Reconstrucción de Figuras Tridimensionales

Luisa Yurani Contreras Vergel
Facultad Ingeniería
Dpto. Ingeniería de Sistemas
Pontificia Universidad Javeriana
Bogotá, Colombia
Luisa.contreras@javeriana.edu.co

Gabriel Fernando Forero Ortiz
Facultad Ingeniería
Dpto. Ingeniería de Sistemas
Pontificia Universidad Javeriana
Bogotá, Colombia
Forero-gabriel@javeriana.edu.co

Jairo Hernán Vanegas García
Facultad Ingeniería
Dpto. Ingeniería de Sistemas
Pontificia Universidad Javeriana
Bogotá, Colombia
Jairo.vanegas@javeriana.edu.co

***Abstract*—In this document the development of the reconstruction of the figure of a jug will be reflected by means of the Interpolation of curves and surfaces of Béizer**

Keywords—interpolation, surface of Beizer, reconstruction of figures, PathInterpolatR, gridBezier,vwline

I. INTRODUCCIÓN

El presente trabajo la propone a reconstrucción de figuras tridimensionales, por medio de la modelación de las misma por medio de la herramienta Blender para luego aplicar funciones de interpolación como lo son Interpolación por Splines, superficies y curvas de Béizer de esta manera se obtendrán los puntos que nos dará la imagen de la figura seleccionada además en el desarrollo del mismo usaremos el algoritmo Marching cubes, todo esto siendo aplicado en la herramienta Rstudio, la cual nos permite

II. MARCO TEORICO

1) *Blender:* Blender es la suite de creación 3D gratuita y de código abierto. Es compatible con la totalidad de la canalización 3D: modelado, montaje, animación, simulación, renderizado, composición y seguimiento de movimiento. Blender es software libre. Blender se puede usar para cualquier propósito, incluyendo el comercio o para la educación. Blender es multiplataforma y se ejecuta igualmente bien en computadoras Linux, Windows y Macintosh. Su interfaz usa OpenGL para brindar una experiencia consistente. [1]

2) *Interpolación por Splines:* En el campo de análisis numérico, un spline es una curva diferenciable definida en funciones mediante polinomios, es decir, una función spline está formada por varios polinomios, cada uno definido sobre un subintervalo, que se unen entre sí obedeciendo a ciertas condiciones de continuidad [2].
Además, la interpolación por medio de Splines Cúbicos permite crear funciones de orden superior que aproximan los datos tomados a una curva continua que conserva la suavidad inherente a toda variable física. Además de la utilidad que presentan los Splines en la interpolación de datos, también se encuentran múltiples aplicaciones de este tipo de funciones en diferentes áreas de la ingeniería, especialmente en el campo del procesamiento digital de imágenes, pues son ideales en la reconstrucción, alisado, filtrado, ampliación y reducción de fotografías. [3]

3) *Curvas de Béizer:* Es un sistema que se desarrolló hacia los años 1960, fue pensado para el trazado de dibujos técnicos, se usa especialmente en el diseño aeronáutico y de carros [4]. Fue propuesto por Pierre Etienne Bézier, quien fue jefe de diseño y producción de automóviles Renault durante la mayor parte de suvida profesional. Su investigación comenzó sobre el diseño y la fabricación asistidos por computadora en 1960, desarrollando así herramientas interactivas para el diseño de curvas y superficies, e inició el fresado generadopor computadora para el modelado de automóviles [5]. Actualmente en el diseño computacional, las curvasde Bézier son de gran importancia.

Propiedades de las curvas de Béizer:

a) Generalmente siguen la forma del polígono de control, que consiste en los segmentos que unen los puntos de control.

b) Siempre pasan por el primer y último punto de control.

c) Están contenidos en el casco convexo de los puntos de control que los definen.

d) El grado del polinomio que define el segmento de la curva es uno menos que el número de puntos que definen el polígono. Por lo tanto, para 4 puntos de control, el grado del polinomio es 3, es decir, polinomio cúbico.

e) Una curva de Bezier generalmente sigue la forma del polígono que la define.

f) La dirección del vector tangente en los puntos finales es la misma que la del vector determinado por el primer y último segmento.

g) La propiedad de casco convexo para una curva de Bezier asegura que el polinomio siga suavemente los puntos de control.

h) Ninguna línea recta interseca una curva de Bézier más veces de las que intersecta su polígono de control.

i) Son invariantes bajo una transformación afín.

j) Las curvas de Bézier exhiben un control global, lo que significa que mover un punto de control altera la forma de toda la curva.

k) Una curva de Bezier dada se puede subdividir en un punto t = t_0 en dos segmentos de Bezier que se unen en el punto correspondiente al valor del parámetro t = t_0.

