

Despliegue de curvas algebraicas en coordenadas baricéntricas.

Graphic display of algebraics curves in barycentrics coordinates.

Tovar, Francisco* ; Daza, Julio; Rivas, Luis

Departamento de Matemática, Facultad de Ciencias y Artes, Universidad Metropolitana, Caracas, 1073, Venezuela.

*ftovar@unimet.edu.ve

Resumen

Basados en el trabajo realizado por K. C. Hui y Z. H. Jiang (Hui y col., 1999) sobre el despliegue de superficies en coordenadas tetraédricas, se aplicaron las ideas básicas de ese artículo para desarrollar un algoritmo que permite el despliegue de curvas algebraicas en coordenadas baricéntricas. Con el algoritmo desarrollado se visualiza el segmento de curva contenida en el interior de un triángulo, cuyos vértices definen unas coordenadas baricéntricas del plano, desechando el resto del gráfico de la curva. Los sistemas de graficación estándar que tienen programas como Maple, MatLab, Mathematica, etc., grafican curvas implícitas en sistemas de coordenadas Cartesianas, incluso en coordenadas polares. Sin embargo, para el diseño de splines y otras curvas de CAGD (Computer Aided Geometric Design) es conveniente el despliegue de segmentos de curvas implícitas en coordenadas baricéntricas. El algoritmo desarrollado se resume de la siguiente forma: Dado un triángulo en el plano, se definen las coordenadas baricéntricas del mismo y se procede a refinar el triángulo en cientos de subtriángulos más pequeños. Este primer refinamiento sirve para evaluar la curva en los vértices de los subtriángulos y de acuerdo a ciertos criterios cada subtriángulo es clasificado. La clasificación inicial está definida en tres grupos: Triángulos-Semillas, Triángulos-Frutas y Triángulos-Nulos, estos últimos son la mayoría y se descartan porque no contienen partes de la curva. Los Triángulos-Semillas se utilizan para graficar la curva y los Triángulos-Frutas pasan a un proceso adicional de refinamiento y regresan a la rutina inicial de clasificación. Finalmente, con los subtriángulos clasificados como Triángulos-Semillas, se construye un segmento de recta en el interior del subtriángulo, con los puntos de corte de los lados del subtriángulo y la curva. Los puntos de corte se aproximan mediante un proceso simple de bisección. La unión de los segmentos de recta sirven para construir una aproximación del trazo de la curva.

Palabras clave: Coordenadas baricéntricas, método de bisección, curvas algebraicas, triángulos.

Abstract

Based on the work done by K. C. Hui and Z. H. Jiang (Hui y col., 1999) on the deployment of surfaces in tetrahedral coordinates, we apply the basic ideas of that article to develop our algorithm which allows the deployment of algebraics curves in barycentrics coordinates. With the developed algorithm, the segment of curve contained inside a triangle is displayed, whose vertices define barycentric coordinates of the plane, discarding the rest of the curve. Standard graphing systems that have programs such as Maple, MatLab, Mathematica, etc., graph implicit curves in rectangular systems, even in polar coordinates, however, for the design of splines and other CAGD curves, it is convenient to display of segments of implicit curves in barycentric coordinates. The developed algorithm is summarized in the following way: Given the triangle in the plane that defines the barycentric coordinates thereof, the triangle is refined into hundreds of smaller sub-triangles. This first refinement serves to evaluate the curve in the vertices of the subtriangle and according to certain criteria each subtriangle is classified. The initial classification is defined in three groups: Triangles-Seeds, Triangles-Fruits and Triangles-Nulls, the latter being the majority and discarded because they do not contain parts of the curve. Triangles-Seeds are used to plot the curve and the Triangles-Fruits go through an additional refinement process together with the initial classification process. Finally, with the sub-triangles classified as Triangles-Seeds, a line segment is created between the cut points of the sides of the sub-triangle and the curve. These cut points will be approximated by a simple bisection process. The union of the line segments works to build an approximation of the curve.

Key words: Barycentric coordinates, bisection process, algebraic curves, triangles.

1 Introducción.

A inicios de la década de los 60 surge una nueva disciplina: *Diseño Geométrico Asistido por Computadora*, más conocido por sus siglas en inglés **CAGD** (*Computer Aided Geometric Design*), como resultado de la unión de la Geometría y la Computación. A partir de los trabajos de P. Bézier y P. de Casteljaou hubo una revolución en esta área del conocimiento.

