Celestools

Release 1.0.5

Alexis Petit

Apr 22, 2020

CONTENTS:


Share My Space
Space debris removal & management

CHAPTER

ONE

INTRODUCTION TO CELESTOOLS

This Python package contains modules useful for celestial mechanics and in particular for orbital flight.

Last update: 2020/04/06 by Alexis Petit, Delphine Ly.

- Version 1.0.0: Initial version from CelestialPy (constants, coordinates, time scales, TLE).
- Version 1.0.1: Add coordinate and frame transformations using Orekit.
- Version 1.0.2: Determination of the satellite coordinate systems RSW and NWT as defined by Vallado, 2013, p.157.
- Version 1.0.3: Modification of the keys.
- Version 1.0.4: Vectorization of the conversion modules. Add arguments in the TLE reader.
- Version 1.0.5: Reference frame transformation (with Orekit and Astropy).

CHAPTER

TWO

INSTALLATION

From the repo

```
$ git clone ssh://git@SMS2:/srv/git/celestools.git
```

Install the CelesTools Python package (setup.py in the package directory)

```
$ python setup.py install
```

To remove

```
$ pip uninstall
```

CHAPTER

THREE

BENCHMARK

In the directory *tests*, you check the package with

```
$ python vallado.py
```

CHAPTER

FOUR

DESCRIPTION OF THE MODULES

This chapter describes the main modules of the Celestools code.

coordinates.py

- Authors: Alexis Petit, PhD Student, Namur University
- Date: 2017/03/09
- It allows to perform coordinate transformations.

celestools.coordinates.**cart_2_kepl**(*cart*, *mu*, *unit='rad'*)
Return keplerian elements from cartesian coordinates.

Parameters

- **cart** (*float array*) – Cartesian coordinates (r[m],v[m/s]).
- **mu** (*float*) – Gravitational constant time the mass of the central body.
- **unit** (*string*) – Unit of the angles ('rad' or 'deg').

Returns **kepl** – Orbital elements (a,e,i,raan,omega,ma).

Return type float array

celestools.coordinates.**kepl_2_cart**(*kepl*, *mu*, *unit='rad'*)
Return cartesian coordinates from orbital elements.

Parameters

- **kepl** (*float array*) – Orbital elements (a,e,i,raan,omega,ma).
- **mu** (*float*) – Gravitational constant time the mass of the central body.
- **unit** (*string*) – Unit of the angles ('rad' or 'deg').

Returns **cart** – Cartesian coordinates (x[m],v[m/s])

Return type float array

celestools.coordinates.**df_kepl_2_cart**(*ic_kepl*, *unit='rad'*)
Return a dataframe of cartesian coordinates from a dataframe of orbital elements.

Parameters

- **ic_kepl** (*dataframe*) – Orbital elements (a,e,i,raan,omega,ma).
- **unit** (*string*) – Unit of the angles ('rad' or 'deg').

Returns **ic_cart** – Cartesian coordinates (x[m],v[m/s]).

Return type dataframe

celestools.coordinates.**df_cart_2_kepl**(*ic_cart*, *unit='rad'*)
Return a dataframe of orbital elements from a dataframe of cartesian coordinates.

Parameters

- **ic_cart** (*dataframe*) – Cartesian coordinates (x[m],v[m/s]).
- **unit** (*str*) – Unit of the angles ('rad' or 'deg').

Returns **ic_kepl** – Orbital elements (a,e,i,raan,omega,ma).

Return type dataframe

celestools.coordinates.**cart_2_sph**(*cart*)
Return spherical coordinates from cartesian coordinates.

Parameters **cart** (*float array*) – Cartesian coordinates (x[m],v[m/s]).

Returns **kepl** – Spherical coordinates (alt,long,lat,dalt,dlong,dlat).

Return type float array

celestools.coordinates.**sph_2_cart**(*sph*)
Return cartesian coordinates from spherical coordinates.

Parameters **sph** (*float array*) – Spherical coordinates (alt,long,lat,dalt,dlong,dlat)

Returns **cart** – Cartesian coordinates (x[m],v[m/s])

Return type float array

celestools.coordinates.**kepl_2_equinoctial**(*kepl*, *mu*)
Return equinoctial coordinates from orbital elements. Ref: Vallado, 2012, p. 108

Parameters

- **kepl** (*float array*) – Keplerian elements (a,e,i,raan,omega,ma).
- **mu** (*float*) – Gravitational constant time the Earth mass.

Returns **equinoc** – Equinoctial coordinates (a,k,h,l,p,q)

Return type float array

celestools.coordinates.**mean_anomaly_2_ecc_anomaly**(*e*, *ma*, *precision=1e-14*, *unit='rad'*)
Compute the eccentric anomaly from the mean anomaly.

ecc: float Excentricity

ma: float Mean anomaly

Returns **ea** – Eccentric anomaly

Return type float

celestools.coordinates.**true_anomaly_2_ecc_anomaly**(*ecc*, *nu*, *unit='rad'*)
Compute the eccentric anomaly from the true anomaly.

ecc: float Eccentricity

nu: float True anomaly

unit: string Unit of the angle.