Casos de curvas de Béizer

Existen tres grados para las curvas de Bézier, los cuales se encontrarán a continuación:

1. *Curvas lineales de Béizer :* Dados los puntos P0 y P1 la curva de Bézier es una línea recta entre los dos puntos:

$$B(t) = P_0 + (P_1 - P_0)t = (1-t)P_0 + tP_1 , t\ 2\ [0;\ 1]$$

Ecuación 1 Curvas Lineales de Béizer

2. *Curvas cuadráticas de Bézier*: Es el camino trazado por la función B(t), dados los puntos: P0, P1,y P2:

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

Ecuación 2 Curvas cuadráticas de Bézier

3. *Curvas cúbicas de Bézier:* 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. La forma paramétrica de la curva es:

$$B(t) = P_0(1-t)^3 + 3P_1t(1-t)^2 + 3P_2t^2(1-t) + P_3t^3 , t\ 2\ [0;\ 1]$$

Ecuación 3 Curvas cúbicas de Béizer

Generalización: Cuando se generalizan las curvas de Bézier pasan a ser curvas Spline, es decir, la curva de Bézier es una curva Spline de tercer grado, la curva de n grado de Bézier se generaliza de la siguiente manera:

$$B(t) = \sum_{i=0}^{n} \binom{n}{i} P_i(1-t)^{n-i}t^i = P_0(1-t)^{n-i} + \binom{n}{1} P_1(1-t)^{n-1}t + \cdots + P_nt_n , t\ 2\ [0;\ 1]$$

Ecuación 4 Generalización

*4) Algoritmo Marching cubes:*El algoritmo Marching cubes usa una clasificacion de celdas que pertenecen a una superficie umbral. Dado un valor umbral y una grilla cúbica conteniendo los datos del objeto a modelar, el algoritmo original la procesa considerando cada celda en un orden determinado, generando localmente una isosuperficie. Esta isosuperficie queda representada a través de triángulos cuyos vértices están en las aristas de las celdas que procesa. Cada arista de la celda puede tener a lo sumo una intersección con la isosuperficie, la cual se produce cuando el valor umbral queda acotado inferior y superiormente por los valores en los vértices de la arista. La forma de la superficie dentro de cada celda depende solamente de la combinación de puntos con valor mayor o menor al umbral en dicho cubo (puntos marcados o no marcados). [6]

5) Rstudio: En el presente ejercicio es imprensindible el uso de la herramienta R por el entorno que nos ofrece al encontrar los puntos por medio de los conceptos ya contemplados anteriormente. Las librerías que se usarán serán las definidas a continuación:

a) PathInterpolatR:métodos para la interpolación de rutas: Se proporcionan métodos para realizar la interpolación de trayectorias en datos de movimiento (por ejemplo, datos de seguimiento GPS). Se incluyen funciones para realizar interpolaciones usando lineales, curvas de Bezier, splines Catmull-Rom, paseos aleatorios con restricciones geográficas de tiempo e interpolación cinemática. [7]

b) GridBezier: Calcula puntos en una curva de Bezier y / o tangentes y / o normales a la curva en esos puntos. Todas las funciones devuelven una lista con los componentes xey. Para BezierPoints, estas son ubicaciones en la curva. Para BezierTangent y BezierNormal, estas son las distancias a los puntos finales de segmentos de línea tangentes o normales. Todos los valores están en pulgadas. [8]

