Fiddlin' John Carson

General Information

The service provides you with the two cartesian state vectors (ICRF), a position and a velocity vector of our satellite after connecting via SSH. In addition, the time is given. To solve the task, you have to provide the orbit expressed as Keplerian elements.

Solution

The task can be solved in different ways.

One way would be to calculate the orbit like provided in [1] by hand. Here we have to take the units into account, since [1] uses meters and meters per second and our vectors are in kilometers and kilometers per second.

Another way is to simply input the two vectors into the online calculator on [2] and copy the solution.

The preferred way is to solve this task with Astropy [3] and Poliastro [4], since the code can be reused for the next task in this category. To get the Keplerian elements, the two vectors are defined and an orbit with the Poliastro function Orbit.from_vectors() with the earth as the central body is created. After that, the Keplerian elements of the orbit can be converted to the desired units and printed out.

Code

```
from astropy import units as u

from poliastro.bodies import Earth
from poliastro.twobody import Orbit
from astropy import time

r = [8449.401305, 9125.794363, -17.461357] * u.km
v = [-1.419072, 6.780149, 0.002865] * u.km / u.s
epoch = time.Time("2021-06-26 19:20:00")

orb = Orbit.from_vectors(Earth, r, v, epoch=epoch)

print('a: ', orb.a)
print('e: ', orb.ecc)
print('i: ', orb.inc.to(u.deg))
print('\u03A9: ', orb.raan.to(u.deg))
print('\u03C9: ', orb.argp.to(u.deg))
print('v: ', orb.nu.to(u.deg))
```

Sources

- [1] O. Montenbruck and G. Eberhard, Satellite orbits: models, methods, and applications. Berlin: New York: Springer, 2000.
- [2] "Orbital Mechanics Calculator: Calculator." https://elainecoe.github.io/orbital-mechanics-calculator/calculator.html (accessed Jul. 16, 2021).
- [3] "Astropy." https://www.astropy.org/ (accessed Jul. 16, 2021).
- [4] poliastro/poliastro. poliastro, 2021. Accessed: Jul. 16, 2021. [Online]. Available: https://github.com/poliastro/poliastro