

**IMPLEMENTACIÓN COMPUTACIONAL DE UN ALGORITMO DE
OPTIMIZACIÓN PARA EL CÁLCULO DE SUPERFICIES DE REVOLUCIÓN**

María A. Jamilis Ricladoni *, **Fernando Wasylszyn †**, **Patricia M. Gauzellino ζ**,
Angel F. Queizan +

* Universidad Tecnológica Nacional. Regional La Plata
Calle 60 y 125, 1900 La Plata, Argentina
alejamilis@gmail.com

† Universidad Tecnológica Nacional. Regional La Plata
Calle 60 y 125, 1900 La Plata, Argentina
ferwasy@yahoo.com.ar

ζ Fac. de Cs. Astronómicas y Geofísicas, U.N.L.P.
Paseo del Bosque s/n, 1900 La Plata, Argentina
gauze@fcaglp.fcaglp.unlp.edu.ar

+ Universidad Tecnológica Nacional. Regional La Plata
Calle 60 y 125, 1900 La Plata, Argentina
aqueizan@frlp.utn.edu.ar

Palabras claves: Optimización, FAIPA, Superficie mínima, Visualización

Resumen: *Es bien conocido que el tratamiento de problemas que desean minimizar una función sujeta a restricciones de igualdad y desigualdad es de interés para el diseño ingenieril. Con esta motivación, el presente trabajo consiste en el estudio e implementación computacional del algoritmo de punto interior FAIPA (Feasible Arc Interior Point Algorithm) para optimización no lineal.*

Como una aplicación del algoritmo, se calcula la mínima superficie de revolución que debe pasar por una determinada cantidad de puntos relacionados mediante B-spline cúbicos satisfaciendo ciertas restricciones geométricas.

1 INTRODUCCIÓN

Se ha estudiado e implementado computacionalmente el algoritmo de punto interior FAIPA (Feasible Arc Interior Point Algorithm) para optimización no lineal, el cual es una herramienta matemática para optimización de diseño en ingeniería.¹

En el presente trabajo se busca establecer una generatriz que rota en torno al eje de las z, de manera tal que la superficie desarrollada sea mínima satisfaciendo ciertas restricciones en desigualdad.

La generatriz se define mediante una técnica de interpolación B-Spline, para lo cual definimos inicialmente un conjunto de puntos de control que definen una poligonal que nos conduce a establecer los coeficientes de los polinomios propuestos de Bezier.^{2,3,4}

Como criterio de diseño se establece que los puntos de control se desplacen en ciertas direcciones prefijadas, buscando reducir la superficie de revolución desarrollada y satisfaciendo restricciones que limitan el desplazamiento de estos puntos.

Además, se ha desarrollado un algoritmo que permite visualizar la curva generatriz junto con los puntos de interpolación en el plano (X,Z) y la superficie de revolución generada por la misma.⁵

2 DESARROLLO TEÓRICO

2.1 Interpolación B-Spline de la generatriz

La interpolación de la generatriz se va a definir mediante m polinomios segmentarios en función de un parámetro $0 \leq v \leq 1$, los cuales tienen la forma

$$\underline{P}_k(v) = \begin{Bmatrix} P_k^x \\ P_k^z \end{Bmatrix} = (1-v)^3 \begin{Bmatrix} b_{3k-2}^x \\ b_{3k-2}^z \end{Bmatrix} + 3(1-v)^2 v \begin{Bmatrix} b_{3k-1}^x \\ b_{3k-1}^z \end{Bmatrix} + 3(1-v)v^2 \begin{Bmatrix} b_{3k}^x \\ b_{3k}^z \end{Bmatrix} + v^3 \begin{Bmatrix} b_{3k+1}^x \\ b_{3k+1}^z \end{Bmatrix} \quad (1)$$

$$k = 1 \dots m.$$

Además, se exige que exista continuidad de la derivada primera y segunda entre los polinomios $\underline{P}_k(v)$ y $\underline{P}_{k+1}(v)$, es decir,

-Para la derivada primera:

$$\underline{P}_k'(v=1) = \underline{P}_{k+1}'(v=0) \rightarrow \underline{b}_{3k+1} - \underline{b}_{3k} = \underline{b}_{3k+2} - \underline{b}_{3k+1} \quad k = 1 \dots m-1. \quad (2)$$

-Para la derivada segunda:

$$\underline{P}_k''(v=1) = \underline{P}_{k+1}''(v=0) \rightarrow \underline{b}_{3k-1} - 2\underline{b}_{3k} + \underline{b}_{3k+1} = \underline{b}_{3k+1} - 2\underline{b}_{3k+2} + \underline{b}_{3k+3} \quad (3)$$

$$k = 1 \dots m-1.$$

Esta última ecuación permite definir los puntos de control \underline{d}_{k+1} que establecen la poligonal

$$\underline{d}_{k+1} := \underline{b}_{3k} + (\underline{b}_{3k} - \underline{b}_{3k-1}) = \underline{b}_{3k+2} - (\underline{b}_{3k+3} - \underline{b}_{3k+2}). \quad (4)$$

