_0 - \vec{p}_1)u + (\vec{p}_2 - \vec{p}_1)v$. In such case our conics equation will be represented as $(1-\lambda)uv + \lambda(u + v - 1)^2 = 0$ ($l_1 \to v = 0$, $l_2 \to u = 0$, $l_3 \to u + v - 1 = 0$).
3. The problem of the previous equation — it relates to 2 components (u, v). Our goal to make it relative only to one. In the end of it the equation appears

$$\vec{r}(t) = \frac{(1-t)^2\vec{p}_0 + 2t(1-t)\omega\vec{p}_1 + t^2\vec{p}_2}{(1-t)^2 + 2t(1-t)\omega + t^2},$$

where ω is a weight of the point $\vec{p}_1$.

Bezier splines & conic sections (proof in (1) p. 42-48) (2)

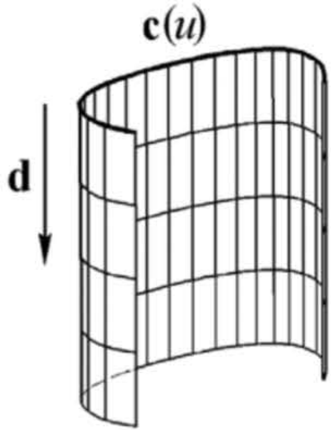


Extrude and shear surfaces (proof in (1) p. 114-118)

When a generating curve moves along a guiding curve, the orientation of the generating curve may remain unchanged related to guiding line.

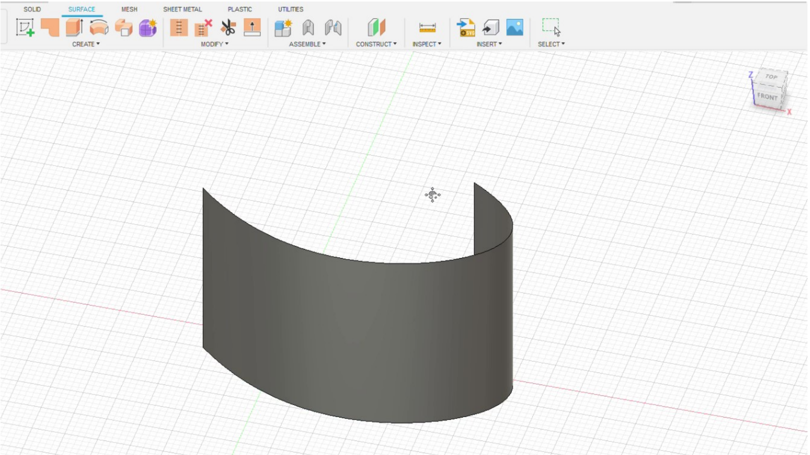
If the guiding line is a straight line — **extrude surface**, otherwise — **shear surface**