c) Vwline: Calcule puntos en el borde o límite de una línea de ancho variable. Si la distancia es numérica, se supone que es una proporción de la longitud de un borde.Lo que constituye un borde varía entre los diferentes métodos: algunos métodos producen bordes izquierdo y derecho distintos (ignorando los finales de línea), en cuyo caso se pueden encontrar ubicaciones en cualquier borde; otros métodos generalmente producen un límite único (incluidos los finales de línea), pero las líneas que se entrecruzan pueden producir límites adicionales. Para algunos métodos, es posible que un límite forme bucles, por lo que no se garantiza que la ubicación del borde esté en el límite externo de la línea de ancho variable.Para las líneas con distintos bordes izquierdo y derecho, la dirección hacia adelante es la dirección de la curva principal. Para líneas con un solo límite, la dirección de avance es en sentido antihorario. Para las líneas con un solo límite, el punto de inicio en el límite se define como el punto más cercano en el límite a la ubicación especificada por x0 e y0. [9]

Para los métodos con un solo límite, una lista con los componentes xey, dando ubicaciones en el borde de la línea de ancho variable y tangente, dando la tangente al borde en cada ubicación.Para métodos con distintos bordes izquierdo y derecho, una lista con los componentes izquierdo y derecho, cada uno como la lista anterior. [9]

III. Problema Planteado

El problema consiste en que mediante métodos de interpolación se logre modelar un objeto 3D dado unos puntos de control, para este problema se usará el método de Bézier y se iniciará modelándola en la herramienta Blender para luego dividirlo, ya que estamos frente a una figura simetría obteniendo los puntos e interpolándolos por las herramientas descritas en el marco teórico y luego obtener una curva con dichos puntos que posteriormente será proyectada en el eje Z y replicada en cada uno de los otros cuadrantes.

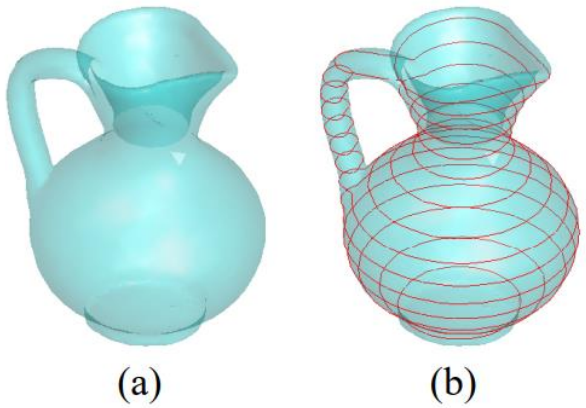

Ilustración 1 Objeto a Modelar

Tal como se muestra la ilustración 1, es la figura tridimensional a modelar además del resultado que se espera y que en el presente documento se presentará el desarrollo del mismo.

IV. DESARROLLO DE LA SOLUCION PROPUESTA

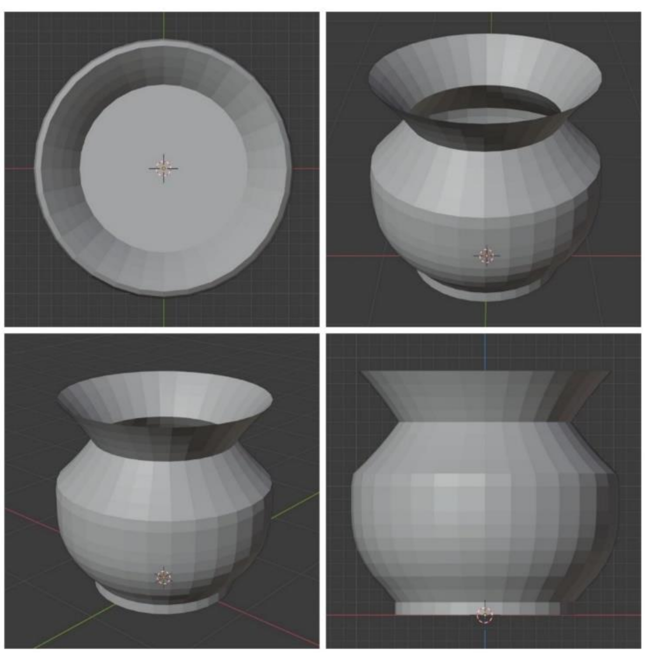

Ilustración 2 Modelación Objeto en Blender

En la ilustración 2, podemos observar la modelación obtenida con la herramienta Blender desde diferentes ángulos.