Sin embargo, el uso de las curvas algebraicas definidas implícitamente (Farin 2002), en la solución de problemas de CAGD, comenzó a mediados de la década de los 80 (Chandler 1988). Dentro de este conjunto de curvas, son de especial importancia las llamadas por Chandrajit Bajaj (Bajaj y col., 1994, Bajaj 1992) con el nombre de **A-Spline** o **Splines Algebraicos**, que son curvas definidas por tramos, donde cada sección es una curva algebraica definida implícitamente, conectadas con cierto grado de suavidad, por ejemplo rectas tangentes continuas o curvatura continua.

Los splines que consisten de curvas paramétricas han sido estudiados a lo largo de los últimos cincuenta años, teniendo mucha importancia las curvas de Bézier. Por otra parte, los splines formados por curvas dadas por ecuaciones implícitas no habían recibido mucha atención por parte de los investigadores.

En 1985 Sederberg (Sederberg 1985) inicia el estudio de los A-Splines, los cuales resultan ser una herramienta muy atractiva en la modelación geométrica, pues no se requieren cálculos vinculados con cambios de parametrización. Entre las ventajas de trabajar con curvas implícitas en vez de las curvas paramétricas racionales, se destaca que el conjunto de curvas definidas implícitamente es estrictamente mayor que las curvas parametrizadas racionalmente, es decir, el conjunto de las curvas parametrizadas racionalmente está contenido estrictamente en el conjunto de las curvas implícitas (esto ocurre para curvas de grado mayor o igual que tres). En la práctica los parámetros libres adicionales que se obtienen al emplear la representación implícita, se pueden utilizar para imponer condiciones extras de interpolación o restricciones geométricas, aproximar una curva compleja con el menor número de parámetros, lograr un mejor ajuste de datos, garantizar un mayor orden de suavidad. Usando la ecuación implícita es muy fácil resolver el problema de determinar si un punto dado se encuentra o no sobre la curva. En el caso que el punto no yace sobre la curva, se puede determinar de que lado de la misma se encuentra. Otro problema interesante de resolver es el de hallar los puntos sobre la curva más cercanos a un punto dado en el plano y calcular la distancia entre ellos. La solución de estos problemas usando la expresión paramétrica de una curva, es mucho más costosa computacionalmente.

Aprovechando que un sistema de coordenadas baricéntricas es un caso particular de coordenadas proyectivas (Fernández 1999, Montes de Oca 2000), se utilizan las herramientas de la geometría proyectiva para enfocar ciertos

problemas, fundamentalmente en la visualización de curvas implícitas mediante algunos algoritmos.

2 Despliegue de curvas algebraicas.

Dados tres puntos no alineados V_1, V_2, V_3 , se define el triángulo T , donde $V_i = (x_i, y_i)$ para $i = 1, 2, 3$, son las coordenadas de los vértices del triángulo.

Cualquier punto P en el interior de T se expresa en coordenadas cartesianas como:

$$P = \lambda_1 V_1 + \lambda_2 V_2 + \lambda_3 V_3$$

mientras que en coordenadas baricéntricas P se expresa de la siguiente forma:

$$P = [\lambda_1, \lambda_2, \lambda_3]$$

donde,

$$\sum_{i=1}^3 \lambda_i = 1 \quad y \quad \lambda_i \geq 0.$$

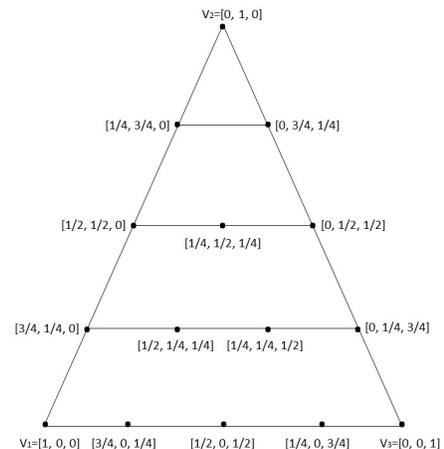


Fig. 1. Ejemplo de las coordenadas de los puntos en un triángulo baricéntrico.

Cualquier punto fuera del triángulo T puede también ser expresado en coordenadas baricéntricas, siempre que P sea una combinación afín de los V_i y en este caso, no todos los λ_i son positivos.

