Calculo de Curvas Geodésicas numéricamente en superficie Toroide

Angel Fernando García Núñez

Tensor métrico

$$g_{ij} = (u^k)_i (u^k)_j = \hat{e}_i \cdot \hat{e}_j$$

Símbolos de Chistoffel

Primera Clase

$$\Gamma_{jkl} = \frac{1}{2}\left[(g_{jl})_k - (g_{jk})_l + (g_{lk})_j\right]$$

Segunda Clase

$$\Gamma^i_{jk} = \frac{1}{2} g^{li}\left[(g_{jl})_k - (g_{jk})_l + (g_{lk})_j\right] = \Gamma_{jkl} g^{li}$$

Ecuación Geodésica

$$u^{i\prime\prime} + \Gamma^i_{jk} u^{j\prime} u^{k\prime} = 0$$

Parametrización de un toro

$$\vec{x}(u^1,u^2) = \left[(R+r\cos u^1)\cos u^2, (R+r\cos u^1)\sin u^2, r\sin u^1\right]$$

Vectores tangentes

$$\hat{e}_1 = \frac{\partial \vec{x}}{\partial u^1} = \left[-r\sin u^1\cos u^2, -r\sin u^1\sin u^2, r\cos u^1\right]$$

$$\hat{e}_2 = \frac{\partial \vec{x}}{\partial u^2} = \left[-(R+r\cos u^1)\sin u^2, (R+r\cos u^1)\cos u^2, 0\right]$$

Tensor métrico

$$g_{11} = \hat{e}_1\cdot\hat{e}_1 = r^2\sin^2 u^1\cos^2 u^2 + r^2\sin^2 u^1\sin^2 u^2 + r^2\cos^2 u^1 = r^2$$

$$g_{12} = \hat{e}_1\cdot\hat{e}_2 = (R+r\cos u^1)r\sin u^1\cos u^2\sin u^2 - (R+r\cos u^1)r\sin u^1\sin u^2\cos u^2 = 0$$

$$g_{21} = \hat{e}_2\cdot\hat{e}_1 = \hat{e}_1\cdot\hat{e}_2 = g_{12} = 0$$

$$g_{22} = \hat{e}_2\cdot\hat{e}_2 = (R+r\cos u^1)^2\sin^2 u^2 + (R+r\cos u^1)^2\cos^2 u^2 = (R+r\cos u^1)^2$$

$$g_{ij} = \begin{bmatrix} r^2 & 0 \\ 0 & (R+r\cos u^1)^2 \end{bmatrix}$$

Derivadas parciales del tensor métrico

$$(g_{ij})_1 = \begin{bmatrix} 0 & 0 \\ 0 & -2r(R+r\cos u^1)\sin u^1 \end{bmatrix} \quad , \quad (g_{ij})_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Tensor métrico contra variante

$$g^{ij} = \frac{1}{g}\mathrm{cof}(g_{ij})$$

$$g = |g_{ij}| = r^2(R + r\cos u^1)^2 \quad , \quad \mathrm{cof}(g_{ij}) = \begin{bmatrix} (R + r\cos u^1)^2 & 0 \\ 0 & r^2 \end{bmatrix}$$

$$g^{ij} = \begin{bmatrix} \frac{1}{r^2} & 0 \\ 0 & \frac{1}{(R+r\cos u^1)^2} \end{bmatrix}$$

Símbolos de Chistoffel

$$\Gamma^1_{11} = \frac{1}{2}g^{11}\left[(g_{11})_1 - (g_{11})_1 + (g_{11})_1\right] + \frac{1}{2}g^{21}\left[(g_{12})_1 - (g_{11})_2 + (g_{21})_1\right]$$

$$\Gamma^1_{11} = \frac{1}{2r^2}(0 - 0 + 0) + 0(0 - 0 + 0) = 0$$

$$\Gamma^1_{12} = \Gamma^1_{21} = \frac{1}{2}g^{11}\left[(g_{11})_2 - (g_{12})_1 + (g_{12})_1\right] + \frac{1}{2}g^{21}\left[(g_{12})_2 - (g_{12})_2 + (g_{22})_1\right]$$

$$\Gamma^1_{12} = \Gamma^1_{21} = \frac{1}{2r^2}(0 - 0 + 0) + 0(0 - 0 - 2r(R + r\cos u^1)\sin u^1) = 0$$

