Práctica 8 - Geometría Computacional

Marcos Herrero Agustín

1. Introducción

El objetivo de esta práctica es trasladar de forma paralela dos vectores dados tangentes a la esfera unidad del espacio euclídeo tridimensional. Este tarea nos permitirá entender mejor el concepto de transporte paralelo aplicado a variedades Riemannianas.

2. Datos y condiciones iniciales

Para la realización de la práctica se han utilizado los siguientes datos y condiciones iniciales:

- El espacio de parámetros para las coordenadas esféricas:
  - $[0, 2\pi)$ para las longitudes (ϕ).
  - $[-\frac{\pi}{2}, \frac{\pi}{2}]$ para las latitudes(θ).
- Los paralelos (latitudes) a lo largo de los cuales se han de trasladar los vectores: $\theta_0 = 0$ y $\theta_0 \in (0, \frac{\pi}{2})$ a mi elección.
- El punto de la esfera en el cual es tangente el vector a trasladar: $q = (0, \theta_0)$.
- Las coordenadas del vector a trasladar en el plano tangente a q: $v_0 = (0, \frac{\pi}{5})$.

3. Metodología

Teniendo en cuenta las restricciones sobre el espacio de parámetros (la longitud $\phi \in [0, 2\pi)$ y la latitud $\theta \in [-\frac{\pi}{2}, \frac{\pi}{2}]$) parametrización de la esfera que hemos escogido es:

$$(x, y, z) = (\cos\theta\cos\phi, \cos\theta\sin\phi, \sin\theta)$$

Describimos a continuación el método utilizado para trasladar cada uno de los vectores. El objetivo es obtener las coordenadas cartesianas del vector trasladado un tiempo t para poder representarlo. Inicialmente, el vector se representa en el espacio tridimensional por el bipunto $q + v_0$ donde q tiene coordenadas esféricas $(\phi, \theta) = (0, \theta_0)$ y v_0 tiene coordenadas $(0, \frac{\pi}{5})$ respecto a la base usual del plano tangente. Dado que trasladamos a lo largo del paralelo $\theta = \theta_0$, el vector trasladado vendrá dado por el bipunto $o + v_t$ donde o tendrá coordenadas $(2\pi t, \theta_0)$ y v_t resultará de aplicar una cierta función f_t (mostrada en la sección siguiente) sobre el vector original y la longitud ϕ del transporte completo. Una vez obtenidas las coordenadas de o (esféricas) y v_t (respecto de la base usual del plano tangente a o) se opera para obtener sus coordenadas cartesianas, formato en el que ya pueden sumarse y representarse en el espacio euclídeo.

Nosotros tomamos siempre $\phi = 2\pi$, ya que el parámetro t ya nos permite regular la progresividad y queremos que, para $t = 1$, el vector haya dado una vuelta completa.

4. Resultados y discusión

4.1. Apartado *i)*

Se ha construido una familia paramétrica no lineal que reproduce desde la identidad hasta la traslación paralela del vector v_0. La familia es la siguiente:

$$f_t : [0, 2\pi] \times \left[-\frac{\pi}{2}, \frac{\pi}{2}\right] \to S_1^2$$

$$\begin{pmatrix} \phi \\ \theta_0 \end{pmatrix} \mapsto \frac{\pi}{5} \begin{pmatrix} \frac{\sin(\sin(\theta_0)\phi t^2)}{\cos\theta_0} \\ \cos(\sin(\theta_0)\phi t^2) \end{pmatrix}$$

para $t \in \mathbb{R}$, aunque se utilizará en el intervalo $[0, 2]$. El hecho de utilizar una familia paramétrica que depende de t^2 en lugar de t hará que el movimiento, en lugar de uniforme, sea uniformemente acelerado.

4.2. Apartado *ii)*

Se ha realizado una animación de 40 fotogramas que muestra el transporte paralelo progresivo de dos copias del vector $v_0 = (0, \frac{\pi}{5})$. Una de ellas (azul) es, inicialmente, tangente a la esfera en el punto de coordenadas esféricas $(\phi, \theta) = (0, 0)$ y la otra (roja) en $(\phi, \theta) = (0, \frac{\pi}{6})$. Se utiliza para ello la familia paramétrica anterior. El hecho de que $\sin(\frac{\pi}{6}) = 0.5$ hace que el segundo vector requiera un tiempo $t = 2$ para volver al punto de origen en la misma posición en que comenzó, mientras que para el primer vector esto se logra en un tiempo $t = 1$.