El algoritmo propuesto consta de los siguientes pasos:

- 1) Dado un número entero positivo N , se subdividen los tres lados del triángulo N veces, creando así un refinamiento de N^2 subtriángulos dentro del triángulo principal T , este es el primer refinamiento o subdivisión uniforme de T de orden N . Cada subtriángulo se denota por $T_j = [V_{j1}, V_{j2}, V_{j3}]$ para $j = 1, \dots, N^2$.
- 2) Se evalúa la función que define la curva en cada vértice de los triángulos T_j con $j = 1, \dots, N^2$ y se clasifica cada triángulo T_j como Frutas-Semillas ó Frutas-Nulos.
- 3) A estos dos conjuntos de subtriángulos, se les aplica una subrutina diferente (las cuales serán explicadas en la

sección 2.2.3) y de acuerdo a los criterios establecidos en cada subrutina, los subconjuntos anteriores se clasifican en: Semillas, Frutas y Nulos. Los triángulos que se clasifican **Nulos** son aquellos que no contienen parte de la gráfica de la curva, no se almacenan, simplemente se descartan luego de ser clasificados. Las **Semillas** son los triángulos que contienen, sin ambigüedad, parte del gráfico de la función, y los triángulos catalogados como **Frutas** pasan a un nuevo proceso de refinamiento de orden 2, ya que no está bien definida la intersección del triángulo con la curva y se necesita un refinamiento local. Una vez refinados se realizan los pasos 2 y 3 hasta que desaparezcan las Frutas por completo, quedando solo los triángulos Semillas, o hasta que las Frutas satisfagan las condiciones del criterio de parada.

- 4) Finalmente queda un conjunto de subtriángulos Semillas. Sobre estos triángulos se realiza un proceso simple de bisección, para aproximar los puntos de intersección entre la curva y cada lado del triángulo, optimizando el despliegue de la curva.
- 5) El gráfico de la curva, consiste en la familia de pequeños segmentos de rectas contenidos en los subtriángulos que resultaron Semillas.

2.1 Subdivisión uniforme del triángulo.

Una subdivisión de T consiste en particionar $N - veces$ cada lado del mismo, y así, crear N^2 subtriángulos, los cuales quedarán etiquetados y en orden.

En este proceso se fracciona cada lado del triángulo y se escoge cada punto como vértice, tal como se ilustra en la Figura 2.

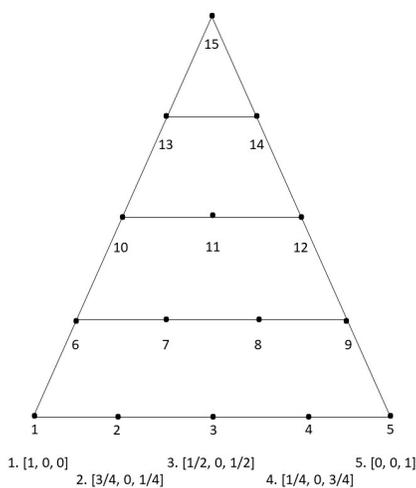


Fig. 2. Discretización del Triángulo para $N = 4$.

Luego se realiza un etiquetado de todos los subtriángulos que resultan de la partición inicial. El etiquetado funciona como guía para evaluar la función en cada vértice de los

triángulos. Su seguimiento será de izquierda a derecha en cada fila del triángulo principal sin importar si el triángulo está hacia arriba o hacia abajo. El etiquetado del primer refinamiento se muestra en la Figura 3.

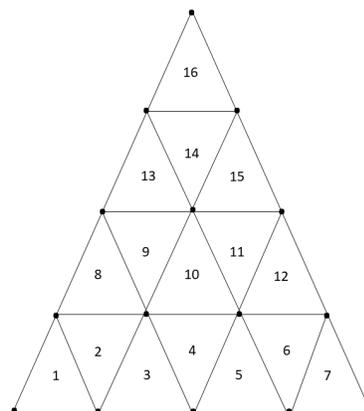


Fig. 3. Etiquetado en orden de los triángulos para $N = 4$.

El etiquetado de todos los subtriángulos que resultan de esta subdivisión, es importante para su uso posterior junto a las correspondientes coordenadas de cada uno de los vértices. La unión de este nuevo conjunto de subtriángulos se corresponde con el triángulo más grande, esto significa que:

$$T = \bigcup T_j$$

con $j = 1, \dots, N^2$. Donde T_j denota el j -ésimo triángulo de la subdivisión.