En este caso, una de las ventajas es la simetría del objeto. Es por ello, que se tomó la decisión de su división en 4 partes iguales y simétricas, para de este modo facilitar el proceso de interpolación.

De este modo, logramos la obtención del archivo Vasija.ply [Anexo 1] exportado desde Blender, el cuál alberga toda la información de los vértices, en especial las coordenadas XYZ. Sin embargo, en este documento, no es claro el orden de los mismos, además de falta procesarlos para poder utilizarlos de la manera correcta, por lo que se recurrió a la creación de una herramienta que transforma los puntos agrupándolos en el eje Z, y ordenándolos de forma ascendente en el eje X, por medio del documento Parser [Anexo 2] que tiene el formato de un script de JavaScript, claramente.

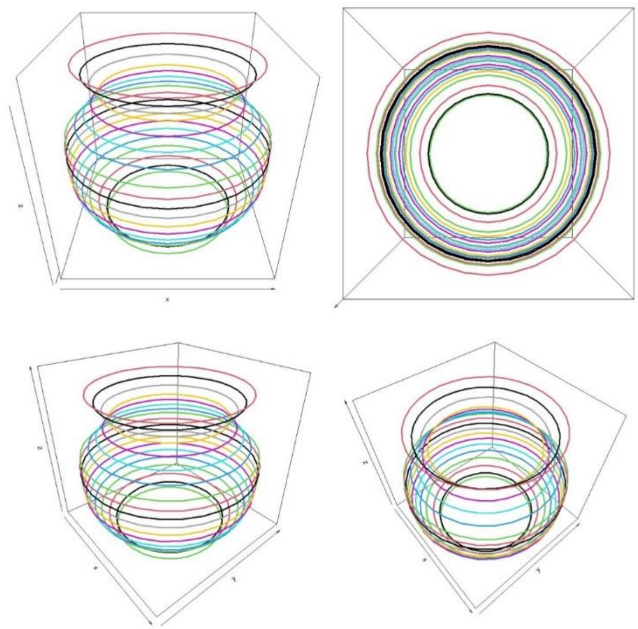

Ilustración 3 Modelado Curvas de Nivel

En la ilustración 3 podemos observar como su nombre lo indica el modelado de la vasija a través de las curvas de nivel obtenidas luego de la organización de los puntos y unión de las mismas.

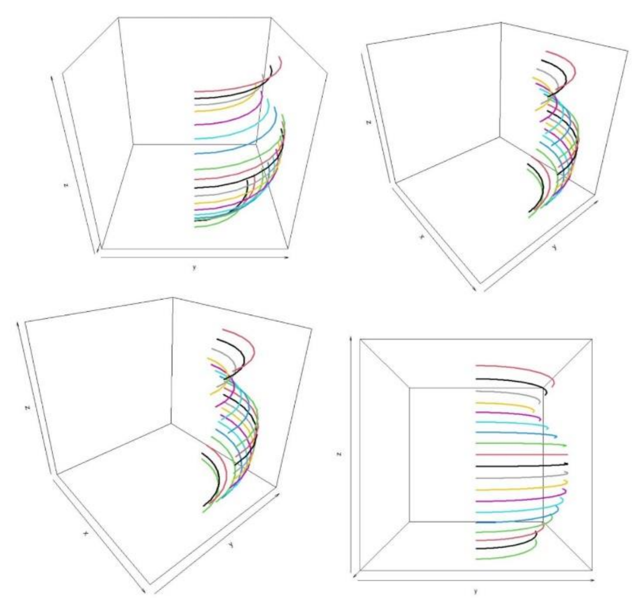

Ilustración 4 Interpolación de Curvas de Bézier

En la ilustración 4 podemos observar el primer cuadrante de las curvas de nivel ya interpoladas. Se crearon puntos intermedios entre los vértices que son los que le dan la curvatura a la figura, que se explica de la siguiente manera:

Lo primero que se realizó fue que por el método de curvas de Bezier el primer paso fue hacer una pequeña modificación que fue la creación de puntos nuevos ya mencionados (intermedios), El proceso de creación de nuevos se puede observar en la siguiente manera.

