ASTRODINÁMICA PARA TODOS (2ª PARTE)

PYTHON COMO HERRAMIENTA CLAVE DE APRENDIZAJE EN CIENCIA

—

EN LA SEGUNDA PARTE DE ESTE ARTÍCULO AVANZAREMOS EN EL USO DE POLIASTRO, REPASANDO CONCEPTOS COMO LA VELOCIDAD DE ESCAPE Y LAS FAMOSAS ÓRBITAS DE TRANSFERENCIA DE HOHMANN TAN FRECUENTEMENTE NOMBRADAS Y QUE, COMO EL LECTOR PODRÁ DESCUBRIR, NO TIENEN NINGÚN SECRETO CON POLIASTRO.

CARLOS MORALES SOCORRO, DANIEL MARÍN ARCONES Y MIGUEL LEÓN SANTANA

1. ¡ESCAPA! ¡ESCAPA!

Tal y como sabemos, la segunda velocidad cósmica o velocidad de escape de un objeto de masa m de un objeto masivo de masa M, siendo r la distancia inicial que separa los centros de masa de ambos cuerpos, viene dada por la fórmula:

$$v_{esc} = \sqrt{\frac{2GM}{r}};$$

donde, como se puede observar, no aparece la masa del objeto fugitivo. Así pues, y siguiendo el ejemplo planteado en la referencia [1], resulta muy sencillo calcular la velocidad de escape de un cuerpo situado en el punto P (-6045, -3490, 2500), en kilómetros con respecto al centro del planeta Tierra. Aplicando el Teorema de Pitágoras:

$$r = \sqrt{(-6045)^2 + (-3490)^2 + (2500)^2} \simeq 7414 \text{ km} = 7{,}414 \times 10^6 \text{m};$$

por lo que, sustituyendo valores,

$$G = 6{,}674 \times 10^{11} \text{ Nm}^2 \text{ kg}^2,\ M_T = 5{,}972 \times 10^{24} \text{ kg: } v_{esc} = \sqrt{\frac{2GM_T}{r}} \rightarrow v_{esc} \simeq 10{,}4 \text{ km s}^{-1}$$

Llega el turno de comprobarlo con Poliastro:

```
In [1]: import matplotlib.pyplot as plt
        from astropy import units as u
        from poliastro.util import norm
        from poliastro.bodies import Earth
        from poliastro.twobody import Orbit
```

FIGURA 1A. Carga de librerías para el cálculo de la velocidad de escape. (Todas las imágenes son cortesía de Carlos Morales Socorro)

```
In [2]: r = [-6045, -3490, 2500] * u.km
        norm(r)
Out[2]: 7414.3189 km
```

FIGURA 1B. Cálculo de la distancia r.

Veamos qué ocurre con una velocidad de 10,3 km s^{1}. Deberíamos obtener una órbita elíptica de alta excentricidad, pero sin llegar a escapar. ¡Efectivamente!

```
In [3]: v = [0, 0, 10.3] * u.km / u.s
        ss = Orbit.from_vectors(Earth, r, v)
        ss.plot()
        ss.ecc
Out[3]: 0.97643191
```

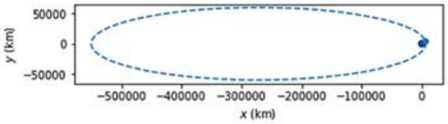

FIGURA 1C. Órbita elíptica de alta excentricidad.

¿Y qué ocurre si la velocidad es de 10,5 km s^{1}? Deberíamos obtener una órbita hiperbólica con una excentricidad ligeramente superior a 1. Tal y como esperábamos, el objeto logra escapar de la atracción terrestre.

```
In [4]: v = [0, 0, 10.5] * u.km / u.s

        ss = Orbit.from_vectors(Earth, r, v)
        ss.plot()
        ss.ecc
Out[4]: 1.0451015
```

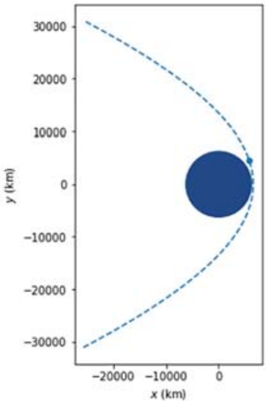

FIGURA 1D. El objeto logra escapar describiendo una órbita hiperbólica.

