UNIVERSIDAD NACIONAL DEL ALTIPLANO

FACULTAD DE INGENIERÍA CIVIL Y ARQUITECTURA

ESCUELA PROFESIONAL DE CIENCIAS FÍSICO MATEMÁTICAS

INFORME DE ANIMACIÓN ENTRE SUPERFICIES

GEOMETRÍA DIFERENCIAL II

Presentado por:

217237 - HERRERA CHIPANA JHEREMY GIOVANY

jherrerac@est.unap.edu.pe

Docente:

Dr. CARLOS RONAL MAMANI MAMANI

PUNO - PERÚ

11 de diciembre de 2024

Índice

1 INTRODUCCIÓN

En este trabajo se presenta una explicación detallada sobre la transición de una **catenaria** a una **helicoide**. Esta transición se modela a través de un parámetro de interpolación que cambia suavemente entre las dos formas. A través de este proceso, se mostrará cómo una superficie inicialmente plana pasa a ser una catenaria, y luego se transforma en una helicoide.

2 CATENARIA

Definition 2.1 — Catenaria. La ecuación paramétrica de la catenaria es una curva hiperbólica que describe el comportamiento de una cuerda suspendida bajo su propio peso. Su ecuación es la siguiente:

$$\mathbf{r}_{\text{catenaria}}(u, v) = (a \cdot \cosh(v) \cdot \cos(u),\ a \cdot \cosh(v) \cdot \sin(u),\ a \cdot v) \tag{1}$$

Donde:
- $u \in < 0, 2\pi >$ es el parámetro angular.
- $v \in \mathbb{R}$ es el parámetro que controla la altura vertical de la catenaria.
- a es un parámetro que define la amplitud de la catenaria.

3 HELICOIDE

Definition 3.1 La ecuación paramétrica de la helicoide describe una superficie generada por un movimiento helicoidal, en el cual un punto se mueve en espiral mientras se traslada a lo largo de un eje. La ecuación de la helicoide es:

$$\mathbf{r}_{\text{helicoide}}(u, v) = (a \cdot \sinh(v) \cdot \cos(u),\ a \cdot \sinh(v) \cdot \sin(u),\ a \cdot u) \tag{2}$$

Donde:
- $u \in < 0, 2\pi >$ es el parámetro angular.
- $v \in \mathbb{R}$ es el parámetro vertical que controla la altura de la helicoide.
- a es un parámetro que controla la amplitud de la helicoide.

4 INTERPOLACIÓN PARA LA TRANSICIÓN

Para realizar la transición de la catenaria a la helicoide, utilizamos un parámetro de interpolación α, que varía entre 0 y 2. La interpolación se divide en dos fases:

1. Para $\alpha \in [0, 1]$, la transición ocurre del plano a la catenaria.
2. Para $\alpha \in [1, 2]$, la transición ocurre de la catenaria a la helicoide.

De esta forma, el parámetro α controla la interpolación entre estas tres formas geométricas: plano, catenaria y helicoide.

4.1 De Plano a Catenaria

Para $\alpha \in [0, 1]$, interpolamos entre el plano $z = 0$ y la catenaria. La ecuación para la interpolación en esta fase es:

$$\mathbf{r}(u, v, \alpha) = (a \cdot ((1 - \alpha) + \alpha \cdot \cosh(v)) \cdot \cos(u), a \cdot ((1 - \alpha) + \alpha \cdot \cosh(v)) \cdot \sin(u), a \cdot \alpha \cdot v) \tag{3}$$

Cuando $\alpha = 0$, esta ecuación describe el plano $z = 0$, y cuando $\alpha = 1$, se obtiene la catenaria.

4.2 De Catenaria a Helicoide

Para $\alpha \in [1, 2]$, interpolamos entre la catenaria y la helicoide. La ecuación de la interpolación en esta fase es:

$$\mathbf{r}(u, v, \alpha) = (a \cdot ((1-\alpha) \cdot \cosh(v) + \alpha \cdot \sinh(v)) \cdot \cos(u),\ a \cdot ((1-\alpha) \cdot \cosh(v) + \alpha \cdot \sinh(v)) \cdot \sin(u),\ a \cdot (\tag{4}$$

En este caso, cuando $\alpha = 1$, obtenemos la catenaria, y cuando $\alpha = 2$, obtenemos la helicoide.

5 INTERPOLACIÓN

La interpolación entre la catenaria y la helicoide, a través de un parámetro α, permite una transición suave entre estas dos superficies. El parámetro α varía de 0 a 2, y a medida que cambia, la forma de la superficie evoluciona de un plano a una catenaria, y finalmente a una helicoide. Esto proporciona una forma continua y suave de modelar la transición entre estos dos tipos de superficies.

A continuación se presenta el código que muestra como se ejecuta la transición

```python
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.animation import FuncAnimation
a = 1
u = np.linspace(0, 2 * np.pi, 100)
v = np.linspace(-2, 2, 100)
fig = plt.figure(figsize=(12, 10))
ax = fig.add_subplot(111, projection='3d')
def get_coordinates(u, v, alpha):
    U, V = np.meshgrid(u, v)
    if alpha <= 1:
        alpha1 = alpha
        X = a * ((1 - alpha1) + alpha1 * np.cosh(V)) * np.cos(U)
        Y = a * ((1 - alpha1) + alpha1 * np.cosh(V)) * np.sin(U)
        Z = a * alpha1 * V
    else:
        alpha2 = alpha - 1
        X = a * ((1 - alpha2) * np.cosh(V) + alpha2 * np.sinh(V)) * np.cos(U)
        Y = a * ((1 - alpha2) * np.cosh(V) + alpha2 * np.sinh(V)) * np.sin(U)
        Z = a * (alpha2 * U + (1 - alpha2) * V)
    return X, Y, Z
def update(frame):
    ax.clear()
    alpha = frame / 200.0 * 2
    X, Y, Z = get_coordinates(u, v, alpha)
    surf = ax.plot_surface(X, Y, Z, cmap='viridis', edgecolor='none')
    ax.set_title(f"Transici n - Frame {frame}")
    ax.set_xlabel('X')
    ax.set_ylabel('Y')
    ax.set_zlabel('Z')
    ax.set_xlim([-3, 3])
    ax.set_ylim([-3, 3])
    ax.set_zlim([-3, 3])
ani = FuncAnimation(fig, update, frames=200, interval=20)
plt.show()
```

Listing 1: Código de la trancisón en Python

6 VIDEO DE ANIMACIÓN

Figura 1: Video de Animación