Teniendo en cuenta esta ultima expresión y la condición de continuidad de la derivada primera, se pueden definir los coeficientes de los polinomios en función de los puntos de control

$$\left\{ \begin{array}{l} \underline{b}_{3k-1} = \frac{1}{3} (\underline{d}_{k+1} + 2 \underline{d}_k) \quad \rightarrow k = 1 \dots m, \\ \underline{b}_{3k} = \frac{1}{3} (2 \underline{d}_{k+1} + \underline{d}_k) \quad \rightarrow k = 1 \dots m, \\ \underline{b}_{3k+1} = \frac{1}{6} (\underline{d}_k + 4 \underline{d}_{k+1} + \underline{d}_{k+2}) \quad \rightarrow k = 1 \dots m-1. \end{array} \right. \quad (5)$$

2.2 Planteo del problema de programación no lineal

Las variables de diseño se definen como una magnitudes escalares x_i ($i = 2 \dots m$), las cuales manejan los desplazamientos de los puntos de control \underline{d}_k mediante la fórmula

$$\underline{d}_k^{final} = \underline{d}_k^{anterior} + x_k \underline{m}_k, \quad (6)$$

siendo \underline{m}_k las direcciones de búsqueda predefinidas.

Inicialmente, el problema es encontrar la generatriz parametrizada que hace mínima la siguiente función objetivo

$$\min \left\{ 2 \pi \int_{t_o}^{t_f} x \sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dz}{dt}\right)^2} dt \right\}, \quad (7)$$

la cual expresada en función de los polinomios de Bezier es

$$\min \left\{ 2 \pi \sum_{k=1}^m \int_0^1 P_k^x \sqrt{\left(\frac{dP_k^x}{dv}\right)^2 + \left(\frac{dP_k^z}{dv}\right)^2} dt \right\}. \quad (8)$$

La resolución de la integral se realiza mediante la aplicación de la técnica de cuadratura de Gauss, usando dos puntos de integración, de esta manera se obtiene de la expresión anterior

$$\min \left\{ 2 \pi \sum_{k=1}^m \sum_{p=1}^2 w_i f_k (v_i) \right\} \quad (9)$$

donde

$$f_k (v_i) = P_k^x (v_i) \sqrt{\left(\frac{dP_k^x}{dv} \Big|_{v_i} \right)^2 + \left(\frac{dP_k^z}{dv} \Big|_{v_i} \right)^2}, \quad (10)$$

$$P_k^x (v_i) = (1-v_i)^3 b_{3k-2}^x + 3(1-v_i)^2 v_i b_{3k-1}^x + 3(1-v_i) v_i^2 b_{3k}^x + v_i^3 b_{3k+1}^x, \quad (11)$$

$$\frac{dP_k^x}{dv} \Big|_{v_i} = (-3 + 6 v_i - 3 v_i^2) b_{3k-2}^x + (9 v_i^2 - 12 v_i + 3) b_{3k-1}^x + (-9 v_i^2 + 6 v_i) b_{3k}^x + 3 v_i^2 b_{3k+1}^x, \quad (12)$$

$$\frac{dP_k^z}{dv} \Big|_{v_i} = (-3 + 6 v_i - 3 v_i^2) b_{3k-2}^z + (9 v_i^2 - 12 v_i + 3) b_{3k-1}^z + (-9 v_i^2 + 6 v_i) b_{3k}^z + 3 v_i^2 b_{3k+1}^z. \quad (13)$$

Como restricciones en desigualdad se definen las siguientes

$$\left\| \underline{d}_k^{nuevo} - \underline{d}_k^{anterior} \right\|_2^2 = x_k^2 \left\| \underline{m}_k \right\|_2^2 = x_k^2 \leq R_k^2. \quad (14)$$

Esta últimas acotan el desplazamiento en el entorno de $\underline{d}_k^{anterior}$ a un radio menor o igual a R_k .

Es resumen, el problema de programación no lineal queda definido como

$$\text{Min} \left\{ 2 \pi \sum_{k=1}^m \sum_{p=1}^2 w_i f_k (v_i) \right\} \quad (15)$$

Sujeto a:

$$x_k^2 \leq R_k^2.$$

3 METODOLOGÍA APLICADA EN LA RESOLUCIÓN DEL PROBLEMA DE PROGRAMACIÓN NO LINEAL

Se aplicó el método FAIPA (Feasible Arc Interior Algorithm) desarrollado por el Dr. Jose Herskovits, con el cual se van obteniendo diseños que satisfacen las restricciones en desigualdad y realizando búsquedas a lo largo de arcos; esto presenta una gran ventaja en el caso de poseer restricciones fuertemente no lineales.

