

# Círculos en círculos

Maximiliano Barajas Sánchez

16 de septiembre de 2023

## 1. Introducción

A lo largo de esta práctica se analizarán dos tipos de trayectorias de gran relevancia geométrica, se trata de las hipocicloides y epicicloides. Comenzaremos introduciendo las definiciones de ambos objetos para mas adelante compartir observaciones sobre simulaciones realizadas a través de software y una implementación sencilla realizada en Python.

## 2. Metodología

En cuanto a los datos a utilizar en esta practica en particular nos centramos en epicicloides e hipocicloides en su forma paramétrica con una serie de relaciones establecidas entre los parámetros de las curvas para experimentar, se utilizó una aplicación de espiró-grafo en línea como Python para como herramientas principales de esta práctica yo decidí utilizar las librerías de Numpy y Plotly, a continuación analizaremos el fundamento teórico de la práctica.

### 2.1. Fundamento Teórico

Comenzaremos definiendo las hipocicloides, este tipo de curvas consisten en la trayectoria generada por un punto de una circunferencia que rueda dentro de otra circunferencia de radio mas grande, a lo largo de este apartado nos referiremos al radio mas grande como  $a$  y al pequeño como  $b$ , consideremos la siguiente imagen:

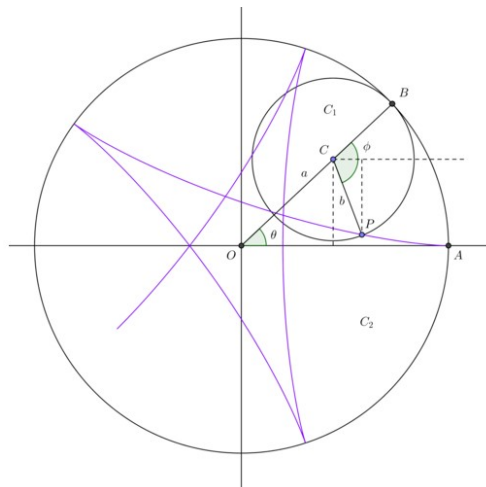


Figura 1: Hipocicloide auxiliar para la obtención de las ecuaciones paramétricas de las hipocicloides

De la figura anterior podemos extraer en primera instancia las siguientes ecuaciones para cada coordenada mediante proyecciones sobre el eje  $x$  y el eje  $y$  respectivamente:

$$x = (a - b)\cos(\theta) + b\cos(\phi - \theta)$$

$$y = (a - b)\sin(\theta) - b\sin(\phi - \theta)$$

Los primeros términos,  $(a - b)\cos(\theta)$  y  $(a - b)\sin(\theta)$  vienen de proyectar la distancia del origen al punto  $C$  o bien el centro de la circunferencia menor la cual esta sujeta al movimiento sobre el eje  $x$  y el eje  $y$  respectivamente, homologamente si proyectamos el segmento del centro del círculo de radio menor al punto  $P(x, y)$  obtenemos los términos  $b\cos(\phi - \theta)$  y  $b\sin(\phi - \theta)$  para  $x$  y  $y$  respectivamente, se realiza la diferencia de los ángulos  $\phi$  y  $\theta$  dado que es el ángulo perteneciente al triángulo rectángulo formado por  $b$  como su hipotenusa y los segmentos punteados como el resto de sus catetos. Para encontrar el ángulo  $\phi$  basta con darnos cuenta que el arco entre los puntos  $P$  y  $B$  es de misma longitud que el arco definido entre  $A$  y  $B$  de lo cual podemos concluir la siguiente relación:

$$b\phi = a\theta$$

$$\phi = \frac{a\theta}{b}$$

De lo anterior se sigue lo siguiente:

$$\phi - \theta = \frac{a\theta}{b} - \theta = \theta\left(\frac{a}{b} - 1\right) = \theta\frac{a - b}{b}$$

Finalmente obtenemos las siguientes expresiones para las coordenadas  $x$  y  $y$ :

$$x = (a - b)\cos(\theta) + b\cos\left(\theta\frac{a - b}{b}\right)$$

$$y = (a - b)\sin(\theta) - b\sin\left(\theta\frac{a - b}{b}\right)$$

Tal como establecen George F. Simmons y ProofWiki

## 2.2. Experimentación inicial con el espirografo

Inicialmente probé con el espirografo en línea el cual es accesible a través del siguiente link obteniendo el siguiente patrón:

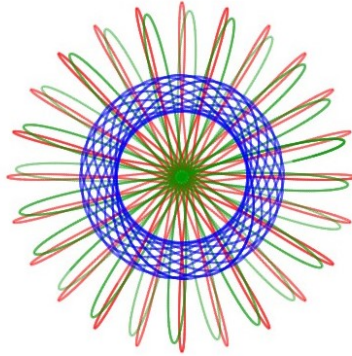


Figura 2: Ejemplo de patrón realizado con el espirografo virtual

### 2.3. Implementación en Python

Personalmente decidí modularizar el código con ayuda de una función la cual fuese la encargada de realizar el proceso de generar la gráfica de la hipocicloide y mostrarla a partir de únicamente los parámetros, decidí hacer uso de la biblioteca Plotly dado que permite generar gráficas interactivas las cuales son a mi parecer también mas atractivas que las realizadas por matplotlib, a continuación la implementación de mi función la cual solo recibe 2 parámetros obligatorios los cuales son  $a$  y  $b$  que son los parámetros de las ecuaciones paramétricas de las hipocicloides.

```
1 import numpy as np
2 import plotly.graph_objects as go
3 from plotly.subplots import make_subplots
4 from random import choice
5 def mostrar_curva(a: np.float64, b: np.float64, t: np.float64 = np.linspace(0, 200 * np.pi,
6     100000 + 1)) -> None:
7     #Ecuaciones Parametricas de la hipocicloide
8     x = (a - b) * np.cos(t) + b * np.cos(t * (a - b) / b)
9     y = (a - b) * np.sin(t) - b * np.sin(t * (a - b) / b)
10    #Inicializamos un objeto subplots para poder construir la grafica
11    fig = make_subplots(rows=1, cols=1)
12    #Creamos nuestra grafica, el color lo decide al azar dentro de las opciones en la lista
13    #dentro del argumento del constructor
14    hipocicloide = go.Scatter(x=x, y=y, mode='lines', line=dict(color=choice(['red', 'blue', '
15        green', 'yellow', 'black', 'magenta', 'cyan'])))
16    #Agregamos la grafica al objeto figura como una trayectoria
17    fig.add_trace(hipocicloide)
18    fig.update_layout(
19        #Le damos la misma escala para ambos ejes X y Y a la grafica
20        xaxis=dict(scaleanchor="y", scaleratio=1),
21        yaxis=dict(constrain="domain"),
22        showlegend=False
23    )
24    #Mostramos la grafica
25    fig.show()
```

Listing 1: Implementación de la función para mostrar hipocicloides dados solamente  $a$  y  $b$

Una vez ya poseemos la implementación podemos proceder con los resultados requeridos para la práctica.

### 3. Discusión

En este apartado discutiremos los resultados obtenidos según las razones entre los parámetros  $a$  y  $b$  las cuales se nos pidieron en la práctica a realizar las hipocicloides asociadas a las mismas, inicialmente elegí  $a = 3$  y  $b = 1$  y obtuve la siguiente gráfica:

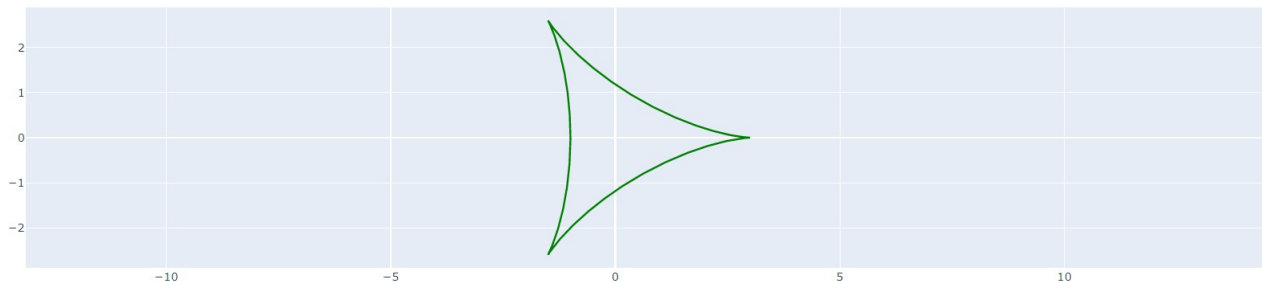


Figura 3: Hipocicloide con parámetros  $a = 3$  y  $b = 1$

Notemos que elegí en mi caso los parámetros de esta manera para que así se cumpliera la relación de cociente establecida en la práctica, después realicé lo mismo con los parámetros  $a = 4$  y  $b = 1$