2.2 Clasificación de los triángulos.

Inicialmente, tomando como referencia el trabajo de K. C. Hui y Z. H. Jiang (Hui y col., 1999), cada subtriángulo T_j , con $j = 1, \dots, N^2$ son clasificados como nulos, frutas ó semillas, a través del siguiente criterio (nótese que al decir que el signo de un vértice de un triángulo es positivo o negativo, se refiere a la evaluación de la función en dicho vértice):

- a) **Semilla**: Triángulo donde existe un vértice con un signo diferente que el resto y contiene parte de la curva.
- b) **Fruta**: Triángulo que contiene parte de la curva pero, posee una intersección ambigua con la curva (la intersección ambigua será definida más adelante, en la sección 2.2.2).
- c) **Nulo**: Triángulo cuyos tres vértices son de signos iguales y no hay intersección con la curva.

Debido a las diferentes intersecciones de los triángulos con la curva implícita $F(x, y) = 0$, en este algoritmo se realiza una clasificación inicial de los triángulos T_j en Frutas-Nulos y Frutas-Semillas, de esta forma resultó un algoritmo diferente al planteado en (Hui y col., 1999).

2.2.1 Clasificación inicial.

Como se estableció anteriormente, la subdivisión resultante de T es $\cup T_j$ con $j = 1, \dots, N^2$. Los vértices de cada triángulo T_j cumplen una función importante, si un lado $[V_a, V_b]$ de un triángulo cualquiera intersecta a la curva en un único punto, eso quiere decir que el sistema:

$$\begin{pmatrix} 1 & 1 \\ F(V_a) & F(V_b) \end{pmatrix} \begin{pmatrix} \lambda_a \\ \lambda_b \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

tiene única solución para $\lambda_a, \lambda_b \geq 0$. Un punto de intersección aproximado P existe entre los vértices V_a y V_b si y solo si $F(V_a)F(V_b) < 0$ y $P = \lambda_a V_a + \lambda_b V_b$ (esto es según el criterio usado en (Hui y col., 1999)). Cabe acotar, que en este trabajo, se optimiza la ubicación de ese punto P de intersección, a través de un proceso simple de bisección.

Para realizar esta clasificación inicial, se evalúa la función $F(x, y)$ en cada vértice V_{ji} con $j = 1, \dots, N^2, i = 1, 2, 3$, y se clasifican todos los triángulos donde las evaluaciones en los tres vértices tengan el mismo signo, esto es:

$$F(V_{j1}) \vee F(V_{j2}) \vee F(V_{j3}) \geq 0$$

o

$$F(V_{j1}) \wedge F(V_{j2}) \wedge F(V_{j3}) \leq 0$$

con $j = 1, \dots, N^2$, como triángulos Frutas - Nulos, en cualquier otro caso, los triángulos T_j se clasifican como Frutas - Semillas.

2.2.2 Intersecciones ambiguas.

Ya se sabe que existe, un único punto de intersección en un lado $[V_{ja}, V_{jb}]$ de un triángulo si y solo si $F(V_{ja})F(V_{jb}) < 0$. Entonces:

$$\begin{pmatrix} \lambda_a \\ \lambda_b \end{pmatrix} = \frac{1}{F(V_{jb}) - F(V_{ja})} \begin{pmatrix} F(V_{jb}) \\ F(V_{ja}) \end{pmatrix}$$

para $j = 1, \dots, N^2$.

Si $F(V_{ja})F(V_{jb}) > 0$, no hay solución con $\lambda_a, \lambda_b \geq 0$. Esto implica que no hay intersección entre el lado $[V_{ja}, V_{jb}]$ y la curva. Sin embargo, esto puede que no sea cierto en el caso que la curva tenga alta curvatura (Valdés 2013), como se muestra en la Figura 4.

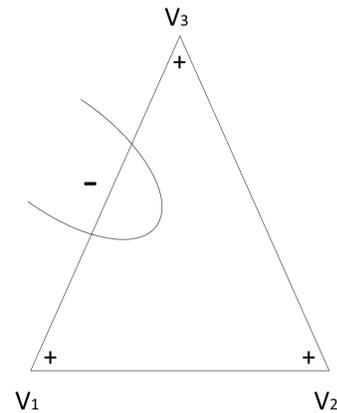


Fig. 4. Ejemplo de una intersección ambigua.

Estos triángulos entran en la clasificación de Frutas-Nulos, pues, a pesar de que satisface la condición $F(V_{ja})F(V_{jb}) > 0$, el triángulo es intersectado por la curva.

También existe el caso donde a pesar de entrar en el conjunto de Semillas, según el criterio dado en (Hui y col., 1999), el gráfico que arrojaría el algoritmo no se corresponde correctamente con el gráfico de la curva. En la Figura 5, se puede apreciar un ejemplo de este caso, donde, a pesar de ser un candidato para semilla, no muestra un gráfico correcto. Se puede notar que si la curva interseca al triángulo de la forma ilustrada en la Figura 5, se realizaría una línea entre los puntos de corte donde se originan los cambios de signo, lo cual no se corresponde con la traza de la curva en el interior del triángulo.