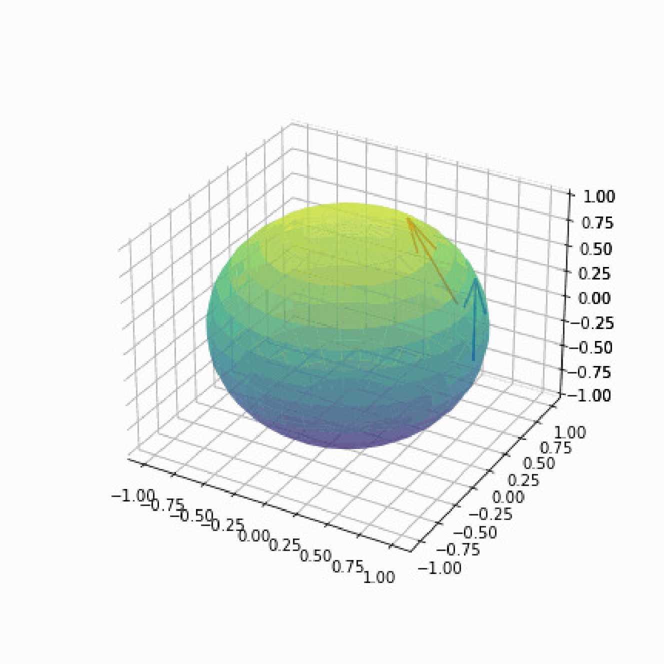

Figura 1: $t = 0$

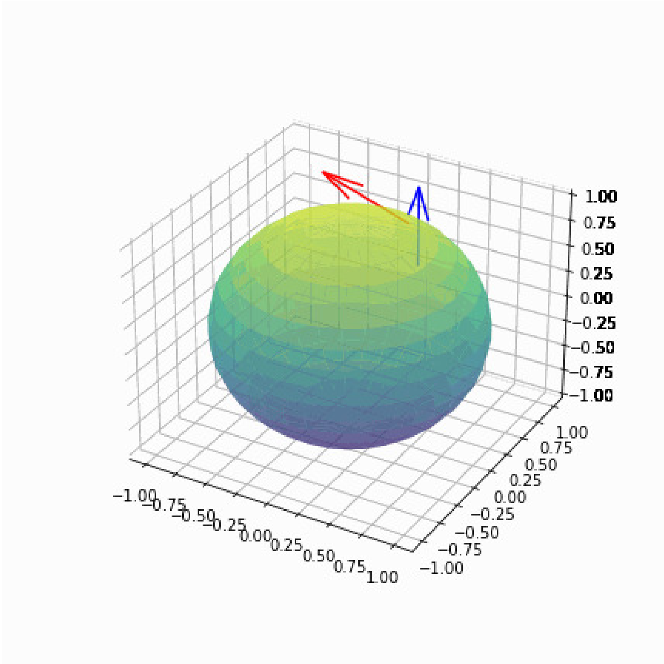

Figura 2: $t = 0.5$

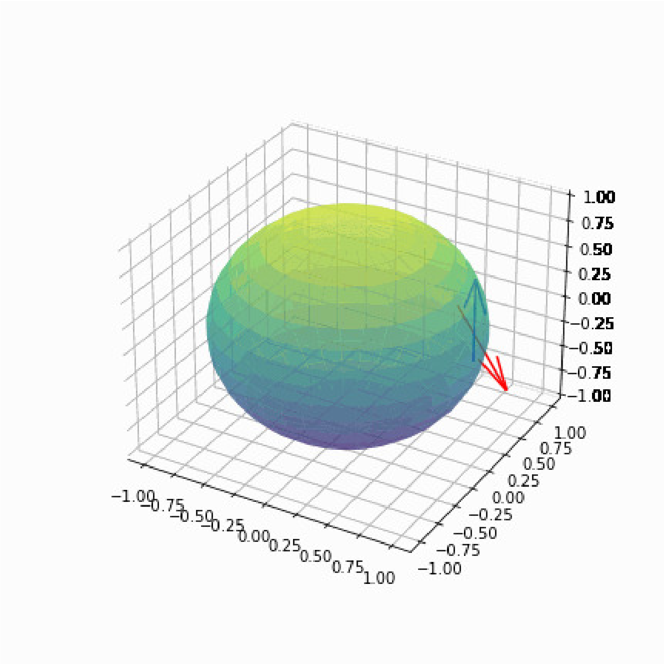

Figura 3: $t = 1$

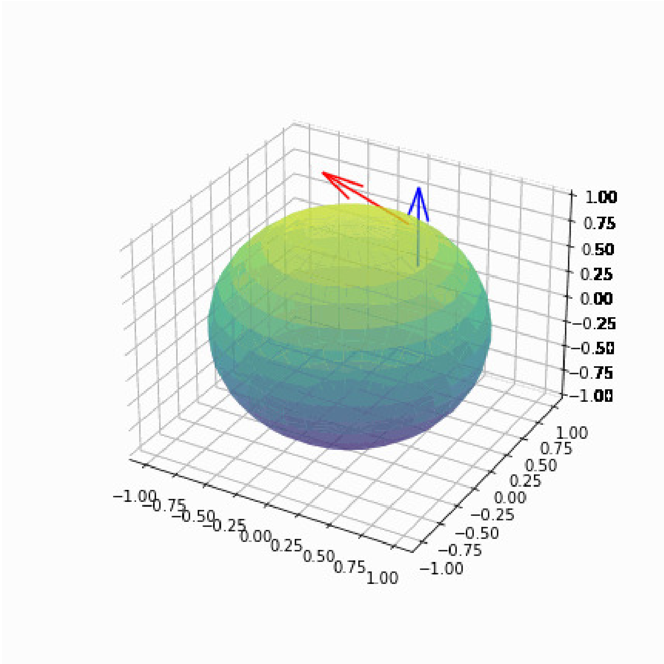

Figura 4: $t = 1.5$

La animación se encuentra en el archivo "animacion.gif" que acompaña a esta memoria. Las figuras 1,2,3 y 4 representan algunos fotogramas de la misma.