Se define como restricción activa a aquella que es menor o igual a cero. Si la restricción es estrictamente menor que cero se la denomina restricción pasiva.

3.1 Aplicaciones realizadas

Ejemplo 1: Se propuso inicialmente una poligonal que define seis polinomios de Bezier, aproximadamente contenida en un círculo con centro en $z = 75 \quad x = 0$ y radio igual a 75. En este caso se tienen 5 puntos de control y por lo tanto igual numero de variables de diseño. Las direcciones de búsqueda fueron definidas por variaciones en la dirección únicamente de X y dejando fijos los puntos extremos. Los puntos de control inicial son $(0,0);(25,4.2893);(50,19.098);(75,75);(50,130.9);(25,145.71)$ y $(0,150)$. Las cotas de las restricciones se determinan por $\pm 5, \pm 5, \pm 25, \pm 30$ y ± 10 , respectivamente.

Los resultados obtenidos son :

- Función objetivo inicial: 59908.7
- Función objetivo final: 27630.5
- Cantidad de iteraciones: 25
- Variable se diseño obtenidas:
 $x = [-4.9999 \quad -4.9999 \quad -49.9999 \quad -29.9999 \quad -9.9999]$

Todas las restricciones son activas.

En la figura 1 se muestran las generatrices inicial y final con sus respectivos puntos de control. Las figuras 2 y 3 representan los resultados obtenidos.