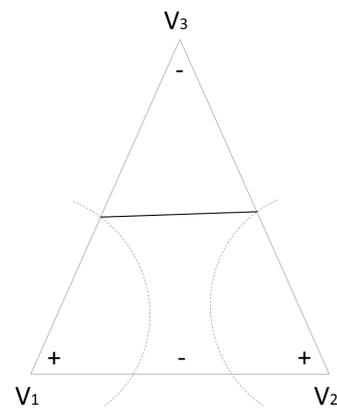


Fig. 5. Otro tipo de intersección ambigua.

Estos son casos que se consideran como intersecciones ambiguas; lo cual motiva la pre-clasificación inicial en Frutas-Nulos y Frutas-Semillas.

2.2.3 Clasificación final.

Una vez obtenidos los conjuntos de triángulos Frutas-Nulos y Frutas-Semillas, se procede a separarlos de forma definitiva a través de dos subrutinas, una para cada conjunto de triángulos.

Para separar los triángulos clasificados como Frutas-Nulos (se recuerda que en este caso, los signos de los tres vértices de cada triángulo son iguales), en Frutas o (excluyente) Nulos se realizan tres evaluaciones:

- Se evalúa la función en el punto medio de cada lado del triángulo.
- Se comparan los signos de las evaluaciones de los puntos medios, con los de la función evaluada en los vértices. Si el signo de la evaluación de alguno de los puntos medios es opuesto al de las evaluaciones de la función en los vértices, entonces ese triángulo pasa a la clasificación Fruta.
- Los triángulos que no satisfacen el ítem b), se les hace una evaluación adicional. Se comparan los valores absolutos de las evaluaciones de los puntos medios y la evaluación de la función en los vértices, de modo que si se cumple la siguiente condición:

$$\left| F\left(\frac{V_{ja} + V_{jb}}{2}\right) \right| < \min\{|F(V_{ja})|, |F(V_{jb})|\} \quad \wedge$$

$$\left| F\left(\frac{V_{ja} + V_{jb}}{2}\right) \right| > \max\{|F(V_{ja})|, |F(V_{jb})|\}$$

donde $j = 1, \dots, n$ y n es el número total de triángulos clasificados Frutas-Nulos. Entonces los triángulos también serán clasificados como Frutas. Los triángulos Nulos simplemente son los que no satisfacen las condiciones de los ítems b) y c), por lo tanto son descartados en su totalidad.

Para separar los triángulos clasificados como Frutas-Semillas (se recuerda que en este caso, los triángulos tienen un signo diferente), en triángulos Frutas o (excluyente) Semillas, se realizan estas dos evaluaciones:

- Se determinan cuales vértices tienen el mismo signo, esto es:

$$F(V_{ja})F(V_{jb}) > 0$$

para $j = 1, \dots, m$, donde m es el número total de triángulos clasificados como Frutas - Semillas.

- Sea $VM_j = F\left(\frac{V_{ja} + V_{jb}}{2}\right)$ la evaluación del valor medio entre los vértices V_{ja} y V_{jb} en el triángulo T_j para $j = 1, \dots, m$. Se consideran los tres siguientes casos:

- El valor de la evaluación de la función en el vértice es de signo opuesto al de VM_j :

$$F(V_{ja})VM_j \leq 0$$

- El valor absoluto de VM_j es menor que el mínimo del valor absoluto de la evaluación de la función en cada vértice de ese lado del triángulo:

$$|VM_j| < \min\{|F(V_{ja})|, |F(V_{jb})|\}$$

- El valor absoluto es mayor que el máximo del valor absoluto de la evaluación de la función en cada vértice de ese lado del triángulo:

$$|VM_j| > \max\{|F(V_{ja})|, |F(V_{jb})|\}$$

para $j = 1, \dots, m$. Si la evaluación del triángulo cumple con alguna de las condiciones dadas en los ítems b.1) o b.2) o b.3), entonces será clasificado como triángulo Fruta. En cualquier otro caso pasa directamente a triángulos Semillas, los cuales serán tratados en la sección 2.5.