$$\Gamma^2_{11} = \frac{1}{2}g^{12}\left[(g_{11})_1 - (g_{11})_1 + (g_{11})_1\right] + \frac{1}{2}g^{22}\left[(g_{12})_1 - (g_{11})_2 + (g_{21})_1\right]$$

$$\Gamma^2_{11} = 0(0 - 0 + 0) + \frac{1}{2(R + r\cos u^1)^2}(0 - 0 + 0) = 0$$

$$\Gamma^2_{12} = \Gamma^2_{21} = \frac{1}{2}g^{12}\left[(g_{11})_2 - (g_{12})_1 + (g_{12})_1\right] + \frac{1}{2}g^{22}\left[(g_{12})_2 - (g_{12})_2 + (g_{22})_1\right]$$

$$\Gamma^2_{12} = \Gamma^2_{21} = 0(0-0+0) + \frac{1}{2(R+r\cos u^1)^2}(0-0-2r(R+r\cos u^1)\sin u^1) = -\frac{r\sin u^1}{(R+r\cos u^1)}$$

$$\Gamma^1_{22} = \frac{1}{2}g^{11}\left[(g_{21})_2 - (g_{22})_1 + (g_{12})_2\right] + \frac{1}{2}g^{21}\left[(g_{22})_2 - (g_{22})_2 + (g_{22})_2\right]$$

$$\Gamma^1_{22} = \frac{1}{2r^2}(0 + 2r(R+r\cos u^1)\sin u^1 + 0) + 0(0+0) = \frac{(R+r\cos u^1)\sin u^1}{r}$$

$$\Gamma^2_{22} = \frac{1}{2}g^{12}\left[(g_{21})_2 - (g_{22})_1 + (g_{12})_2\right] + \frac{1}{2}g^{22}\left[(g_{22})_2 - (g_{22})_2 + (g_{22})_2\right]$$

$$\Gamma^2_{22} = 0(0 + 2r(R+r\cos u^1)\sin u^1 + 0) + \frac{1}{2(R+r\cos u^1)^2}(0-0+0) = 0$$

Ecuaciones Geodésicas

$$u^{1\prime\prime} + \Gamma^1_{22} u^{2\prime} u^{2\prime} = 0 \quad \rightarrow \quad u^{1\prime\prime} + \frac{(R + r\cos u^1)\sin u^1}{r} u^{2\prime} u^{2\prime} = 0$$

$$u^{2\prime\prime} + \Gamma^2_{12} u^{1\prime} u^{2\prime} = 0 \quad \rightarrow \quad u^{2\prime\prime} - \frac{r\sin u^1}{(R + r\cos u^1)} u^{1\prime} u^{2\prime} = 0$$

Solución numérica

Sistema de ecuaciones diferenciales de primer orden

$$y_1' = u_1' = y_2$$

$$y_2' = u_1'' = -\frac{(R + r\cos y^1)\sin y^1}{r} y_4^2$$

$$y_3' = u_2' = y_4$$

$$y_4' = u_2'' = \frac{r\sin y^1}{(R + r\cos y^1)} y_2 y_4$$

$$\vec{y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} \quad , \quad \vec{f}(\vec{y}) = \begin{pmatrix} y_2 \\ -\frac{(R+r\cos y^1)\sin y^1}{r} y_4^2 \\ y_4 \\ \frac{r\sin y^1}{(R+r\cos y^1)} y_2 y_4 \end{pmatrix} \quad : \quad \vec{f}(\vec{y}) = \vec{y}'$$

