Reto 2

Estiben Giraldo, Julio Andrés Mejía, Angela Sofia Moreno y Anggie Carolina Correa

Abril 19 2020

Contents

1 Introducción

El presente trabajo se realizó con el objetivo de modelar un mortero valenciano tratando de aproximarse lo máximo posible a este objeto en la realidad (Figura 1), el método de interpolación utilizado en el proceso fue basado en curvas de Bézier, ya que este nos permite cumplir con dicha condición.

Figure 1: Mortero real

2 Marco teórico

2.1 Blender

Programa informático utilizado, el cual tiene como funciones el modelado, iluminación, renderizado, animación y creación de gráficos tridimensionales.

2.2 Curvas de Bézier

Sistema desarrollado para el trazado de dibujos técnicos, en el diseño aeronáutico y al de automóviles.

Para definir geométricamente las formas: un punto del plano puede definirse por coordenadas. Por ejemplo, un punto A tiene unas coordenadas $(x1, y1)$ y a un punto B le corresponde $(x2, y2)$. Para trazar una recta entre ambos basta con conocer su posición. Si en lugar de unir dos puntos con una recta se unen con una curva, surgen los elementos esenciales de una curva Bézier; los puntos se denominan "puntos de anclaje" o "nodos". La forma de la curva se define por unos puntos invisibles en el dibujo, denominados "puntos de control", "manejadores" o "manecillas".

Una curva de Bézier es una curva paramétrica basada en cuatro puntos de control. La curva comienza en el primer punto de control con su pendiente tangente a la línea entre los dos primeros puntos de control y la curva termina en el cuarto punto de control con su pendiente tangente a la línea entre los dos últimos puntos de control.

$B(t) = \sum_{(\ i=0)}^{n} P_i b_{i,n}(t), t \in [0,1]$ donde las polinomiales: $b_{i,n}(t) = \sum_{i}^{n} t^i (1 - t)^{n-i}, i = 0, ..., n$

Los puntos Pi son llamados puntos de control de las curvas de Bézier. El polígono formado por la conexión de los puntos de Bézier con rectas, comenzando por P0 y terminando en Pn, se denomina polígono de Bézier (o polígono de control). La envolvente convexa del polígono de Bézier contiene las curvas de Bézier.

Casos con curvas de Bézier:

1. Curvas lineales:

   Dados los puntos P0 y P1, una curva lineal de Bézier es una línea recta entre los dos puntos. La curva viene dada por la expresión:

   $$B(t) = (P_1 - P_0)t = (1 - t)P_0 + tP_1, t \in [0, 1]$$

2. Curvas cuadráticas:

Una curva cuadrática de Bézier es el camino trazado por la función B(t), dados los puntos: P0, P1, y P2:

$$B(t) = (1-t)^2 P_0 + 2t(1-t)P_1 + t^2, t \in [0,1]$$

3. Curvas cúbicas:

   Cuatro puntos del plano o del espacio tridimensional, P0, P1, P2 y P3 definen una curva cúbica de Bézier. La curva comienza en el punto P0 y se dirige hacia P1 y llega a P3 viniendo de la dirección del punto P2. Usualmente, no pasará ni por P1 ni por P2. Estos puntos solo están ahí para proporcionar información direccional. La distancia entre P0 y P1 determina qué longitud tiene la curva cuando se mueve hacia la dirección de P2 antes de dirigirse hacia P3.

   La forma paramétrica de la curva es:

$$B(t) = P_0(1-t)^3 + 3P_1 t(1-t)^2 + 3P_2 t^2(1-t) + p_3 t^3, t \in [0,1]$$

Spline de Bézier:

Una spline de Bézier es una curva que consta de varias curvas de Bézier unidas entre sí, la curva es del tipo:

$B(t) = \sum_{(\ i=0)}^{n} P_i(1-t)^{n-i}t^i = P_0(1-t)^n + \binom{n}{i}P_1(1-t)^{n-1}t + ... + P_n t^n, t \in [0,1]$

3 Proceso de implementación:

3.1 Modelado en Blender

Para el modelamiento del mortero valenciano utilizamos la herramienta informática, blender la cual nos permitió modelar este objeto en 3 dimensiones, y tener claras las coordenadas necesarias para la creación de dicho objeto.