Sea un punto (Xi, Yi) y un punto siguiente $(xi, yj); j = i + 1$ entonces se puede hallar un nuevo punto (Xp, Yp) de la siguiente manera:

$$(Xp, Yp) = \frac{(xi; yi) + (xj\ ; yj\ )}{2}$$

La creación de estos nuevos puntos es necesaria para ayudar a formar curvas de Bezier que representen correctamente la superficie objetivo. Esto es importante para este método y no para el de Splines debido a que Bezier usa los puntos a manera de guía, pero sin tocarlos, esto se explica a continuación:

Los puntos recibidos cuando se trabajan con Curvas de Bezier se podrían clasificar en dos tipos.

La primera clasificación son los puntos de inicio y fin, los únicos que Bézier intercepta en la creación de su curva. La segunda clasificación son los puntos de control, Bézier usa los puntos de control como referencias para determinar la pendiente tangente de la línea.

ERROR

El error en este procedimiento es 0, pues al escoger la herramienta de modelado Blender nos da los vértices los cuales son los mismos puntos de control por lo que, los vértices y los puntos de control son los mismos puntos, por esta razón es imposible determinar cuáles son los posibles errores pues existe una sola fuente de información que la tomamos como una verdad absoluta, por la situación presentada.

I. CONCLUSIONES

Podemos concluir que las herramientas de modelado 3D pueden tener problemas, como lo presentamos pues no tienen en cuenta la toma de errores.

Consideramos implementar el algoritmo de Marching cubes sin embargo, pero con las herramientas que tenemos en R y las inexperiencia con el lenguaje, fue muy complicado y no tuvimos ningún resultado fructuoso, pero a su vez, nos ayudó a comprender de una mejor forma de interpretar los otros métodos en un espacio tridimensional.

Por otro lado, podemos ver la utilidad de las herramientas estudiadas de todo el curso se fusionan con los conceptos claves pueden desencadenar el desarrollo de un ejercicio como el planteado.

II. REFERENCIAS

[1] Blender, «Blender.com,» [En línea]. Available: https://www.blender.org/about/#:~:text=It%20supports%20the%20entirety%20of,video%20editing%20and%20game%20creation.&text=Blender%20has%20no%20price%20tag,is%20your%20own%203D%20software.. [Último acceso: 13 11 2020].

[2] C. James, «Multi-variable curve interpolation,» 1964. [En línea]. Available: https://dl.acm.org/doi/10.1145/321217.321225. [Último acceso: 11 13 2020].

[3] Universidad Tecnológica de Pereira, «INTERFAZ GRÁFICA PARA LA INTERPOLACIÓN DE DATOS A TRAVÉS DE SPLINES,» [En línea]. Available: https://www.redalyc.org/articulo.oa?id=84917316035. [Último acceso: 13 11 2020].

[4] P. Bourke, « Bézier curves,» [En línea]. Available: http://astronomy.swin.edu.au/ pbourke/curves/bezier/. [Último acceso: 13 11 2020].

[5] L. D. J. Richard, Numerical Analysis 9th Edition, Youngstown State University, 2010.

[6] Universidad Nacional del Sur, «Una Implementación Eficiente del Algoritmo de,» [En línea]. Available: http://sedici.unlp.edu.ar/bitstream/handle/10915/23546/Documento_completo.pdf?sequence=1&isAllowed=y#:~:text=Uno%20de%20los%20algoritmos%20de,matriz%20volum%C3%A9trica%20de%20datos%20escalares.. [Último acceso: 13 11 2020].

[7] rdrr.io, «jedalong / PathInterpolatR,» [En línea]. Available: https://rdrr.io/github/jedalong/PathInterpolatR/. [Último acceso: 13 11 2020].

[8] P. Mureell, «Package ‘gridBezier’,» [En línea]. Available: https://cran.r-project.org/web/packages/gridBezier/gridBezier.pdf. [Último acceso: 13 11 2020].

[9] P. Murrell, «Package ‘vwline’,» [En línea]. Available: https://cran.r-project.org/web/packages/vwline/vwline.pdf. [Último acceso: 13 11 2020].

.