Método Runge-Kutta de cuarto orden

$$\vec{k}_1 = \vec{f}(\vec{y}_i)$$

$$\vec{k}_2 = \vec{f}\left(\vec{y}_i + \frac{h}{2}\vec{k}_1\right)$$

$$\vec{k}_3 = \vec{f}\left(\vec{y}_i + \frac{h}{2}\vec{k}_2\right)$$

$$\vec{k}_4 = \vec{f}\left(\vec{y}_i + h\vec{k}_3\right)$$

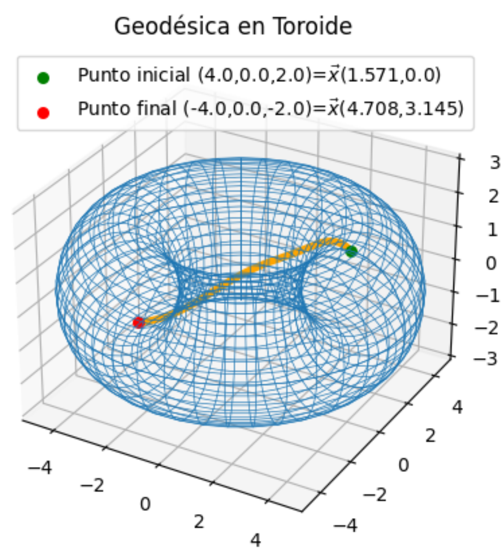
$$\vec{y}_{i+1} = \vec{y}_i + \frac{h}{6}\left(\vec{k}_1 + 2\vec{k}_2 + 2\vec{k}_3 + \vec{k}_4\right)$$

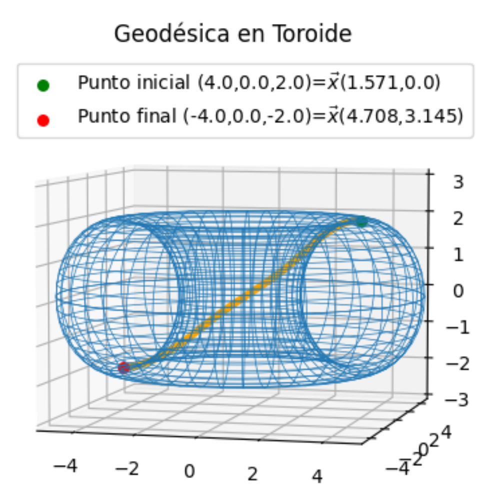
Solución numérica programada en Python, para los puntos:

$$P_1 = \vec{x}\left(\frac{\pi}{2}, 0\right) = (4, 0, 2) \quad , \quad P_2 = \vec{x}\left(\frac{3}{2}\pi, \pi\right) = (-4, 0, -2)$$

Velocidades iniciales obtenidas: $\alpha = 0.4968 \quad , \quad \beta = 0.1926$

Error mínimo obtenido: $\Delta = 0.0058$



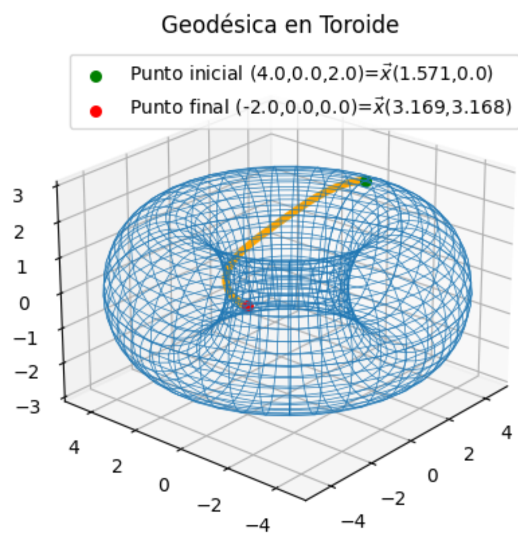


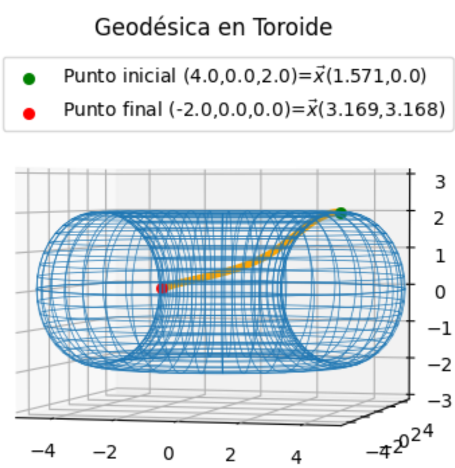
Solución numérica programada en Python, para los puntos:

$$P_1 = \vec{x}\left(\frac{\pi}{2}, 0\right) = (4, 0, 2) \quad , \quad P_2 = \vec{x}\,(\pi, \pi) = (-2, 0, 0)$$