Las siguientes son ilustraciones del trabajo realizado con la esta herramienta:

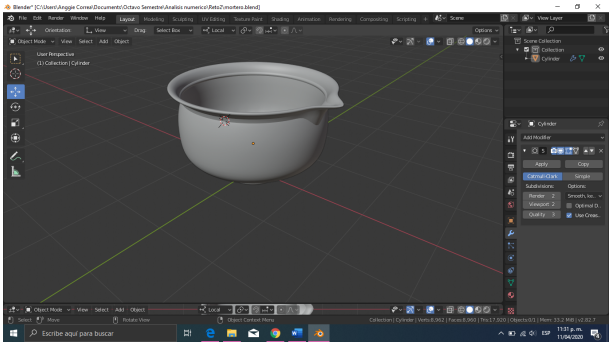

Figure 2: Modelado en Blender

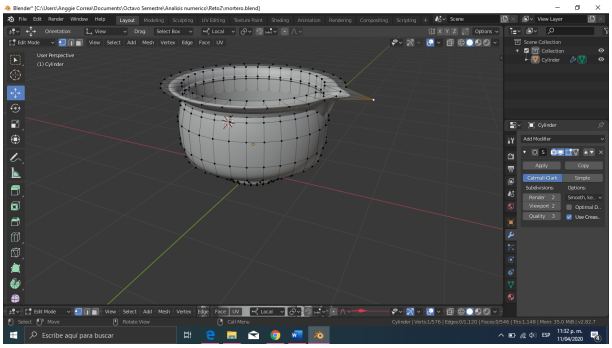

Figure 3: Modelado en Blender con coordenadas x,y,z

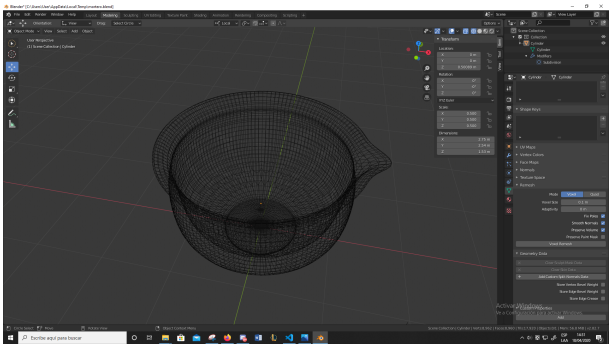

Figure 4: Modelado en Blender sin textura

Del anterior modelamiento se obtuvieron un total de 575 coordenadas y las siguientes son algunas de ellas:

x	y	z
0	1.0	-1.0
-0.980785071849823	0.1950913518667221	-1.0
0.878602147102356	0.5870630741119385	-0.9205889105796814
-1.600400447845459	-0.6629062294960022	-0.4821978211402893
2.357809066772461	-2.1443733544401766e-07	0.3243255615234375
-2.1944737434387207	0.9089831709861755	1.0606906414031982
1.524717092514038	-1.5247197151184082	1.7605034112930298
-0.3462047874927521	0.8358157873153687	-1.0
-2.0174326209598803e-08	-0.290657639503479	-0.881237268447876
0.824410617351532	1.9903031587600708	2.0910534858703613
-1.2805064916610718	-1.2805052995681763	-0.2806932330131531
2.1488771438598633	0.42743873596191406	1.6819599866867065
-1.0960412311078471e-07	2.190979480743408	1.6783939599990845

Table 1: Coordenadas obtenidas

De igual manera el .txt y el archivo Blender se va a agregar al repositorio para poder indentificar todos los puntos usados al momento de crear el mortero propuesto.

3.2 Implementacion en Python3

Para realizar la implementacion del modelado en blender se hizo de las librerias matplotlib, numpy, math y mplot3d para realizar la grafica en 3D del mortero creado punto a punto.

Para empezar se realizo el dibujo en 3D del vaso recto del mortero, ya que este no representaba mayor dificultad al momento de su construccion.