2.3 Refinamiento y criterio de parada.

Los triángulos clasificados finalmente como Frutas, pasan por un refinamiento de orden 2 (o con $N = 2$) para obtener un total de 4 subtriángulos por cada Fruta. Este nuevo conjunto de triángulos pasan al proceso de clasificación a partir del ítem 2.2. Se establece un criterio de parada para este proceso con tres condiciones:

- El número máximo de refinamientos. En la práctica se estableció que a lo sumo 15 refinamientos de los triángulos clasificados como Frutas, muestran buenos resultados en curvas que tienen dificultades propias para su despliegue gráfico, como lo muestra la Figura 6.

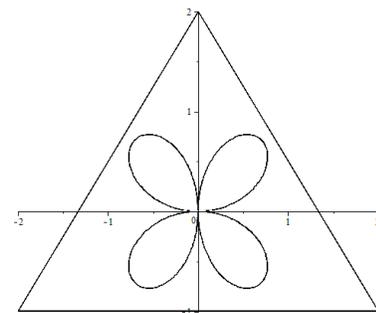


Fig. 6. Despliegue del gráfico de una curva con punto singular de orden 4 en el origen.

- El tamaño mínimo de cada triángulo refinado. Para determinar el tamaño mínimo, primero se define el tamaño del triángulo principal T , el cual se denota por L_T y se define como la longitud máxima de los lados de T . Análogamente se define L_{T_j} como el máximo de la

longitud de los lados de un triángulo clasificado como Fruta. Se considera $\epsilon = \frac{L_T}{10000}$, si L_{T_j} es menor que ϵ entonces, se detiene el proceso.

Se considera ese número para ϵ ya que representa un triángulo lo suficientemente pequeño para que no muestre detalles o errores en el despliegue de la curva.

- c) El conjunto de triángulos Frutas es vacío. Simplemente se procede a desplegar la curva, ya que no se tiene que aplicar ningún otro refinamiento.

2.4 Algoritmo para desplegar curvas algebraicas en coordenadas baricéntricas.

- Input

$F(x, y)$ - Ecuación implícita de la curva.

T - Coordenadas del triángulo inicial.

N - Número inicial de subdivisiones.

- Algoritmo

begin

Refinamiento de orden N para generar N^2 subtriángulos.

Clasificación inicial

if $F(V_{ja})F(V_{jb}) \geq 0$ **then**

Add T-Frutas-Nulos

else

Add T-Frutas-Semillas

end if

Separación de los T-Frutas-Nulos

if $F(V_{ja})VM_j \geq 0$ **then**

Add T-Nulos

elif $|VM_j| < \min\{|F(V_{ja})|, |F(V_{jb})|\}$ **or**
 $|VM_j| > \max\{|F(V_{ja})|, |F(V_{jb})|\}$ **then**

Add T-Frutas

end if

Separación de los T-Frutas-Semillas

if $F(V_{ja})VM_j \leq 0$ **or** $|VM_j| <$
 $\min\{|F(V_{ja})|, |F(V_{jb})|\}$ **or** $|VM_j| >$
 $\max\{|F(V_{ja})|, |F(V_{jb})|\}$ **then**

Add T-Frutas

else

Add T-Semillas

end if

for T-Frutas **from** 1 **to** 15

while Frutas > 0 **or** $L_{T_j} > \epsilon$ **do**

Refinamiento de orden $N = 2$

Clasificación Inicial

Separación Frutas-Nulos

Separación Frutas-Semillas

end while

end for

Método de bisección sobre los T-Semillas.

Generador de segmentos: Se genera el segmento de recta que une los dos puntos obtenidos por el Método de Bisección.

end.

2.5 Aproximación lineal de la curva.

El gráfico de la curva es aproximado por muchos segmentos pequeños de rectas. Cada segmento está contenido en un triángulo Semilla, recordemos que estos triángulos se caracterizan por tener un vértice con el signo opuesto a los otros dos. Se identifican los lados del triángulo adyacentes al vértice con signo opuesto. Estos lados intersecan a la curva. Para identificar cada lado, se evalúa la función en cada vértice y se comparan los signos, si $F(V_{ja})F(V_{jb}) < 0$ para $j = 1, \dots, l$, (l es el número total de triángulos Semillas), entonces, se aproxima el punto de intersección de la curva con cada lado, usando el método de bisección.

El método de bisección genera un punto de corte cercano entre la intersección de la curva y el lado del triángulo, esto implica que para cada triángulo Semilla se obtendrán dos puntos de corte, entonces al tener dos puntos se construye el segmento de recta que los une, y se realiza la aproximación lineal de la curva con todos los segmentos de rectas contenidos en los triángulos Semilla.