Velocidades iniciales obtenidas: $\alpha = 0.4263 \quad , \quad \beta = 0.18$

Error mínimo obtenido: $\Delta = 0.038$



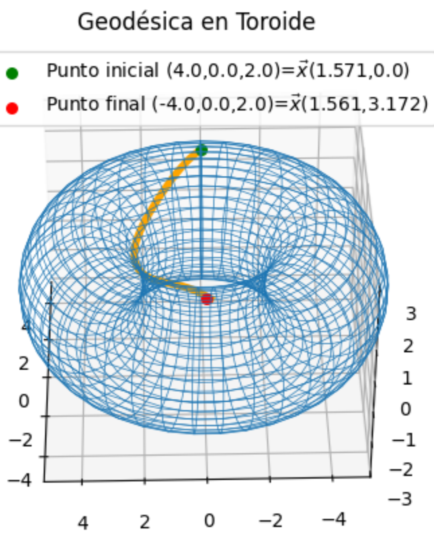


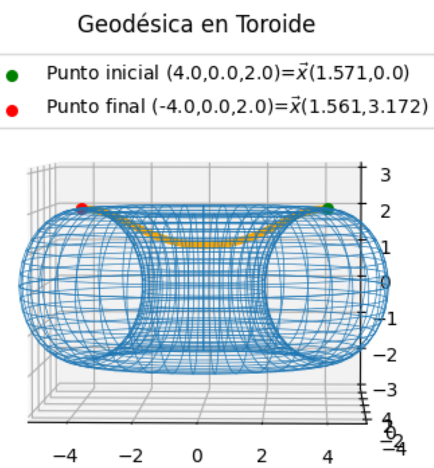
Solución numérica programada en Python, para los puntos:

$$P_1 = \vec{x}\left(\frac{\pi}{2}, 0\right) = (4, 0, 2) \quad , \quad P_2 = \vec{x}\left(\frac{\pi}{2}, \pi\right) = (-4, 0, 2)$$

Velocidades iniciales obtenidas: $\alpha = 0.4526 \quad , \quad \beta = 0.0.2105$

Error mínimo obtenido: $\Delta = 0.0323$



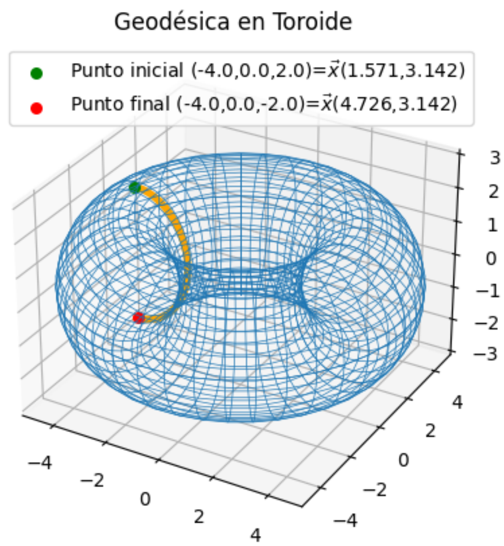


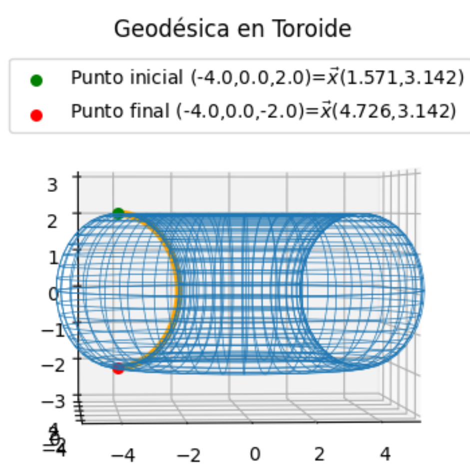
Solución numérica programada en Python, para los puntos:

$$P_1 = \vec{x}\left(\frac{\pi}{2}, \pi\right) = (4, 0, 2) \quad , \quad P_2 = \vec{x}\left(\frac{3}{2}\pi, \pi\right) = (-4, 0, -2)$$

