

Reto 2- Reconstrucción de superficies

Reconstrucción de un jarrón y sus curvas de nivel con interpolación

John Jairo Gonzalez Martinez
johnngonzalez@javeriana.edu.co

Daniel Esteban Tibaquira Galindo
danieltibaquira@javeriana.edu.co

Stiven Gonzalez Olaya
stiven_gonzalez@javeriana.edu.co

Karen Sofia Coral Godoy
corallg_ksofia@javeriana.edu.co

Abstract—Este documento plantea una solución para la reconstrucción de un jarrón y sus curvas de nivel, utilizando métodos como: superficies de Bezier y el método de Splines, expresando los errores obtenidos para cada uno de los métodos para finalmente deducir cual genera una mejor aproximación y modelado del objeto real.

I. INTRODUCCIÓN

Durante décadas se han realizado extensos avances en materia de la representación de objetos de la vida real en entornos digitales. Sin duda han sido de gran ayuda a las industrias de la ingeniería, arquitectura, entretenimiento, diseño entre otras. La mayoría de veces es difícil para el usuario común comprender cómo se pueden realizar tales representaciones, por esto buscamos con este artículo presentar diferentes técnicas de reconstrucción de superficies, de esta manera proveer la teoría y herramientas necesarias para comprender cómo podemos generar abstracciones de cuerpos reales.

II. PROBLEMA PLANTEADO

A. Interpolación con curvas- Superficies de Bézier



Fig. 1. Prototipo de jarrón

Inicialmente se plantea un jarrón, esta tradicional figura ha ido mas allá de la cocina para llegar al mundo de la interpolación, esto debido a sus características como superficie pero también tiene suficientes modificaciones para presentar un problema de interpolación con una complejidad retadora e interesante. Precisamente ésta es la problemática que el presente documento pretende resolver en principio, sin embargo,

hay un añadido. Las curvas de Bezier son un instrumento matemático para la modelización de curvas y superficies. Este planteamiento se desarrolló alrededor de 1960 para el trazado de dibujos técnicos y es ampliamente usado en el diseño aeronáutico y automovilístico. Por su forma de crear una descripción matemática de curvas, Bezier es también una técnica común de interpolación y es por esto que es el añadido del problema. Para resumir, el objetivo planteado es lograr recrear un jarrón usando curvas de Bezier y con Splines.

III. SOLUCIÓN PLANTEADA

La idea principal para la solución del problema es relativamente sencilla, hay que aplicar funciones de interpolación, tanto de Splines como con Curvas de Bezier, a una serie de puntos que representen la superficie objetivo, en este caso un jarrón. Por lo tanto el primer paso es encontrar un conjunto de puntos que representen vértices claves de la superficie, estos puntos deben ser organizados bajo algún criterio que facilite los siguientes pasos, los criterios usados en esta solución se explican más adelante. Teniendo un conjunto de puntos ya definidos y organizados se procede a aplicar funciones de interpolación sobre los puntos, esto sin tener en cuenta la dimensión Z.

Los puntos resultantes de las interpolaciones son los que se van a graficar de manera tridimensional para finalmente lograr dibujar el jarrón. El componente Z se agrega de manera manual, teniendo en cuenta el modelo del 3D y su crecimiento en el mismo componente anterior. La secuencia de pasos realizada para obtener la solución al problema se muestra a continuación.

A. Modelado del jarrón

El primer inconveniente a la hora de pensar en la solución del problema planteado, es hallar una aproximación a los puntos que representan un jarrón en el espacio tridimensional. Para solucionar este problema de la mejor manera, el equipo decidió realizar un modelo 3D en la herramienta Blender, una de las más conocidas y más usadas en éste ámbito.