2. LA MANIOBRA DE HOHMANN

En este último apartado mostraremos cómo llevar al aula o, en su defecto, disfrutar como aficionados a la astronomía, de los primeros pasos del análisis de vuelos espaciales. Para ello adaptaremos el ejemplo descrito en MacDougal, D.W., (2012), referencia [2], donde se desea diseñar una maniobra que cambie la órbita de un satélite de comunicaciones inicialmente en órbita circular ecuatorial, a una altura de 500 km, a una órbita final circular geoestacionaria.
El alumnado empezará encontrando la velocidad circular inicial:

v_i: $v = \sqrt{\frac{GM_T}{r}}$, donde sustituyendo r = 6878 km = 6,878 x 10 m, se obtiene: $v_i \approx 7{,}612$ km s^1

Continuará calculando el semieje mayor de la órbita final apoyándose en la expresión simplificada de la formulación de Newton de la Tercera Ley de Kepler:

$\frac{P^2}{a^3} = \frac{4\pi^2}{GM_T} \rightarrow a = \sqrt[3]{\frac{P^2 GM_T}{4\pi^2}}$, con P = 23^h 56^m 4,09^s = 86 164,09 s $\rightarrow$ a ≈ 42 163 km;

con una velocidad circular final, v_f, de $v = \sqrt{\frac{GM_T}{r}} \rightarrow v_f \approx 3{,}075$ km s^1

La cuestión ahora radicará en encontrar la órbita elíptica de transferencia de Hohmann, que permita pasar de la primera órbita a la segunda:

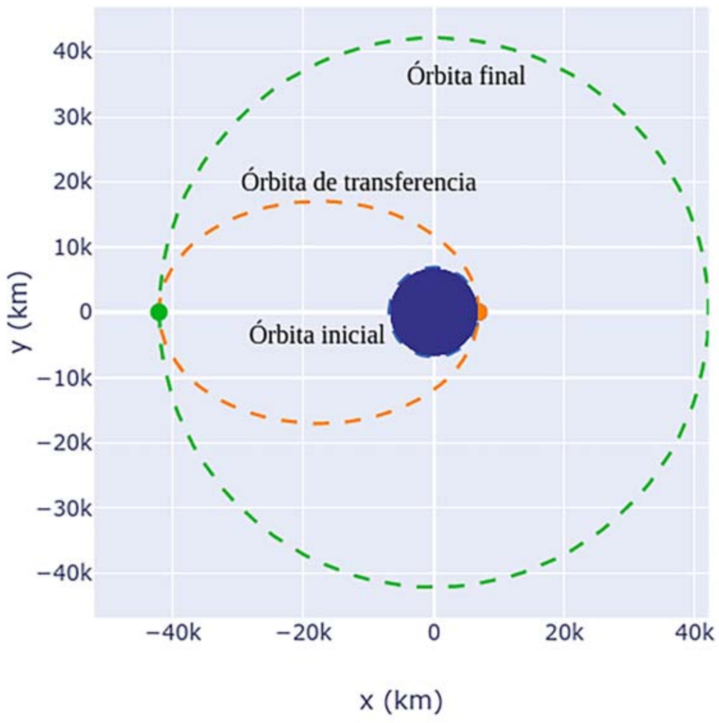

FIGURA 2A. Órbita inicial, de transferencia y final.

Dicha órbita tendrá una excentricidad $e = \frac{Q-q}{Q+q}$, siendo Q la distancia al satélite en apogeo y q en perigeo, que son precisamente los radios de ambas órbitas circulares. Por lo que e ≈ 0,7194.

Procedamos ahora con la obtención de las velocidades en perigeo, vP, y en apogeo, vA necesarias para producir dicha órbita:

$$v_P = \sqrt{GM_T \frac{(1+e)}{q}} \longrightarrow v_P \approx 9{,}980 \text{ km s}^1 \quad \text{y } v_A = \sqrt{GM_T \frac{(1-e)}{Q}} \longrightarrow v_A \approx 1{,}629 \text{ km s}^1$$

Así, la maniobra consistirá en proporcionar el cambio de velocidad, ΔvP, necesario para pasar de la órbita inicial a la de transferencia, en perigeo: ΔvP = vP – vi ΔvP ≈ 9,980 – 7,612 km s^1 = 2,368 km s^1; así como ΔvA, necesaria para pasar de la de transferencia, en apogeo, a la final: ΔvA = vf – vA ΔvA ≈ 3,075 – 1,629 km s^1 = 1,446 km s^1.