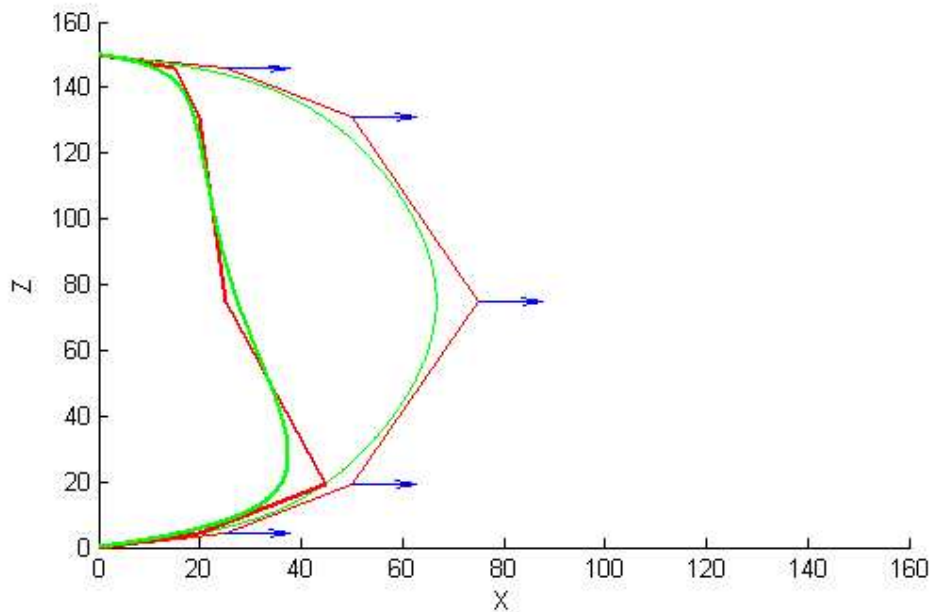


Figura 1. Condiciones iniciales y resultado obtenido

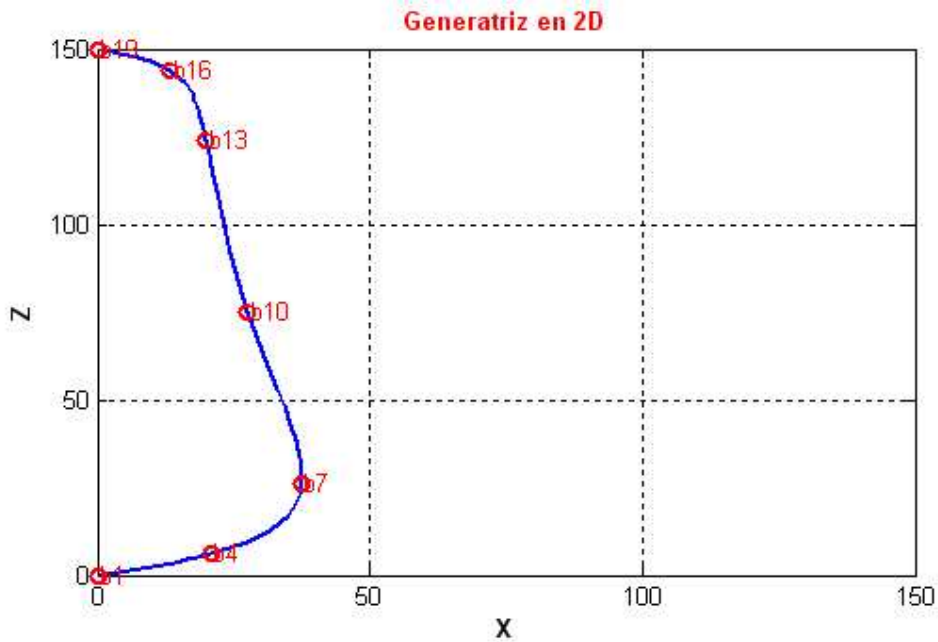


Figura 2. Generatriz obtenida

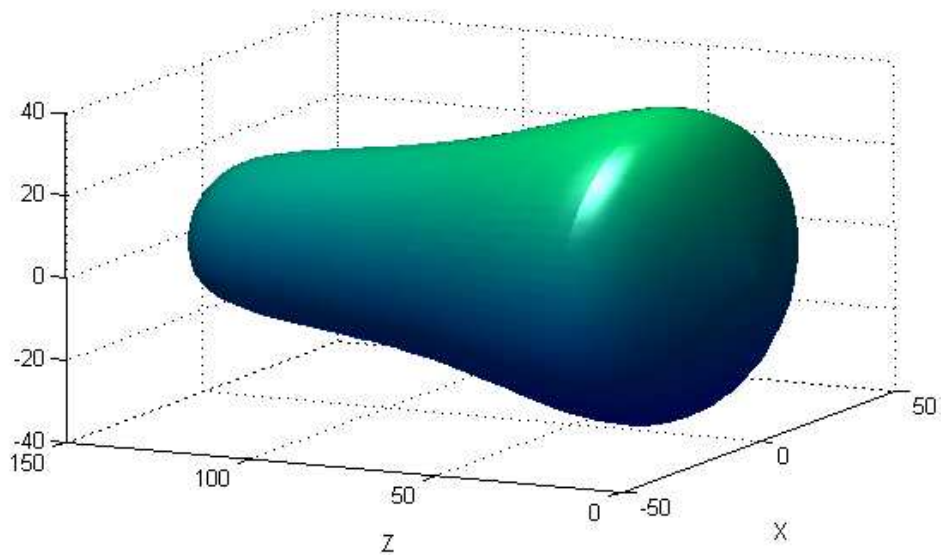


Figura 3. Superficie de revolución obtenida

Ejemplo 2: : Con las mismas condiciones iniciales del Ejemplo 1, si se cambian las restricciones por ± 20 , ± 10 , ± 80 , ± 10 y ± 20 , los resultados obtenidos son :

- Función objetivo inicial: 59908.7
- Función objetivo final: 19229.8
- Cantidad de iteraciones: 19
- Variable se diseño obtenidas:
 $x = [-19.9999 \ -9.9999 \ -79.9999 \ -9.9999 \ -19.9999]$

Todas las restricciones son activas.
Las figuras 4, 5 y 6 ilustran este ejemplo.