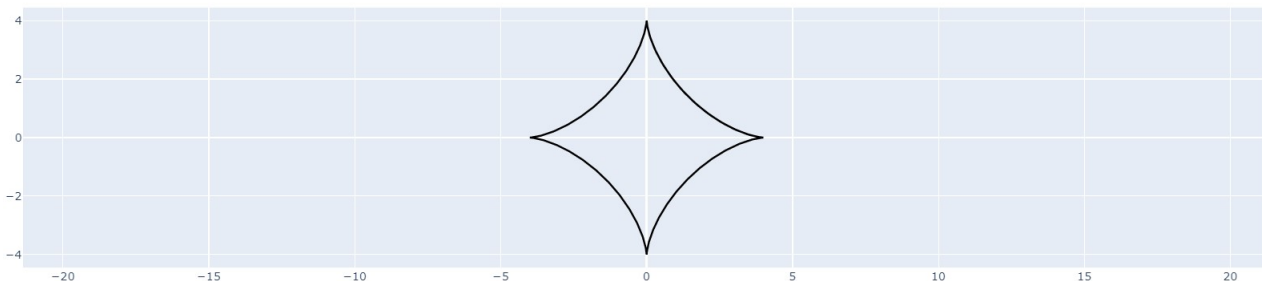


Figura 4: Hipocicloide con parámetros  $a = 4$  y  $b = 1$

Notemos en este caso que se incremento el número de puntos de el patrón similar a una "estrella" formado por la trayectoria asociada a estos parámetros, sucede lo mismo si incrementamos una vez mas el parámetro  $a$  en uno, que consiste en la siguiente gráfica requerida con parámetros  $a = 5$  y  $b = 1$

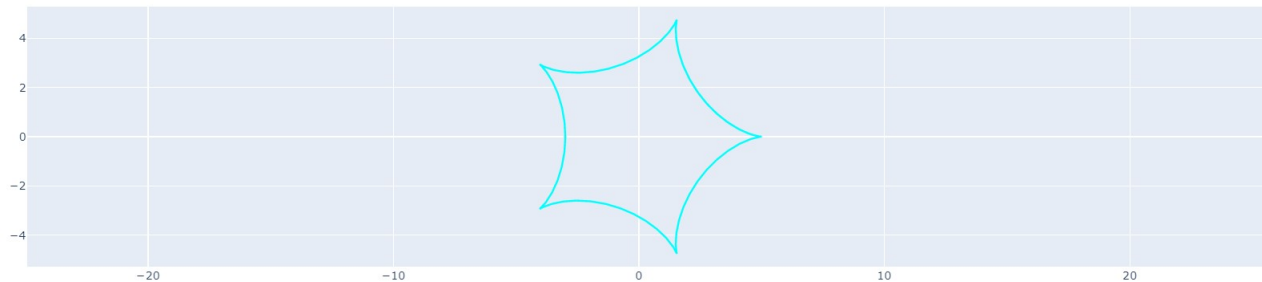


Figura 5: Hipocicloide con parámetros  $a = 5$  y  $b = 1$

Ahora procedemos a incrementar ambos parámetros para obtener la relación de cociente entre ellos requerida por la práctica, a continuación la hipocicloide con parámetros  $a = 5$  y  $b = 3$