B. Obtención de los puntos

Con las herramientas brindadas por el software y basándose en la “Fig. 1” se procedió a crear el objeto. Como únicos

objetos se agregó una esfera y un cilindro, los cuales fueron cortados y modificados hasta acomodarse lo más posible al jarrón. Esto se pudo lograr con acciones de Blender tales como: corte, transformar y deformación elástica. En Fig. 2 y Fig. 3 se muestran diferentes tomas del objeto modelado

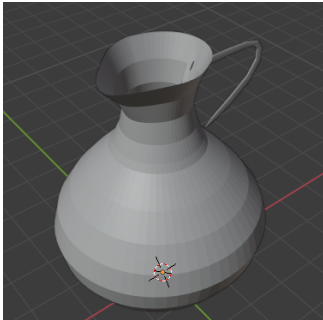


Fig. 2. Vista general del jarrón modelado



Fig. 3. Vista frontal del jarrón modelado

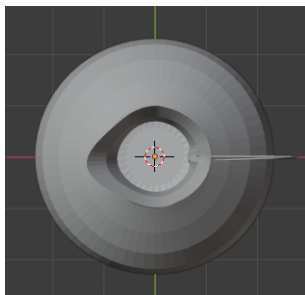


Fig. 4. Vista superior del jarrón modelado

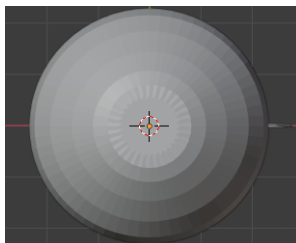


Fig. 5. Vista inferior del jarrón modelado

En este caso, una de las ventajas es la simetría del objeto. Es por ello, que éste se dividió en 2 partes y de esta forma facilitar el proceso de interpolación. En las figuras Fig. 4 y Fig. 5 se muestran las diferentes tomas del objeto que se usó para las interpolaciones.

Con la ayuda de las funcionalidades de Blender se exportó el objeto a un archivo (.obj) que contenía la información general del mismo, entre esta información se encuentran los puntos que utiliza el programa para realizar los trazos, dichos puntos fueron extraídos manualmente y fueron los utilizados para el desarrollo de estos ejercicios de interpolación. Para analizar estos puntos se siguió el siguiente procedimiento:

- Se extrajeron los puntos del archivo .obj y se pasaron a un archivo de texto (anexo Puntos Blender.txt)
- Se filtraron los puntos de forma tal que solo quedaran aquellos con coordenadas y positivas, independientemente de las coordenadas x y z , con el fin de obtener una parte del jarrón que pudiese ser proyectada de forma simétrica para obtener los resultados esperados
- Se crearon niveles para agrupar los puntos según sus alturas (coordenadas z de cada punto)
- Se ajustaron las alturas de algunos puntos con el fin de aumentar la cantidad de puntos por nivel y unificar la altura en los casos donde habían pequeñas variaciones en dicho parametro (donde la diferencia era menor a 0.15), como en el caso de la boca del jarrón
- Se contaron la cantidad de puntos por cada nivel
- Se conservaron todos los puntos que se ajustaban a dichos niveles (anexo Total Puntos Filtrados.txt) y se descartaron los que no

Este proceso dejó como resultado un total de 17 niveles, cada uno con una cantidad de puntos que varía entre 16 y 32 y que se presentan con mayor claridad en la tabla I. Para cada nivel se utilizaron 10 de estos puntos (anexo Puntos Modelos.txt) y los restantes fueron utilizados para comprobar la exactitud en los modelos obtenidos (cálculo del error). Así, para cada nivel se utilizaron los mismos 10 puntos tanto en el proceso de interpolación por Splines como en el de Bezier. Estos puntos fueron escogidos de siguiendo el siguiente procedimiento para cada uno de los niveles obtenidos:

- Los puntos fueron ordenados de menor a mayor (según sus coordenadas x)
- Se seleccionaron los puntos de los extremos (aquellos con el menor y el mayor valor de x)
- Se escogieron los 8 puntos restantes realizando saltos semi-equidistantes (basados en la posición y no en los valores de las coordenadas) sobre la lista ordenada de puntos sobrantes (se dice semi-equidistante ya que no siempre se podían conservar las mismas distancias en los saltos dado que no había una cantidad igual de puntos para todos los niveles o la cantidad presente no permitía particiones iguales)