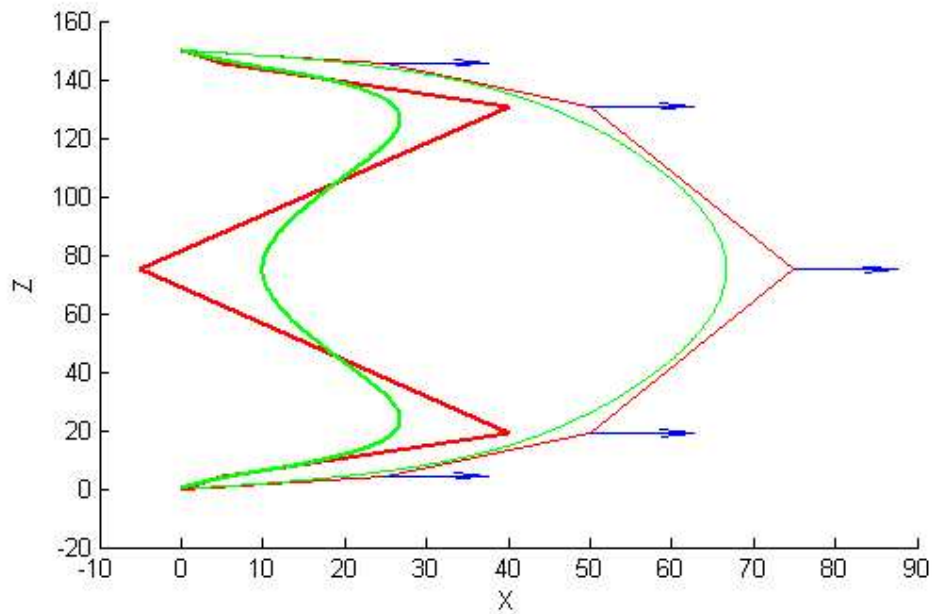


Figura 4. Condiciones iniciales y resultado obtenido

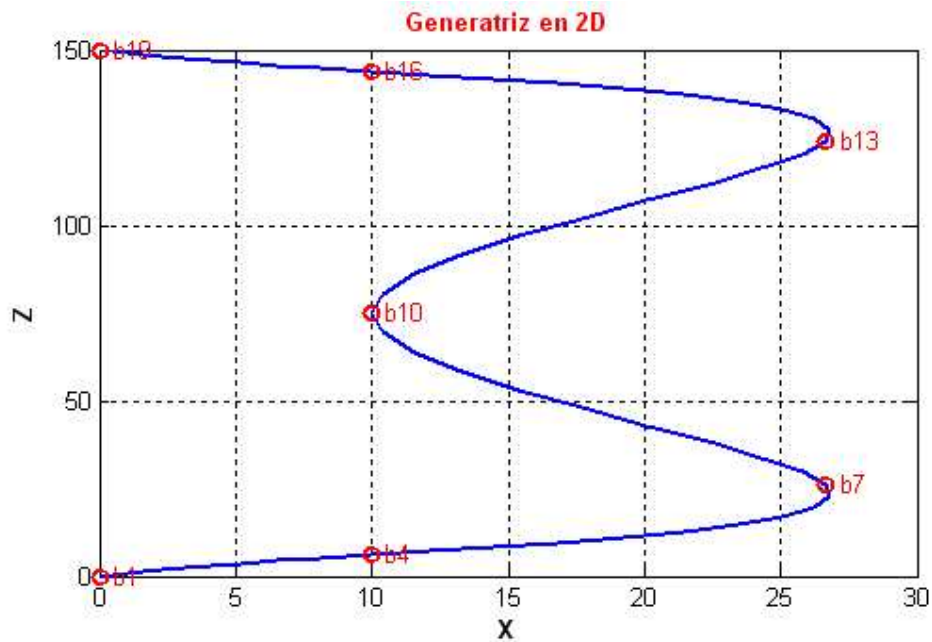


Figura 5. Generatriz obtenida

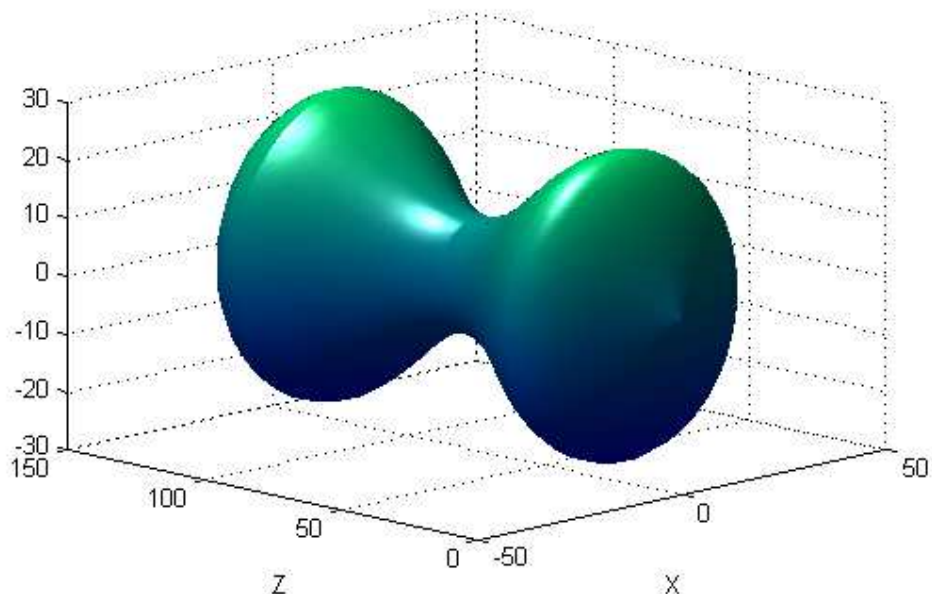


Figura 6. Superficie de revolución obtenida

Ejemplo 3: Se propuso inicialmente una poligonal alineada que une los puntos (1,1) y (10,4).

En este caso se tienen 5 puntos de control y por lo tanto igual número de variables de diseño; las direcciones de búsqueda son las mismas de los ejemplos anteriores. Las cotas de restricción están dadas por ± 5 para todos los puntos.