Velocidades iniciales obtenidas: $\alpha = 0.3158 \quad , \quad \beta = 0$

Error mínimo obtenido: $\Delta = 0.0131$



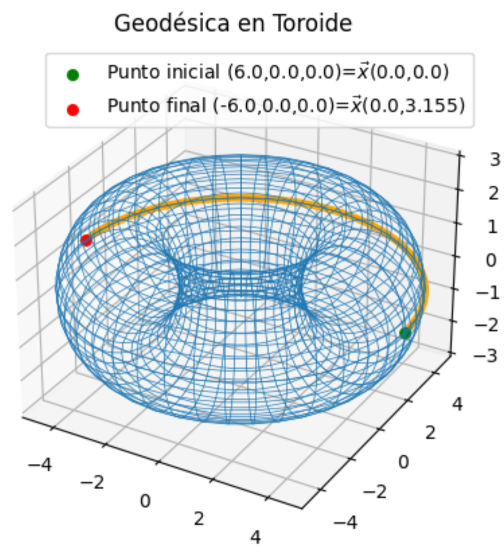


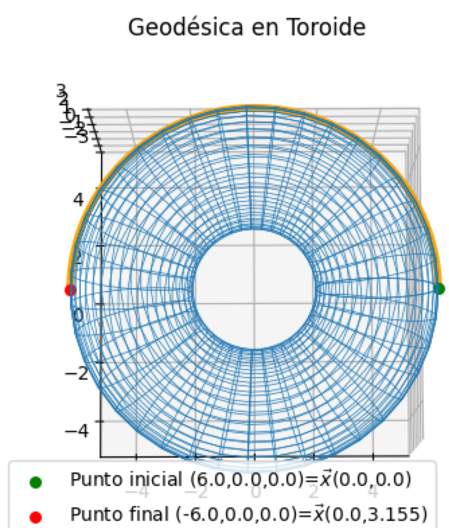
Solución numérica programada en Python, para los puntos:

$$P_1 = \vec{x}\,(0,0) = (6,0,0) \quad , \quad P_2 = \vec{x}\,(0,\pi) = (-6,0,0)$$

Velocidades iniciales obtenidas: $\alpha = 0 \quad , \quad \beta = 0.3158$

Error mínimo obtenido: $\Delta = 0.0131$



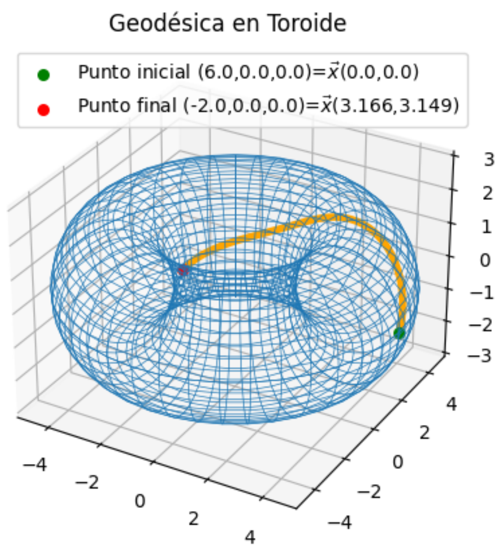


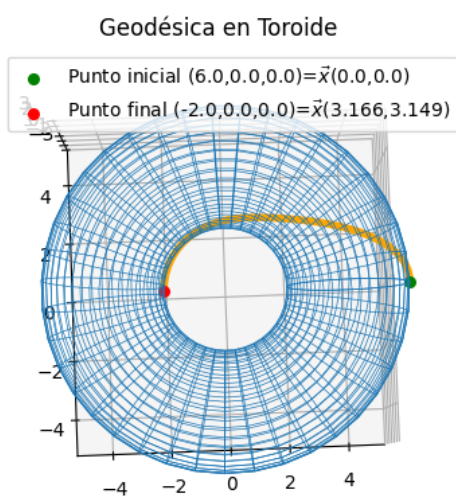
Solución numérica programada en Python, para los puntos:

$$P_1 = \vec{x}\,(0,0) = (6,0,0) \quad , \quad P_2 = \vec{x}\,(\pi,\pi) = (-2,0,0)$$

Velocidades iniciales obtenidas: $\alpha = 0.6368 \quad , \quad \beta = 0.1421$

Error mínimo obtenido: $\Delta = 0.0251$





Solución numérica programada en Python, para los puntos:

$$P_1 = \vec{x}\,(0,0) = (6,0,0) \quad , \quad P_2 = \vec{x}\left(\frac{3.5}{2}\pi, \pi\right) = (-5.414, 0, -1.414)$$

Velocidades iniciales obtenidas: $\alpha = 0.7753 \quad , \quad \beta = 0.1558$

Error mínimo obtenido: $\Delta = 0.0077$