TABLE I
TABLA DE PUNTOS POR NIVEL

Nivel (z)	Interpolados	Validación de error	Total
8e-06	10	6	16
0.02094	10	21	31
0.1674	10	21	31
0.5232	10	21	31
1.005	10	21	31
1.507	10	21	31
2.009	10	21	31
2.511	10	21	31
3.014	10	21	31
3.495	10	21	31
3.851	10	21	31
4.018	10	21	31
4.181	10	18	28
4.526	10	20	30
4.993	10	20	30
5.481	10	20	30
5.968	10	21	31
Total	170	336	506

Cabe resaltar que el total de puntos usados en cada nivel y presentados en la tabla I es en realidad el doble para cada caso, pues para cada nivel para obtener la forma completa del objeto se realizaron 2 procedimientos idénticos cambiando únicamente de signo los valores de y de cada punto, pues el objeto se reconstruyó como el reflejo de dos partes idénticas. Adicionalmente dada la naturaleza de los puntos extraídos, fue necesario realizar una unión forzada entre los puntos extremos de los reflejos, que se evidencia en la forma que las soluciones presentadas acogen en los extremos frontales y traseros de cada nivel, como se alcanza a percibir en las figuras 7, 8, 11, 12, etc., donde dichos extremos se ven de forma más plana, pues aquí fue imposible obtener los puntos usados por Blender para construir el objeto real y rellenar el vacío presente en estas partes, además, fue imposible encontrar puntos aproximados que se acomodaran a los modelos obtenidos sin modificarlos de forma drástica

Este procedimiento fue realizado con la ayuda de un Script programado en Java con el fin de automatizar dicho proceso y así poder hacer cambios de forma ágil en la forma de extraer los puntos, en caso de que fuera necesario realizar nuevos filtros o ajustes a los puntos (que de hecho fue lo que sucedió).

C. Obtención de Resultados

Los puntos obtenidos fueron formateados para ser ingresados a la herramienta R que fue la utilizada para hallar las interpolaciones. El proceso de interpolación por Splines fue realizado utilizando las funciones *spline* y *splinefun* ofrecidas por R, mientras que para Bezier, se utilizó la librería *bezier*, cada una de estas funcionalidades brindaba como resultado una serie de puntos que fueron usados para realizar los gráficos que imitaban la forma del jarrón deseado (ver secciones IV y V). Adicionalmente fueron agregados al gráfico los puntos utilizados como base para la interpolación. Por último, se procedió a calcular los errores respectivos entre

los modelos obtenidos y el objeto real (creado con Blender). Para ello se utilizaron dos cálculos del error, un absoluto utilizando la fórmula(1), comparando la diferencia entre los valores y obtenidos para los puntos x destinados a este cálculo (los puntos sobrantes) en cada modelo, y por otro lado el error relativo, indicando la proporción del error absoluto respecto al valor real, dicho error fue calculado utilizando la fórmula (2)

$$error_{absoluto} = |Y_{exp} - Y_{real}| \quad (1)$$

$$error_{relativo} = \left| \frac{Y_{exp} - Y_{real}}{Y_{real}} \right| * 100\% \quad (2)$$

Donde Y_{exp} es el valor obtenido mediante la interpolación en un punto x_n y Y_{real} el valor que toma el objeto creado en blender en el mismo punto x_n

IV. CURVAS DE BEZIER

A. Teoría

Una curva de Bezier es una curva suave paramétrica generada a partir de dos puntos finales y uno o más puntos de control, dichos puntos no necesariamente pasan por la curva, pero son necesarios para calcular la trayectoria de la dicha curva. Existen diversas curvas de Bezier, entre éstas se encuentran, las curvas lineales, las curvas cuadráticas y las curvas cúbicas.