Puede observarse que el desplazamiento es muy lento en los primeros instantes, pero se va acelerando y alcanza una velocidad muy alta al final de la animación. Esto se debe a haber utilizado un término t^2 en la familia paramétrica.

5. Conclusiones

Esta práctica nos ha ayudado a entender el funcionamiento del transporte paralelo en variedades Riemannianas del espacio euclídeo, como es el caso de la esfera. Si bien la traslación es la esperable en el caso del vector azul, el vector rojo da bastantes vueltas, lo cual nos muestra que incluso en variedades sencillas el transporte paralelo tiene propiedades interesantes que conviene estudiar.

Apéndice A   Código utilizado

```
"""
Práctica 8 de Geometría Computacional
Autor: Marcos Herrero
"""

import numpy as np
import matplotlib.pyplot as plt
from matplotlib import animation

#Transforma de coordenadas esféricas a cartesianas
def esfToCart(phi,theta):
    x = np.cos(theta)*np.cos(phi)
    y = np.cos(theta)*np.sin(phi)
    z = np.sin(theta)

    return np.array([x,y,z])

#Obtiene las coordenadas cartesianas del punto del plano tangente de o determinado
# por el vector v tangente a la esfera en o
def puntoPlano(o,o_phi,o_theta,v_phi,v_theta):
    b1 = np.array([-np.cos(o_theta)*np.sin(o_phi),np.cos(o_theta)*np.cos(o_phi),0])
    b2 = np.array([-np.sin(o_theta)*np.cos(o_phi),-np.sin(o_theta)*np.sin(o_phi),np.cos

    return o + v_phi * b1 + v_theta * b2

#Familia paramétrica pedida en el apartado i)
def famParam(t,phi,theta0,v02):
    p_phi = v02/np.cos(theta0) * np.sin(np.sin(theta0)*phi*t**2)
    p_theta = v02* np.cos(np.sin(theta0)*phi*t**2)
    return p_phi,p_theta


#Función que realiza la animación
def fAnimation(t,theta01,theta02,v02, xesf,yesf,zesf):
    ax = plt.axes(projection = '3d')

    o_phi1 = 2*np.pi*t**2
    o_theta1 = theta01
    v_phi1,v_theta1 = famParam(t,2*np.pi,theta01,c0)

    o1 = esfToCart(o_phi1,o_theta1)
    p1 = puntoPlano(o1, o_phi1,o_theta1,v_phi1,v_theta1)
    X1,Y1,Z1,U1,V1,W1 = np.concatenate((o1,p1-o1))

    o_phi2 = 2*np.pi*t**2
    o_theta2 = theta02
    v_phi2,v_theta2 = famParam(t,2*np.pi,theta02,c0)

    o2 = esfToCart(o_phi2,o_theta2)
    p2 = puntoPlano(o2, o_phi2,o_theta2,v_phi2,v_theta2)
    X2,Y2,Z2,U2,V2,W2 = np.concatenate((o2,p2-o2))

    ax.plot_surface(xesf, yesf, zesf, cmap='viridis', edgecolor='none',alpha = 0.55)
    ax.quiver(X1,Y1,Z1,U1,V1,W1,colors="blue", zorder=3, arrow_length_ratio=0.4)
    ax.quiver(X2,Y2,Z2,U2,V2,W2,colors="red", zorder=3, arrow_length_ratio=0.4)


v02 = np.pi/5
```

```python
#Sistema de referencia
phi = np.linspace(0, 2*np.pi, 30)
theta = np.linspace(-np.pi/2, np.pi/2, 20)
r = 1

x =r* np.outer(np.cos(theta), np.cos(phi))
y = r* np.outer(np.cos(theta), np.sin(phi))
z = r* np.outer(np.sin(theta), np.ones_like(phi))

c0 = np.sqrt(np.pi/5)




'''
Apartado ii): Realizar una animación de la transformación anterior de forma que dos
copias de v0 se trasladen en 2 paralelos diferentes
'''

theta01 = 0
theta02 = np.pi/6

fig = plt.figure(figsize=(6,6))
ani = animation.FuncAnimation(fig, fAnimation,frames=np.arange(0,2.01,0.05),
                              fargs = (theta01,theta02,v02,x,y,z), interval=20)
ani.save('animacion.gif',fps=5)
plt.show()
```