Los resultados obtenidos son :

- Función objetivo inicial: 327.841
- Función objetivo final: 322.042
- Cantidad de iteraciones: 12
- Variable de diseño obtenidas:
 $x = [-1.4427 \quad -2.5073 \quad -3.2683 \quad -3.3720 \quad -2.7408]$

Ninguna de las restricciones es activa, se encuentra un mínimo de la función objetivo dentro de la región de diseño. Se verifica que esta solución se aproxima a la solución exacta $z = \text{arc}\{\cosh(x)\} + 1$.

Esta ejemplificación corresponde a las figuras 7, 8 y 9.

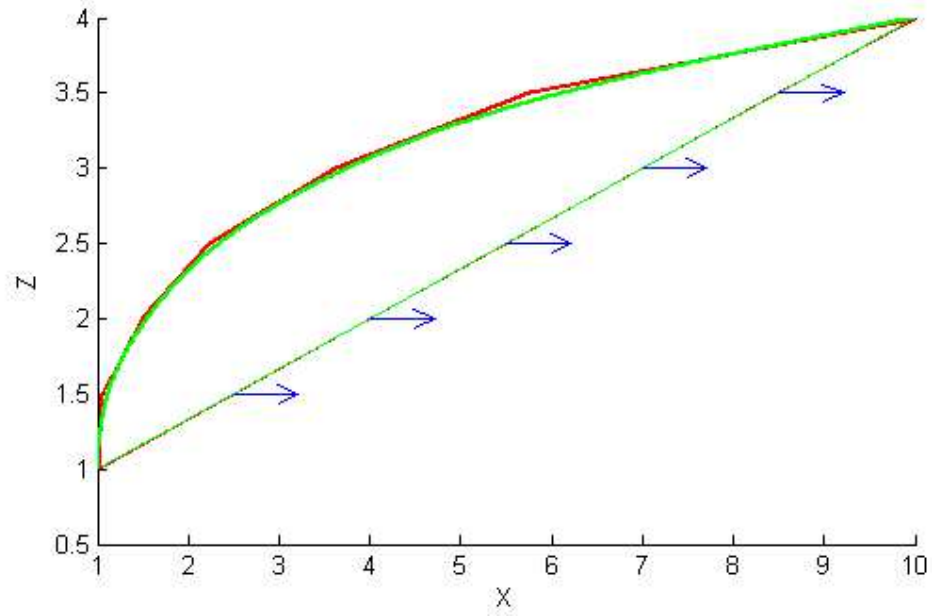


Figura 7. Condiciones iniciales y resultado obtenido