Returns **ea** – Eccentric anomaly

Return type float

celestools.coordinates.**ecc_anomaly_2_mean_anomaly**(*ecc*, *ea*, *unit='deg'*)
Compute the mean anomaly from the eccentric anomaly.

ecc: float Eccentricity

ea: float Eccentric anomaly

Returns **ma** – Mean anomaly

Return type float

celestools.coordinates.**true_anomaly_2_mean_anomaly**(*ecc*, *nu*, *unit='rad'*)
Compute mean anomaly from the true anomaly.

ecc: float Eccentricity

nu: float True anomaly

Returns **ea** – Eccentric anomaly

Return type float

celestools.coordinates.**ecc_anomaly_2_true_anomaly**(*ecc*, *ea*, *unit='rad'*)
Compute the true anomaly from the eccentric anomaly.

ecc: float Eccentricity

ea: float Eccentric anomaly

Returns **nu** – True anomaly

Return type float

celestools.coordinates.**mean_anomaly_2_true_anomaly**(*ecc*, *ma*, *tolerance=1e-14*)
Compute the true anomaly from the mean anomaly.

ecc: float Excentricity

ma: float Mean anomaly

Returns **nu** – True anomaly

Return type float

celestools.coordinates.**flight_path_angle**(*ecc*, *ea*, *unit='rad'*)
Compute the flight path angle.

ecc: float Eccentricity.

ea: float Eccentric anomaly.

unit: string Unit of the angle.

Returns **fpa** – Flight path angle.

Return type float

celestools.coordinates.**dms_2_rad**(*dms*)
Give an angle in radians from an angle in degrees, minutes, and seconds.

dms: array of float Degrees, minutes, and seconds

Returns **angle** – Angle in radians.

Return type float

celestools.coordinates.**hms_2_rad**(*hms*)
Give an angle in radians frim an angle in hours, minutes, and seconds.

hms: array of float Hours, minutes, and seconds

Returns **angle** – Angle in radians.

Return type float

celestools.coordinates.**degree_2_hms**(*angle*)
Give an angle in hours, minutes, and seconds from an angle in degress.

angle: float Angle in degress.

Returns

- **hh** (*float*) – Hours.
- **mm** (*float:*) – Minutes.
- **ss** (*float*) – Seconds.

celestools.coordinates.**degree_2_dms**(*angle*)
Give an angle in degress, minutes, and seconds from an angle in degress.

angle: float Angle in degress.

Returns

- **dd** (*float*) – Degress.
- **mm** (*float:*) – Minutes.
- **ss** (*float*) – Seconds.

celestools.coordinates.**geocentric_2_topocentric**(*theta_lst*, *lat_gd*, *r_geo*, *unit='rad'*)
Compute the position from the geocentric frame to the topocentric frame (Vallado, p.162).

theta_lst: float Local sideral time [deg]

lat_gd: float Geodedic latitude [deg].

r_geo: array of float Position in the geocentric frame.

Returns **r_topo** – Position in the topocentric frame.

Return type array of float

celestools.coordinates.**topocentric_2_geocentric**(*theta_lst*, *lat_gd*, *r_topo*, *unit='rad'*)
Compute the position from the topocentric frame to the geocentric frame (Vallado, p.162).

theta_lst: float Local sideral time.

lat_gd: float Geodedic latitude.

r_topo: array of float Position in the topocentric frame.

Returns **r_geo** – Position in the geocentric frame.

Return type array of float

celestools.coordinates.**latlon_2_ecef**(*phi_gd*, *lamb*, *h_ell*, *unit='rad'*)
Return cartesian coordinate of a station from the geodedic longitude, latitude and altitude. Ref: Vallado, p.144.

Parameters

- **phi_gd** (*float*) – Decimal geodetic latitude.
- **lamb** (*float*) – Decimal longitude.
- **h_ell** (*float*) – Ellipsoidal height [m].
- **unit** (*string*) – Unit of the angles ('rad' or 'deg').

Returns **r** – Cartesian coordinates in [m] and [m/s].

Return type array of float

celestools.coordinates.**ecef_2_latlon**(*r_ecef*)
Return longitude, latitude and altitude from cartesian coordinates in the ECEF frame. Ref: Algorithm 12, p.172 in Vallado 2012.

Parameters **r_ecef** (*float array*) – Cartesian coordinates in the ECEF frame [m].

Returns

- **lon** (*float*) – Longitude [rad]
- **lat** (*float*) – Geodetic latitude [rad]

celestools.coordinates.**rotation**(*alpha*, *axe*, *unit='rad'*)
Return a matrix rotation. Ref: Vallado p.162

Parameters

- **alpha** (*float*) – Angle of rotation.
- **axe** (*integer*) – Axe of rotation.
- **unit** (*string*) – Unit of the angles ('rad' or 'deg').

Returns **rot** – Matrix rotation.

Return type matrix of float [3x3]

celestools.coordinates.**convert_ephemeris_2_snapshot**(*list_ephemeris*)
Return a list of snapshots from a list of ephemerides.

Parameters **list_ephemeris** (*list of dataframes*) – List of ephemeris.

Returns **list_snapshots** – List of snapshots.

Return type list of dataframes

CHAPTER

FIVE

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

C

INDEX