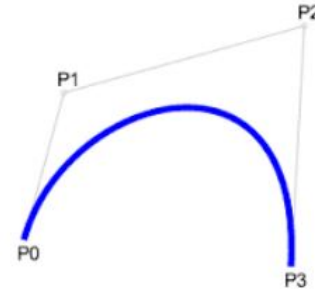


Fig. 6. Curva de Bezier

Las curvas más comunes son las cúbicas que están dadas por cuatro puntos que las definen, donde existe un punto inicial, un punto final, y dos puntos de control que brindan información direccional de la curva, es por ello que estos puntos jamás pasan por la ésta. La forma paramétrica de la curva cúbica de Bezier está definida como:

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

B. Solución

Como se mencionó con anterioridad, la interpolación por Curvas de Bezier se realizó utilizando la funcionalidad *bezier* incluida en la librería del mismo nombre de la herramienta R. Para cada uno de los niveles se utilizó un total de puntos presentado en I, dando como resultado la figuras 7, 8, 9 y 10, donde los trazos rojos representan los resultados de la

interpolación mientras que los puntos negros representan los puntos de control.

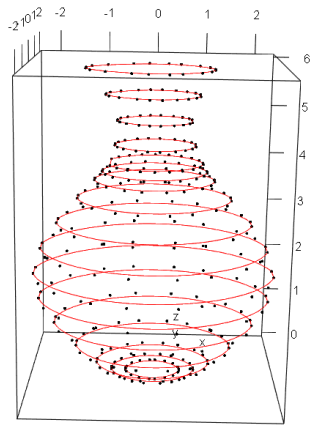


Fig. 7. Vista lateral del jarrón por Bezier

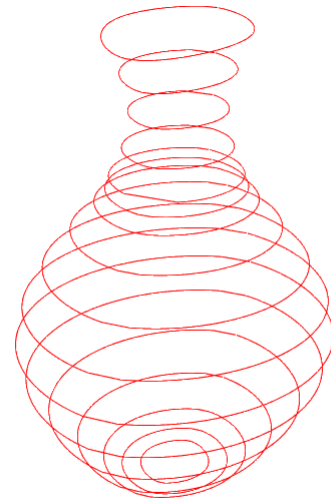


Fig. 10. Vista simple del jarrón por Bezier

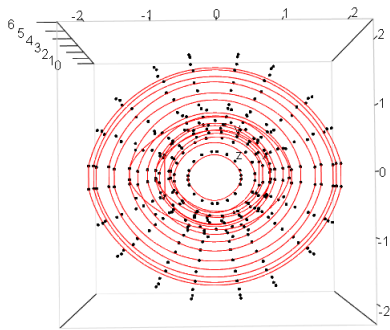


Fig. 8. Vista superior del jarrón por Bezier

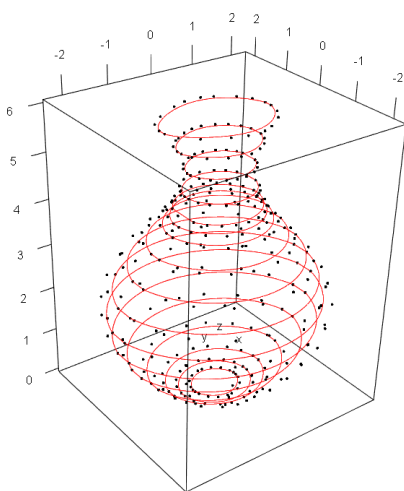


Fig. 9. Vista general del jarrón por Bezier

C. Errores obtenidos

Las curvas obtenidas por este método presentaron un error absoluto promedio de $0,138m$ y un error relativo promedio de 17.392% , respecto al objeto real generado en Blender, dichos errores fueron calculados con las formulas (1) y (2) respectivamente. Estos datos se presentan con mayor detalle en la Tabla II