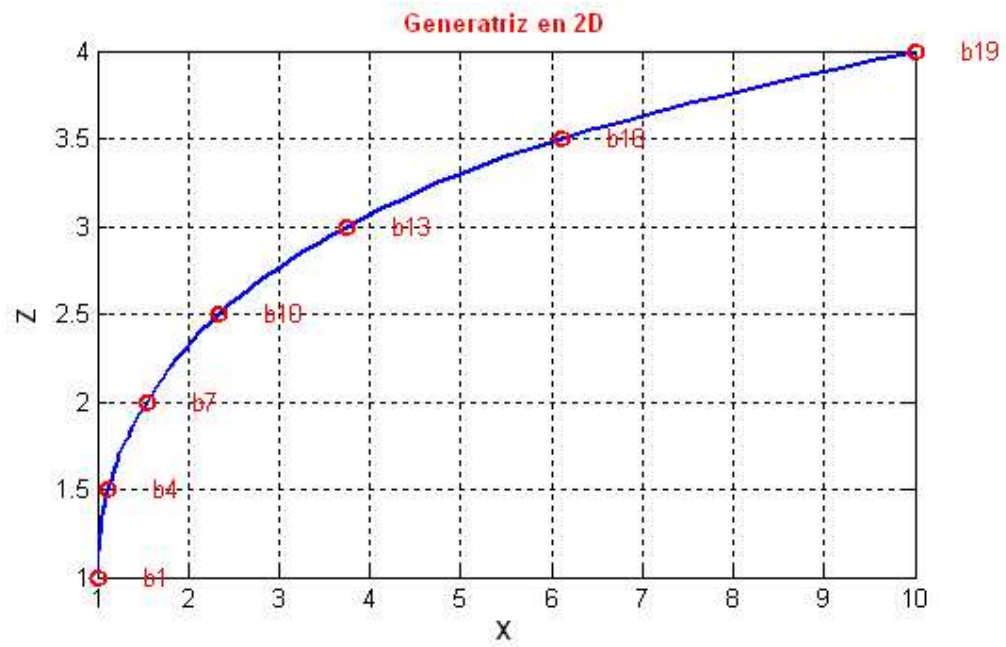


Figura 8. Generatriz obtenida

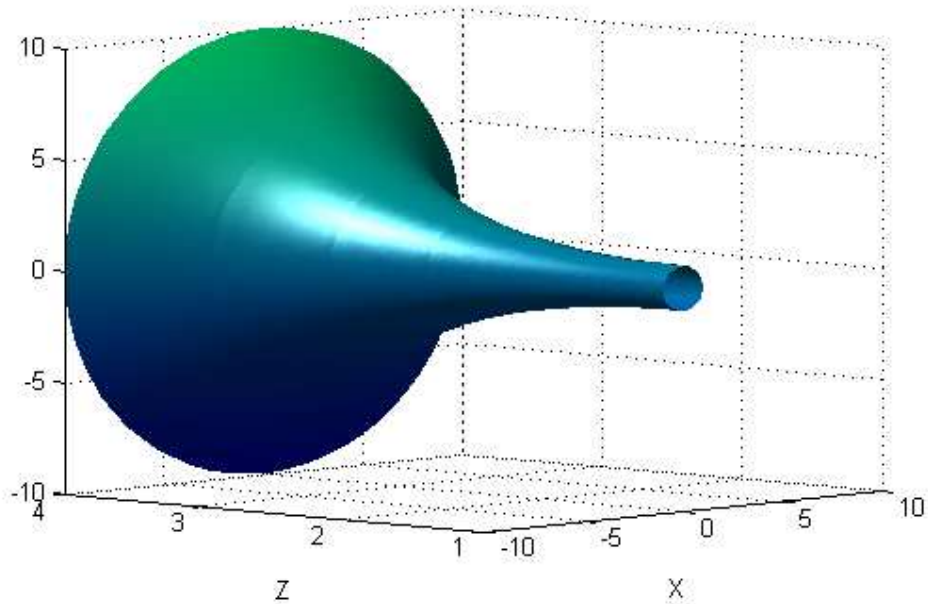


Figura 9. Superficie de revolución obtenida

Ejemplo 4: Al ejemplo 3 solo se le han variado las restricciones por ± 5 ; ± 5 ; ± 0.01 ; ± 5 y ± 5 , obteniéndose los siguientes resultados:

- Función objetivo inicial: 327.841
- Función objetivo final: 325.425
- Cantidad de iteraciones: 24
- Variable de diseño obtenidas:
 $x = [-1.0810 \quad -1.9001 \quad -0.0099 \quad -1.2035 \quad -0.5157]$

Solamente una restricción es activa.

En forma análoga a los demás ejemplos, las figuras 10, 11 y 12 grafican este problema.

