



eul2m_exemple.py

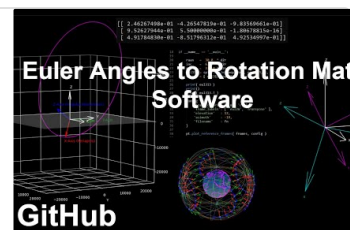


path: .../AWP/src/nmwp009_eul2m/eul2m_exemple.py

- Vídeo que o autor do código mostra suas funcionalidades:

Euler Angles to Rotation Matrix Software | Numerical Methods with Python 9

https://www.youtube.com/watch?v=9GXxXuRgpp4&list=PLOIRBaljOV8gsvlQ_GtiDRSBECHB2vvnp&index=10&t=9s



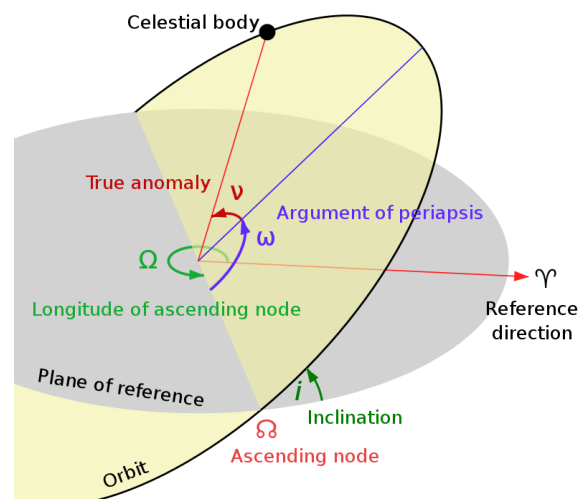
Documentação

Este código calcula a matriz resultante mediante uma rotação de ângulos de euler e printa uma animação dessa rotação referente a um eixo inercial inicial.

O criador do código programou para uma rotação 313, ou seja, rotação referente ao eixo z inercial, ao novo eixo x e ao novo eixo z.

Para esse exemplo, o autor usou os ângulos que definem os elementos da órbita de Kepler[1].

1. raan → Right Ascension of the Ascending: como estamos em órbita geocêntrica, sendo o plano equatorial da terra o plano de referencia, esse ângulo é medido para o leste desde o Ponto Áries(ponto vernal) até o nó ascendente da órbita[1.1]. Ω na imagem.
2. inc → Inclinação: tilt vertical da elipse em respeito ao plano de referencia, medido no nó ascendente. i na imagem
3. aop → Argumento de Periapsis: define a orientação da elipse do plano orbital. ω na imagem



Esses 3 ângulos juntos podem ser descritos como os ângulos de Euler definindo a orientação da órbita em relação ao sistema de coordenadas de referência.

Aplica esses ângulos na função do `spiceypy.eul2m`

```
spiceypy.spiceypy.eul2m(angle3, angle2, angle1, axis3, axis2, axis1) \[source\]
```

Construct a rotation matrix from a set of Euler angles.

http://naif.jpl.nasa.gov/pub/naif/toolkit_docs/C/cspice/eul2m_c.html

Parameters:

- `angle3` (*float*) – Rotation angle about third rotation axis (radians).
- `angle2` (*float*) – Rotation angle about second rotation axis (radians).
- `angle1` (*float*) – Rotation angle about first rotation axis (radians).
- `axis3` (*int*) – Axis number of third rotation axis.
- `axis2` (*int*) – Axis number of second rotation axis.
- `axis1` (*int*) – Axis number of first rotation axis.]

Returns: Product of the 3 rotations.

Return type: 3x3-Element Array of floats

Documentação da Função:

spiceypy.spiceypy - SpicePy 2.3.1 documentation

```
docs]@spiceErrorCheck @spiceFoundExceptionThrower def bodc2n ( code , lenout = _default_len_out ): """ Translate the
SPICE integer code of a body into a common name for that body.
http://naif.jpl.nasa.gov/pub/naif/toolkit_docs/C/cspice/bodc2n_c.html :param code: Integer ID code to be translated into a name.
📖 https://spiceypy.readthedocs.io/en/v2.3.1/_modules/spiceypy/spiceypy.html#eul2m
```

O que nos interessa é a rotação ativa e esta função nos fornece a rotação passiva [2]. Então, para obter o resultado que desejamos pegamos a transposta da matriz resultante.

printa os dois resultados

Por fim, faz a animação do gráfico dessas rotações, em que usando a função da biblioteca autoral dele `plotting_tools`, ele fornece como argumento as 2 matrizes (rot pass e rot ativa) e as configurações que deseja para esse gráfico que substituirão as default.

Referências

▼ [1]

A órbita de Kepler é uma idealização, aproximação matemática de uma órbita em um certo momento.

Orbital elements - Wikipedia

Orbital elements are the parameters required to uniquely identify a specific orbit. In celestial mechanics these elements are considered in two-body systems using a Kepler orbit. There are many different ways to mathematically describe the same orbit, but certain schemes, each consisting of a set of six parameters, are commonly used in astronomy and orbital mechanics.

W https://en.wikipedia.org/wiki/Orbital_elements

[1.1] é o nodo em que o corpo menor em órbita passa do hemisfério sul para o norte (relativamente ao plano de referência)

▼ [2]

- rotação ativa: algum aparato mecanico ou eletronico no satélite permitirá que possamos ajustar a posição do satélite
- rotação passiva: o satélite tem uma certa configuração de posição em órbita sem que nenhum controle ou operação requerida para obtê-la.