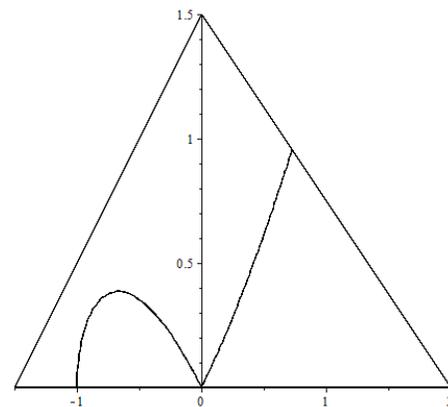


Fig. 7. Despliegue final de una curva algebraica por medio del algoritmo.

Para obtener el gráfico de la curva mostrada en la Figura 7, se consideró $N = 50$ para subdividir uniformemente el triángulo principal (obteniendo un total de 2.500 subtriángulos) y se trabajó con la curva $f(x, y) = x^3 - x^2 + y^2$, que tiene un punto singular en el $(0, 0)$. A lo largo del desarrollo del algoritmo, se llegó a la conclusión que iterar el método de bisección tres veces es suficiente para cada segmento del triángulo Semilla, ya que la aproximación lineal de la curva se

ajusta a la curva original.

En la Figura 8, se observa el despliegue de la curva $f(x, y) = x^3 - x^2 + y^2$, donde no se aplica el proceso de bisección sobre cada triángulo Semilla. Este ejemplo muestra la necesidad de usar algún método que permita aproximar más eficientemente la intersección de la curva con cada segmento de los triángulos Semillas. En particular se escogió el método de bisección por sus buenos resultados en las curvas estudiadas.

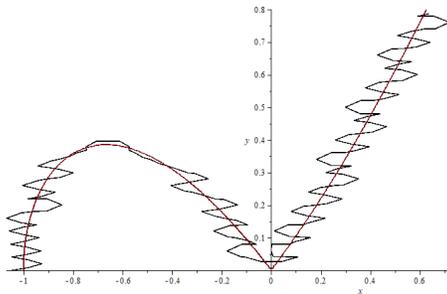


Fig. 8. Despliegue final de una curva algebraica sin aplicar el método de bisección.

3 Resultados.

A continuación se muestran algunos resultados obtenidos con el algoritmo propuesto. Se tienen gráficos de curvas complejas, debido a sus puntos singulares y alta curvatura.

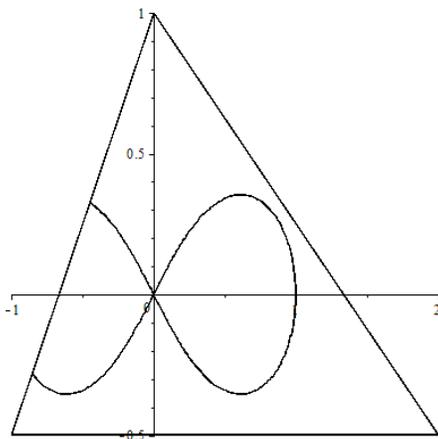


Fig. 9. La curva $(x^2 + y^2)^2 - (x^2 - y^2) = 0$ (Lemniscata).

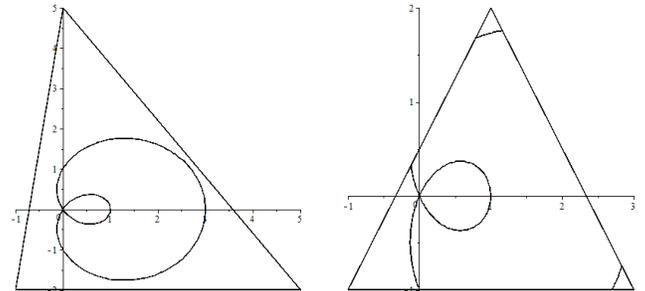


Fig. 10. La curva $(x^2 - 2x + y^2)^2 - (x^2 + y^2) = 0$ (Cardioide), sobre dos triángulos diferentes.

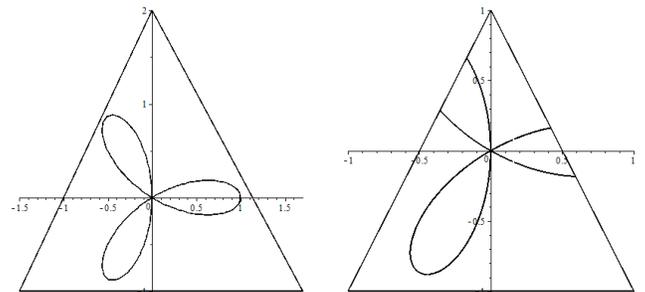


Fig. 11. La curva $(x^2 + y^2)^2 - x(x^2 - 3y^2) = 0$ (Flor de 3 Hojas), sobre dos triángulos diferentes.