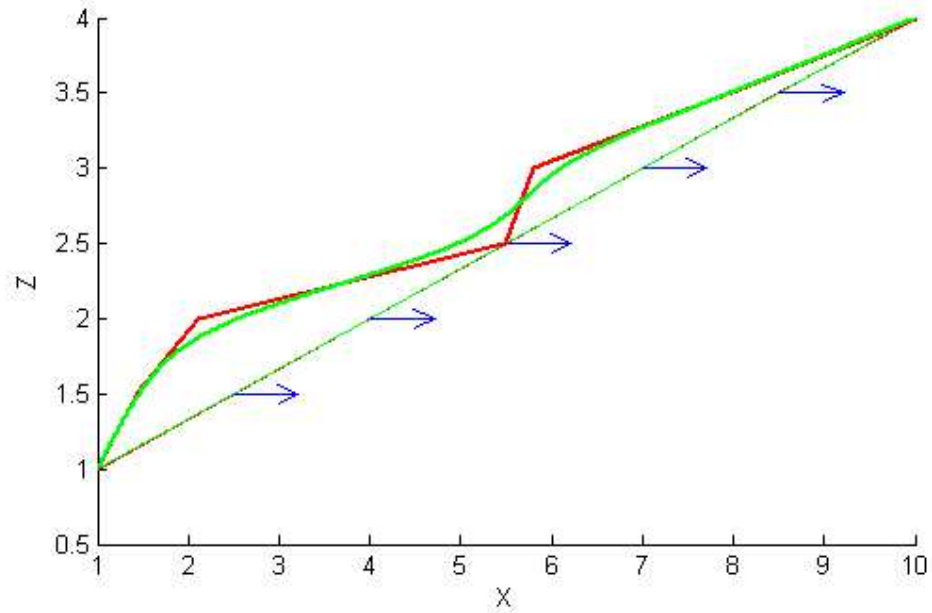


Figura 10. Condiciones iniciales y resultado obtenido

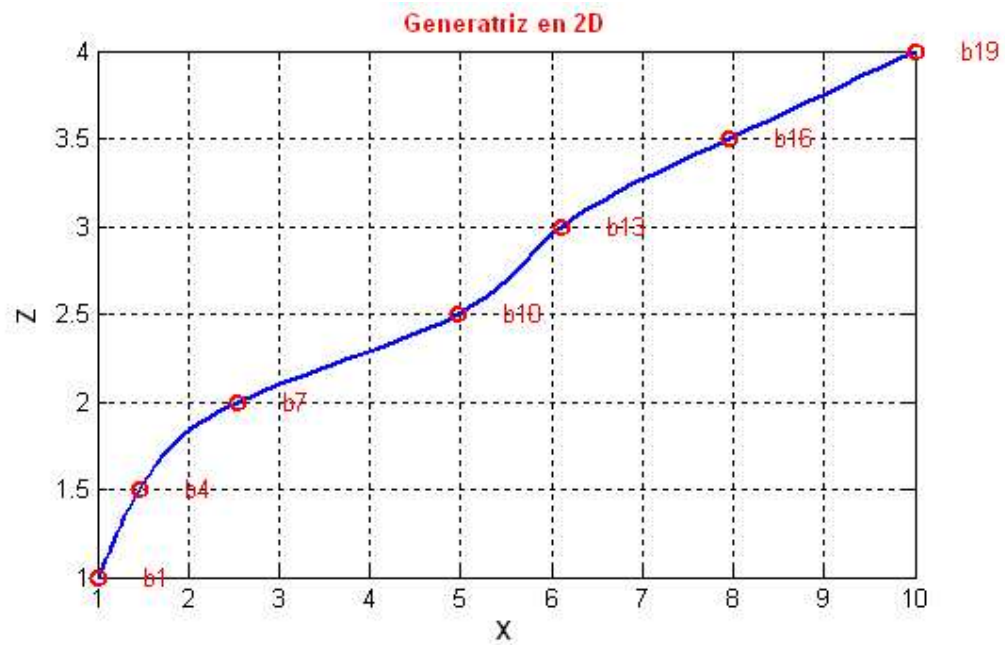


Figura 11. Generatriz obtenida

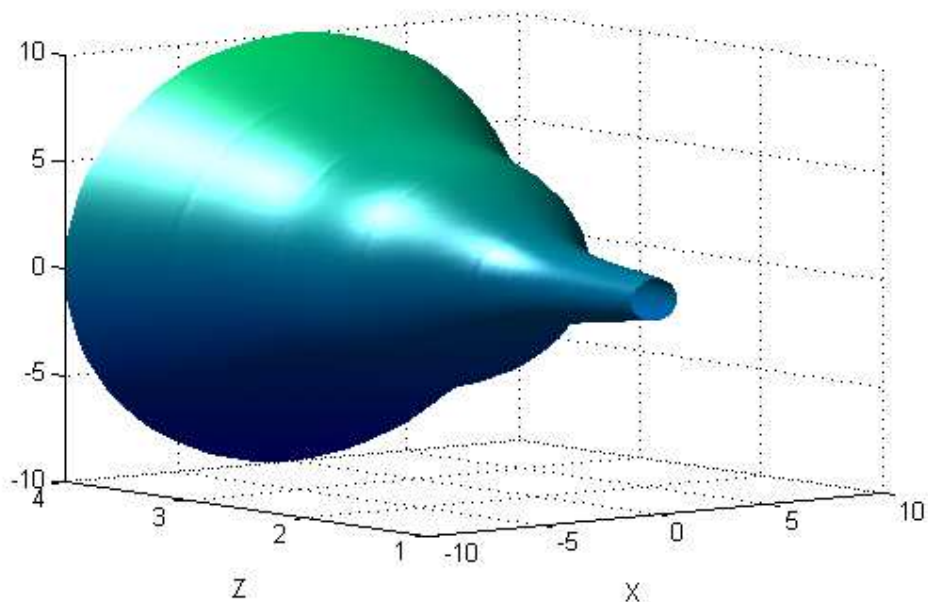


Figura 12. Superficie de revolución obtenida

AGRADECIMIENTOS

Agradecemos toda la colaboración que nos a brindó el Dr. Herskovits para poder realizar este trabajo, como así a su grupo de trabajo.

REFERENCIAS

- ¹ J. Herskovits, P. Mappa, E. Goulart, C. M Mota Soares, “Mathematical programming models and algorithms for engineering design optimization”, *Comput. Methods Appl. Mech. Engrg.* **194** 3244-3268 (2005).
- ² L. Leithold, *El Calculo*, Oxford University Press, 1999.
- ³ S. Nakamura, *Analisis Numerico y Visualizacion Grafica con MatLab*. Prentice-Hall Hispanoamericana, S.A., 1997.
- ⁴ R. Burden y J. Faires, *Numerical Analysis*, Brooks & Cole, 1997.
- ⁵ García de Jalón, J; Rodríguez, J.I.; Brázalez, A., *Aprenda MATLAB como si estuviera en primero*, Universidad Politecnica de Madrid, 2001.