Y ahora, hagámoslo con Poliastro:

```
In [1]: import matplotlib.pyplot as plt
        from astropy import units as u
        from poliastro.util import norm
        from poliastro.bodies import Earth
        from poliastro.twobody import Orbit
        from poliastro.maneuver import Maneuver
```

FIGURA 2B. Carga de librerías para maniobra de Hohmann.

```
In [3]: so500 = Orbit.circular(Earth, alt=500 * u.km)
        so500.ecc

Out[3]: 0
```

FIGURA 2C. Cálculo de órbita inicial y comprobación de excentricidad nula.

```
In [7]: v500 = norm(so500.v.to(u.km / u.s))
        v500

Out[7]: 7.6126084 km/s
```

FIGURA 2D. Cálculo de velocidad circular inicial.

```
In [9]: # sogeo, órbita circular ecuatorial geosíncrona a una altitud de 42163 - 6380 = 35783 km
        sogeo = Orbit.circular(Earth, alt=35783 * u.km)
        sogeo.ecc

Out[9]: 0
```

FIGURA 2E. Cálculo de órbita circular final y comprobación de excentricidad nula.

```
In [11]: vgeo = norm(sogeo.v.to(u.km / u.s))
         vgeo

Out[11]: 3.0747707 km/s
```

FIGURA 2F. Cálculo de velocidad circular final.

```
In [23]: sohohmann_man = Maneuver.hohmann(so500, 42163 * u.km)
         sohohmann_man.impulses

Out[23]: ((<Quantity 0. s>,
           <Quantity [   0.         , 2369.76879851,    0.        ] m / s>),
          (<Quantity 19106.30832785 s>,
           <Quantity [    0.         , -1446.25709591,     0.        ] m / s>))
```

FIGURA 2G. Cálculo de la órbita de transferencia de Hohmann.

Tal y como se puede observar, parece que el cambio ΔvA debe conseguirse en t ≈ 19 106 s. ¿Cómo se ha obtenido ese valor? Si recordamos la Figura 2a, es evidente que el instante en el que deberemos cambiar de la órbita elíptica a la circular final será el correspondiente a la mitad del periodo de la órbita de transferencia, cuando se alcanza el apogeo, el cuál puede obtenerse fácilmente usando la Tercera Ley de Kepler con:

$$a = \frac{Q+q}{2};\ P \approx 38\,216\ \mathrm{s} \rightarrow \frac{P}{2} \simeq 19\,108\ \mathrm{s}$$

```
In [35]: vp = norm(sotransferencia.v.to(u.km / u.s))
         sotransferencia_end = sotransferencia.propagate(19108 * u.s)
         va = norm(sotransferencia_end.v.to(u.km / u.s))
         print("Velocidad en perigeo: {}; Velocidad en apogeo = {}".format(vp,va))

         Velocidad en perigeo: 9.982377193088181 km / s; Velocidad en apogeo = 1.628445681447899 km / s
```

FIGURA 2H. Cálculo de las velocidades en perigeo y apogeo de la órbita de transferencia.

```
In [36]: sotransferencia.ecc

Out[36]: 0.71949522
```

FIGURA 2I. Cálculo de la excentricidad de la órbita de transferencia.

Conclusiones

En estos dos artículos hemos dado nuestros primeros pasos en el mundo de la programación en Python y de la astrodinámica, tanto a nivel de aficionado a la astronomía como, tal y como se ha comentado anteriormente, en entornos educativos, especialmente de la etapa de Bachillerato. Y es que la programación en general, y Python en particular, son un complemento perfecto para el aprendizaje de las matemáticas, la física o la iniciación a la astronomía, dentro y fuera del aula.

Referencias

[1] *docs.poliastro.space/en/stable/user_guide.html*
[velocidad de escape] *dropbox.com/s/96m7wbvqugyqzi8/Velocidad%20de%20escape.ipynb?dl=0*
[2] MacDougal, D. W., (2012). *Newton's Gravity. An Introductory Guide to the Mechanics of the Universe.* Capítulo 14. Springer, New York.

Carlos Morales Socorro y **Miguel León Santana** son profesores de matemáticas en el IES José Frugoni Pérez. **Daniel Marín Arcones** es astrofísico, uno de los mayores expertos en astronáutica de nuestro país. Los tres son miembros de la Agrupación Astronómica de Gran Canaria.