TABLE II
TABLA DE ERRORES POR NIVEL PARA BEZIER

Nivel (z)	Error Absoluto promedio [m]	Error Relativo promedio [%]
0,000008	0,06675656	19,03611
0,020937	0,0906147	19,4298
0,167437	0,1180999	17,86383
0,523222	0,1588065	17,74469
1,004578	0,202032	17,67188
1,506862	0,2356097	17,63414
2,009147	0,2498777	17,62122
2,511432	0,2385451	17,63135
3,013716	0,2062944	17,66646
3,495073	0,1626778	17,73681
3,850857	0,1257466	17,83588
4,017666	0,1090363	17,90364
4,18077	0,07219691	12,36206
4,526049	0,07427511	15,43491
4,993195	0,06532417	18,33652
5,480656	0,07125969	15,49964
5,968112	0,09894211	18,26316
Promedio	0,138005603	17,39247647

V. SPLINES

A. Teoría

La definición matemática de un Spline hace referencia a una curva diferenciable que está formada por pequeños segmentos de línea definidos por polinomios. La interpolación

con Splines es comúnmente usada porque es muy precisa y se pueden obtener resultados satisfactorios con el uso de simples polinomios. Igual que en el caso de las curvas de Bezier, los Splines pueden ser lineales, cuadráticos y cúbicos. En la interpolación segmentaria cuadrática, los splines se construyen a través de polinomios de grado 2:

$$P(x) = ax^2 + bx + c \quad (4)$$

B. Solución

Para este caso, la interpolación por Splines se realizó utilizando las funcionalidades *spline* y *splinefun* incluidas directamente en la herramienta *R*. Para cada uno de los niveles se utilizó un total de puntos presentado en I, dando como resultado las figuras 11, 12, 13 y 14, donde los trazos azules representan los resultados de la interpolación, mientras que los puntos negros representan los puntos de control que dieron origen a la interpolación.