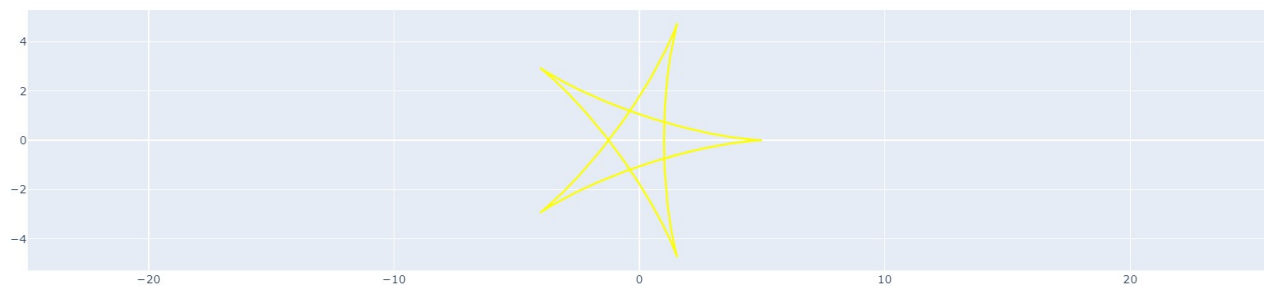


Figura 6: Hipocicloide con parámetros  $a = 5$  y  $b = 3$

Procedemos ahora con la gráfica asociada a los parámetros  $a = 7$  y  $b = 3$ :

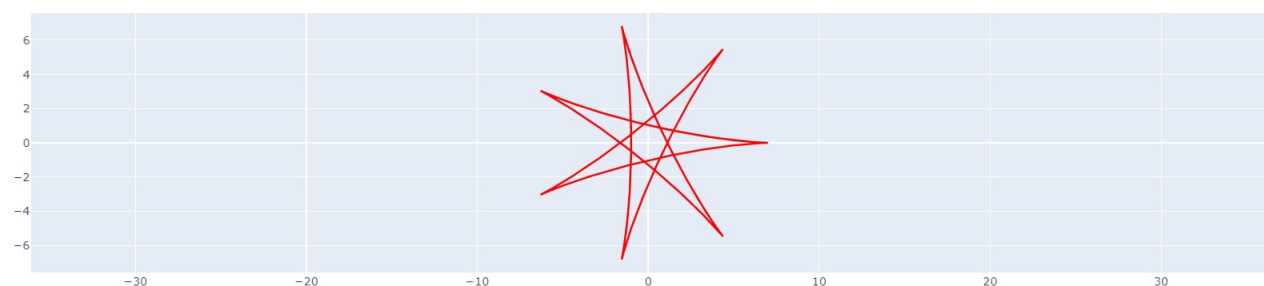


Figura 7: Hipocicloide con parámetros  $a = 7$  y  $b = 3$

Una vez mas incrementaremos el parámetro  $a$  de tal manera de que la siguiente relación requerida en la práctica se cumpla, consideremos los parámetros  $a = 8$  y  $b = 3$ , la hipocicloide asociada a estos es la siguiente:

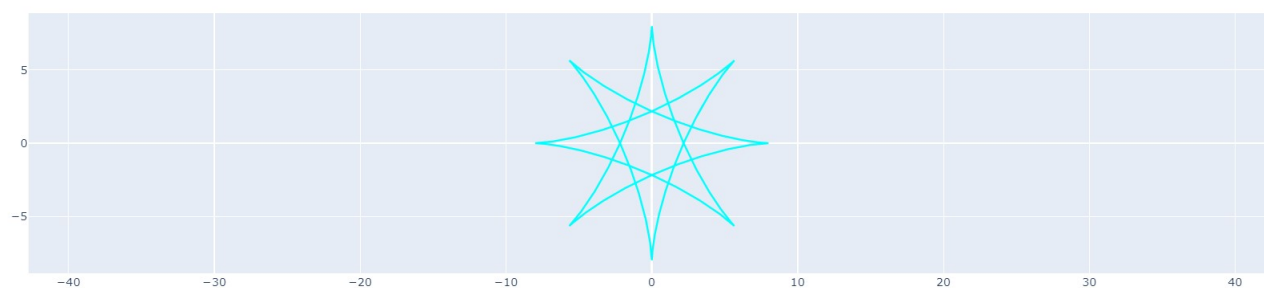


Figura 8: Hipocicloide con parámetros  $a = 8$  y  $b = 3$

Para los siguientes 2 casos donde se trata de una relación establecida por un irracional decidí utilizar las aproximaciones de numpy, inicialmente para la relación de cociente que es igual a  $\pi$  bastó con tomar el parámetro  $a = \frac{1}{\pi}$  y  $b = 1$  obteniendo la siguiente gráfica:

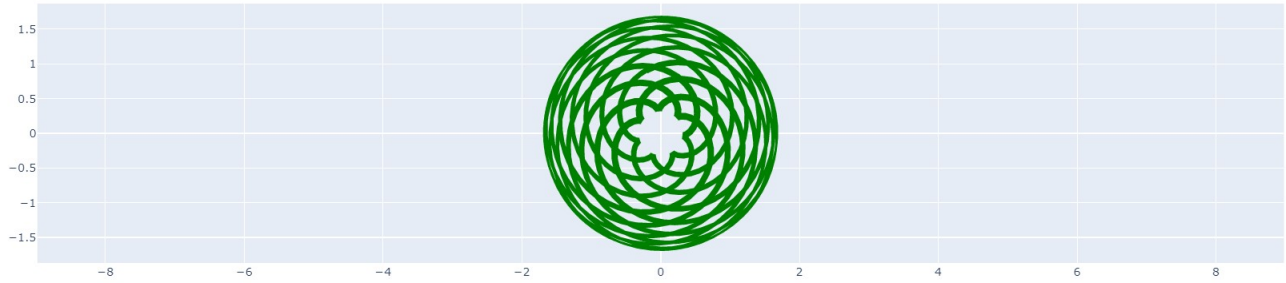


Figura 9: Hipocicloide con parámetros  $a = \frac{1}{\pi}$  y  $b = 1$

Finalmente, de manera similar al inciso anterior decidí tomar el parámetro  $a$  como  $a = \frac{1}{\sqrt{2}}$  y por otra parte  $b = 1$  resultando en la siguiente gráfica:

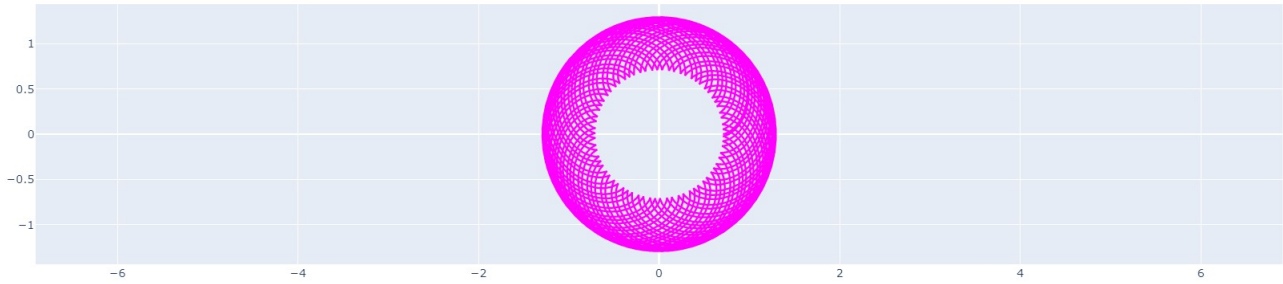


Figura 10: Hipocicloide con parámetros  $a = \frac{1}{\sqrt{2}}$  y  $b = 1$

## 4. Conclusiones

Finalmente podemos concluir que las hipocicloides son un objeto propio de la geometría con un estudio sobre sus características cuanto menos interesante, la peculiaridad de su aspecto me parece atractiva visualmente, sin embargo de igual manera encontré que son de gran utilidad en el diseño de sistemas rotatorios o de engranajes junto con la parametrización de algunos tipos de orbitas de cuerpos celestes en la cosmología, me pareció también muy interesante dentro de mi investigación sobre este tipo de curvas sus trayectorias homólogas que se diferencian en que la circunferencia de radio menor gira por fuera de la de radio mayor llamadas epicicloides, de igual manera Plotly me parece una excelente alternativa a matplotlib en cuanto a lo realizado en esta práctica respecta dado que originalmente realicé mis gráficas con matplotlib pero no podía realizar un zoom en específico sobre algunas de ellas o verificar si los valores eran correctos para cada trayectoria, cosa que se puede hacer solamente arrastrando el cursor con las gráficas generadas por plotly.

## Referencias Bibliográficas

1. Kühnel, W. (2015). Differential geometry (Vol. 77). American Mathematical Soc..
2. Carver, W. B. (1956). Hypocycloids. The American Mathematical Monthly, 63(9), 58-67.
3. George F. Simmons (1992). Calculus Gems. Chapter B.21: The Cycloid.
4. ProofWiki. Equation of Hypocicloid [https://proofwiki.org/wiki/Equation\\_of\\_Hypocycloid](https://proofwiki.org/wiki/Equation_of_Hypocycloid)
5. Weisstein, E. W. (2003). Hypocycloid. <https://mathworld.wolfram.com/>.