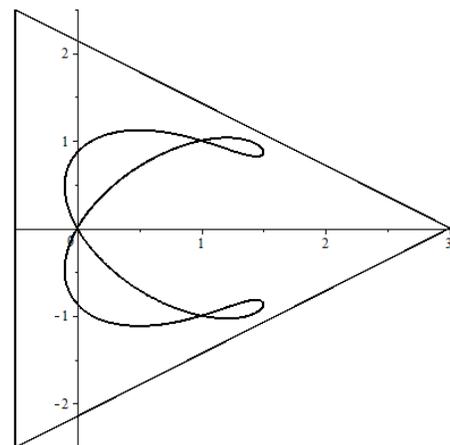


Fig. 12. La curva $(y^2 - x^2)(x - 1)(2x - 3) - 4(x^2 + y^2 - 2x)^2 = 0$, curva con tres puntos singulares.

4 Conclusiones

Se presentó un algoritmo que permite el despliegue gráfico de una curva algebraica en el interior de un triángulo dado, el cual define unas coordenadas baricéntricas en el plano. Este algoritmo es una adaptación del trabajo de K. C. Hui y Z. H. Jiang (Hui y col., 1999) al plano. En el proceso se consideran

las intersecciones ambiguas entre los triángulos y la curva, causados por singularidades y alta curvatura. Para eliminar estos errores se utiliza un proceso de refinamiento de los triángulos, lo cual permite un despliegue gráfico más aproximado a la curva. Para aproximar cada segmento de curva, contenida en cada triángulo, es necesario calcular el punto de corte de la curva con los lados del triángulo. La propuesta dada en (Hui y col., 1999), para encontrar estos puntos, no derivó en buenos gráficos, por lo que se utilizó el método de bisección.

Los tiempos de respuesta del algoritmo con respecto a los ejecutores del software Maple no fueron comparados, sin embargo el algoritmo trabaja eficientemente para curvas complicadas.

Cuando se trabaja con *A-Splines* de grado 3 (Bajaj y col., 1994), es necesario desplegar la gráfica de la curva en el interior de cada triángulo de control, el algoritmo desarrollado en este artículo resulta muy útil para este tipo de despliegue.

Referencias

- Bajaj CL, Xu G, 1994, Data Fitting with Cubic A-splines. Computer Graphics International, CGI94, Melbourne, Australia.
- Bajaj CL, 1992, The Emergence of Algebraic Curves and Surface in Geometric Design, Tech. Report Purdue University, CS-TR-92-056.
- Chandler RE, 1988, A Tracking Algorithm for Implicitly Defined Curves, IEEE Computer Graphics and Applications.
- Farin G, 2002, Curves and Surfaces for CAGD, Arizona State University, pp 216-218.
- Fernández L, 1999, Plano Afín, Coordenadas Baricéntricas y Plano Proyectivo, E.T.S.I. Navales, Universidad Politécnica de Madrid, E-28040-Madrid.
- Hui KC, Jiang ZH, 1999, Tetrahedra Based Adaptive Polygonization of Implicit Surface Patches. The Chinese University of Hong Kong, vol 18, pp 57-68.
- Montes de Oca A, 2000, Geometría Métrica y Proyectiva en el Plano con Coordenadas Baricéntricas. Algunos Tópicos, Versión 2.181231712
- Sederberg TW, 1985, Piecewise Algebraic Surface Patches, Computer Aided Geometric Design 2, pp 53-59.
- Valdés A, 2013, Notas de Geometría Diferencial con Aplicaciones, pp 47.

Recibido: 06 de septiembre de 2021

Aceptado: 12 de noviembre de 2021

Tovar, Francisco: Doctor en Ciencias, mención Matemática. Profesor Titular Jubilado de la Universidad Central de Venezuela. Profesor Titular de la Universidad Metropolitana.
<https://orcid.org/0000-0002-5662-5160>.

Daza, Julio: MSC en Ciencias, mención Matemática. Profesor Asociado de la Universidad Metropolitana, Correo electrónico: jdaza@unimet.edu.ve.
<https://orcid.org/0000-0003-0685-988X>.

Rivas, Luis: Licenciado en Matemática de la Universidad Central de Venezuela. Profesor Instructor de la Universidad Metropolitana, Correo electrónico: lmrivas@unimet.edu.ve.
<https://orcid.org/0000-0002-8695-8337>.