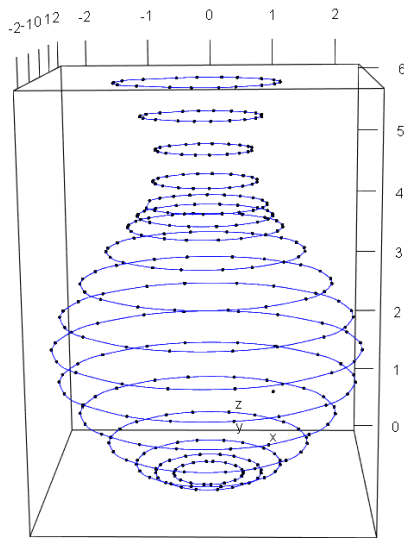


Fig. 11. Vista lateral del jarrón por Splines

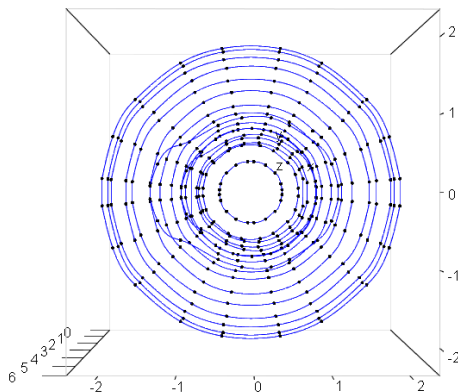


Fig. 12. Vista superior del jarrón por Splines

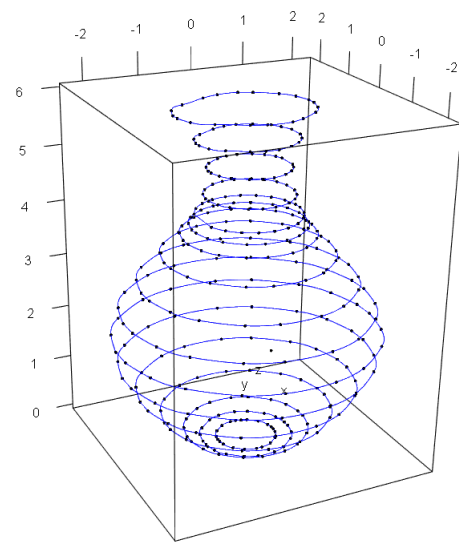


Fig. 13. Vista general del jarrón por Splines

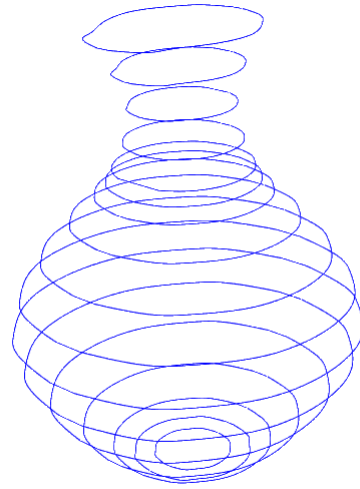


Fig. 14. Vista simple del jarrón por Splines

C. Errores obtenidos

Nuevamente para este caso se utilizó un procedimiento similar al usado para el cálculo de errores en la interpolación por Bezier. Los errores obtenidos son de $0.0154m$ para el error absoluto promedio y un 3.77% para el error relativo promedio, nuevamente estos errores fueron calculados con las formulas (1) y (2) respectivamente. Estos datos se presentan con mayor detalle en la Tabla III

TABLE III
TABLA DE ERRORES POR NIVEL PARA SPLINES

Nivel (z)	Error Absoluto promedio [m]	Error Relativo promedio [%]
0,000008	0,01256446	4,31288
0,020937	0,0169863	5,72114
0,167437	0,0114349	3,105014
0,523222	0,01537643	3,069633
1,004578	0,01956166	3,048217
1,506862	0,02281318	3,037164
2,009147	0,02419439	3,033342
2,511432	0,0230983	3,036429
3,013716	0,01998268	3,047482
3,495073	0,01577728	3,07085
3,850857	0,01221745	3,104125
4,017666	0,01060638	3,127023
4,18077	0,005799234	1,880996
4,526049	0,007821548	2,844628
4,993195	0,01215499	7,668555
5,480656	0,01214308	4,727198
5,968112	0,02013357	6,263633
Promedio	0,015450931	3,770488765

VI. CONCLUSIONES

Como era de esperarse la interpolación por Splines obtuvo mejores resultados (basados en la magnitud de los errores), esto se da debido a la naturaleza del problema, donde los puntos eran extraídos de forma sistematizada y hacían parte del objeto real, lo que pone en desventaja al método de curvas Bezier dado que éste requiere de puntos de control externos a la curva interpolada. Sin embargo, sorprende la proporción del error que tuvo el método de Bezier, pues se esperaba que este fuese mayor dada la cantidad de puntos y las restricciones del problema, esto soportado en los errores absolutos, donde las cifras son pequeñas, menores a $0.2m$, aunque por el tamaño del objeto, las proporciones del caso aumentan la percepción del error por medio del error relativo, que si es considerable.

Por otro lado, se evidencia que los errores son mayores en los extremos verticales de la figura, por un lado, en la parte superior, se debe a la forma irregular que esta tiene, siendo esta asimétrica respecto al eje y a diferencia de las partes inferiores. En el caso de la parte inferior, el error se puede deber a la menor cantidad de puntos disponibles y la aproximación realizada de la extraña forma que toma el objeto real en dicha parte, que se asemeja más a una circunferencia punteada que a una circunferencia completamente uniforme, como sucede en el centro del objeto, donde en efecto, los errores tienen una ligera disminución debido a un comportamiento más predecible por parte del objeto real.

Para finalizar, gráfica y estéticamente, los dos métodos arrojan soluciones bastante aceptables, donde el método de Bezier obtiene una forma más aproximada a la real en la parte superior o la boca del Jarrón, mientras que el método de Splines genera mejores resultados en el centro o cuerpo (aunque obviamente esto es una apreciación subjetiva que puede tener cada persona y es respetable).

REFERENCES

- [1] De la Fuente O'Connor, J. L., 2017. Ingeniería De Los Algoritmos Y Métodos Numéricos. 2nd ed. Círculo rojo.