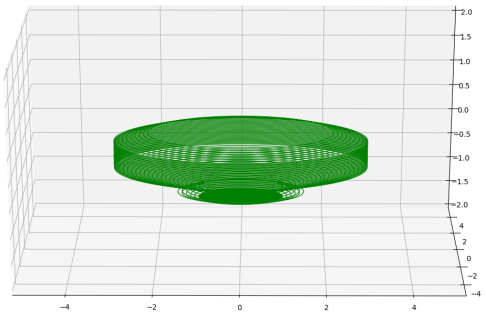

Figure 5: Base del mortero

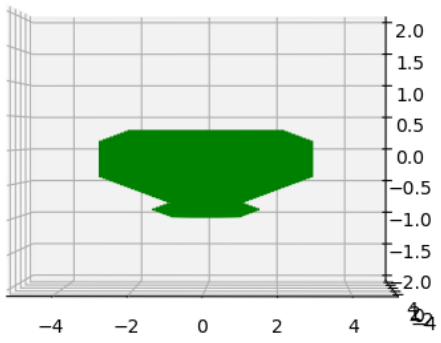

Figure 6: Base del mortero

Haciendo uso del teorema de bezier se diseño la parte superior del mortero, incluyendo en esta una curvatura similar a la de un mortero valenciano, tal y como habia sido solicitado.

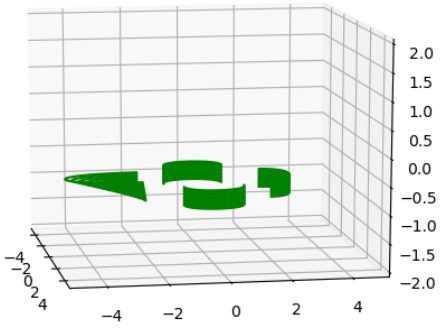

Figure 7: Parte superior del mortero

Al final se unen ambas partes para mostrar el mortero completo en grafico en el grafico en 3D que se ajusta al modelo realizado en blender, donde el mortero presenta una curvatura pronunciada en la parte superior.

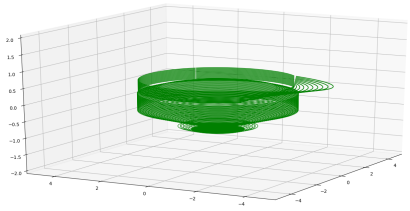

Figure 8: Vista 1

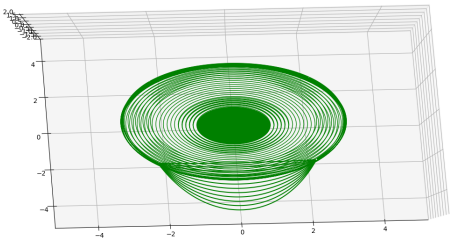

Figure 9: Vista 2

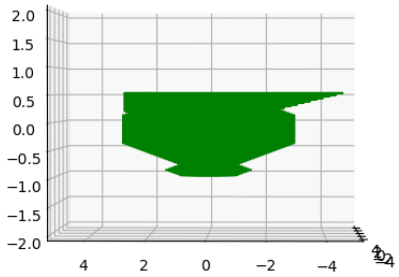

Figure 10: Vista 3

Para determinar el area superficial del mortero se hizo uso de las formulas basicas para calcular el area de media esfera y el area de una piramide. estas operaciones se hicieron teniendo en cuenta la altura, el radio y el ancho del mortero que se obtuvo gracias al modelado que se hizo en blender:

$$\text{Area superficial: } A = (2\pi * 1.0 * 1.0) + 4 * (0.3 * (\tfrac{0.3}{2}) + 3)$$

$$\text{A} = 6.823185307179586$$

Para determinar el error que tenian los calculor realizados se realizo una comparacion entre el resultado obtenido con los calculos anteriores y el resultado que propone blender.

Error del Area:

$$Ea = \frac{7.2156 - 6.823185307179586}{7.2156} * 100$$

Ea=5.4 %

References

[1] Bourke,P., *Bézier curves*, From: http://astronomy.swin.edu.au/ pbourke/curves/bezier/

[2] Knuth,D., *Metafont: the Program*, From: Addison-Wesley 1986, pp. 123-131

[3] Olsen,A., *Toolkit for Bezier Curves and Splines*, From: https://uvirtual.javeriana.edu.co/webapps/blackboard/execute/content/file?cmd=viewcontent-id=-383105-1course-id=-11616-1

[4] Murrell,P., *Variable-Width Bezier Splines in R*, From: https://www.stat.auckland.ac.nz/ paul/Reports/VWline/offsetbezier/offsetbezier.htmlbezier

[5] Kjetil,A., *SpliPy*, From: https://pypi.org/project/Splipy/

[6] Morten,A. , Kjetil,A. and Fonn,E., *SpliPy,BSplineBasis*, From: https://sintefmath.github.io/Splipy/basic-classes.htmlsplipy.BSplineBasis