$$\vec{r}(u,v) = \vec{c}(u) + \vec{d}v$$



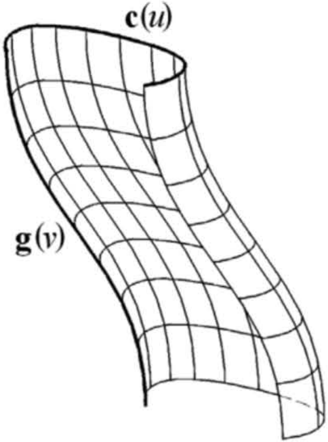
Surface made by extruding in Fusion 360

Video



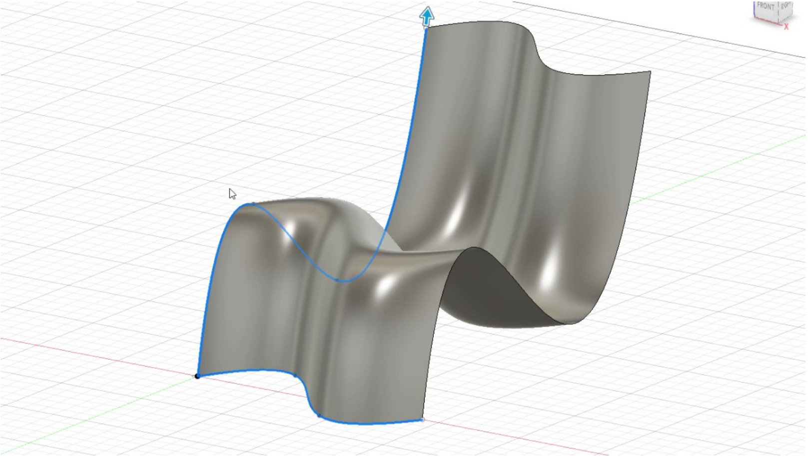
Shear Surface

$$\vec{r}(u,v) = \vec{g}(v) + \vec{c}(u) - \vec{g}(v_{min})$$



Surface made by shearing in Fusion 360

Video



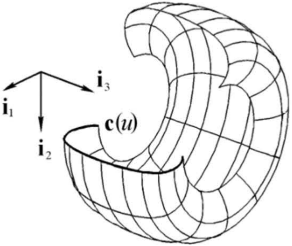
Revolve surface

$$\vec{r}(u, v) = \vec{p} + M(v)(\vec{c}(u) - \vec{p}),$$

$$u \in [u_{min},\ u_{max}],\ v \in [0,\ \alpha]$$

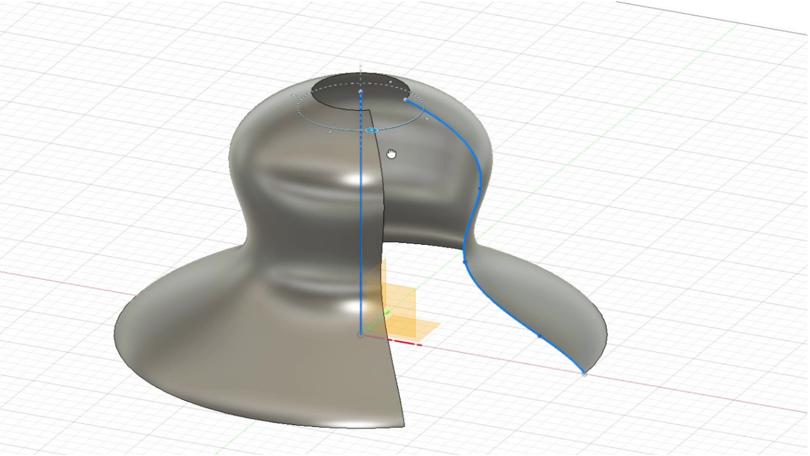
$$M(v) = A \begin{bmatrix} \cos(v) & -\sin(v) & 0 \\ \sin(v) & \cos(v) & 0 \\ 0 & 0 & 1 \end{bmatrix} A^{-1}, \text{ — rotation matrix}$$

$$A = [i_1 \quad i_2 \quad i_3] \text{ — transformation from local to global CS}$$



Surface made by revolving in Fusion 360

Video



Summary

1. We touched the benefits of a parametric form.

Summary

1. We touched the benefits of a parametric form.
2. We know the equation of the line and polyline.

Summary

1. We touched the benefits of a parametric form.
2. We know the equation of the line and polyline.
3. We heard about spline applications.

Summary

1. We touched the benefits of a parametric form.
2. We know the equation of the line and polyline.
3. We heard about spline applications.
4. We know 2 types of splines and the difference between them.
5. We got acquainted with a proof how correlates conic sections and Bezier splines.

Summary

1. We touched the benefits of a parametric form.
2. We know the equation of the line and polyline.
3. We heard about spline applications.
4. We know 2 types of splines and the difference between them.
5. We got acquainted with a proof how correlates conic sections and Bezier splines.
6. We know 3 methods of obtaining surfaces.

Reference material

- Geometrical Modeling, Golovanov N.N. (book, rus)
- Plotting parametric equations in Matlab (video, eng)
- The Beauty of Bézier Curves (video, eng)
- Computer Graphics course, lectures notes 12 and 13 (Imperial College London)
- 12 Spline Curves (video, eng)
- Data Fitting: Polynomial Fitting and Splines, Part 4 (video, eng)
- Qubic spline (habr, rus)

Deserve "A" grade!

– Oleg Bulichev

o.bulichev@innopolis.ru

@Lupasic

Room 105 (Underground robotics lab)