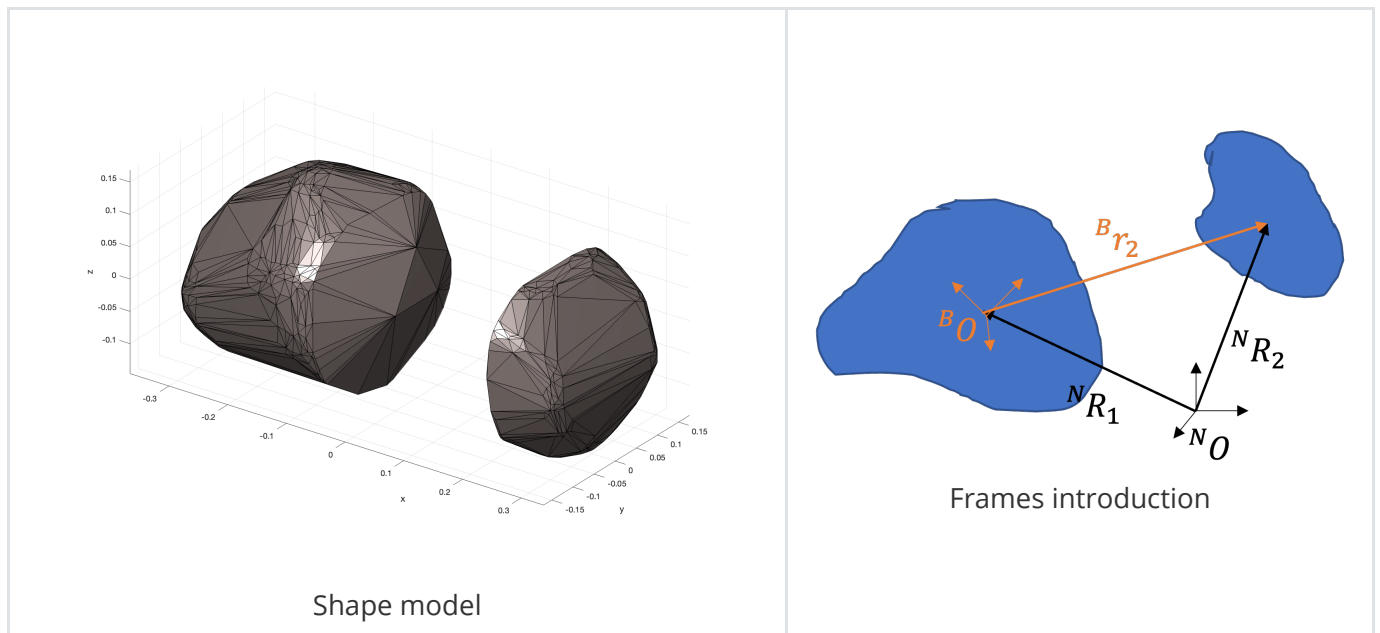


Questions20230911

1. Pykdgrav can't run in the new version code.
- 2.

```
Traceback (most recent call last):
  File "/opt/local/Library/Frameworks/Python.framework/Versions/3.9/lib/python3.9/site-packages/IPython/core/interactiveshell.py", line 3588, in run_code
    exec(code_obj, self.user_global_ns, self.user_ns)
  File "<ipython-input-2-5ff2e96c9b94>", line 1, in <module>
    runfile('/Users/hai/Desktop/PhD/Research/Granular_Sim_Test/PolyhedronAsteroid/code/main_computation_v7.py', wdir='/Users/hai/Desktop/PhD/Research/Granular_Sim_Test/PolyhedronAsteroid/code')
  File "/Applications/PyCharm.app/Contents/plugins/python/helpers/pydev/_pydev_bundle/pydev_umd.py", line 198, in runfile
    pydev_imports.execfile(filename, global_vars, local_vars) # execute the script
  File "/Applications/PyCharm.app/Contents/plugins/python/helpers/pydev/_pydevimps/_pydev_execfile.py", line 18, in execfile
    exec(compile(contents+"\n", file, 'exec'), glob, loc)
  File "/Users/hai/Desktop/PhD/Research/Granular_Sim_Test/PolyhedronAsteroid/code/main_computation_v7.py", line 21, in <module>
    from pykdgrav import *
  File "/Applications/PyCharm.app/Contents/plugins/python/helpers/pydev/_pydev_bundle/pydev_import_hook.py", line 21, in do_import
    module = self._system_import(name, *args, **kwargs)
  File "/Users/hai/Desktop/PhD/Research/Granular_Sim_Test/PolyhedronAsteroid/code/pykdgrav/_init_.py", line 2, in <module>
    from .kdtree import *
  File "/Applications/PyCharm.app/Contents/plugins/python/helpers/pydev/_pydev_bundle/pydev_import_hook.py", line 21, in do_import
    module = self._system_import(name, *args, **kwargs)
  File "/Users/hai/Desktop/PhD/Research/Granular_Sim_Test/PolyhedronAsteroid/code/pykdgrav/kdtree.py", line 1, in <module>
    from numba import int32, deferred_type, optional, float64, boolean, int64, njit, jit, prange
  File "/Applications/PyCharm.app/Contents/plugins/python/helpers/pydev/_pydev_bundle/pydev_import_hook.py", line 21, in do_import
    module = self._system_import(name, *args, **kwargs)
  File "/opt/local/Library/Frameworks/Python.framework/Versions/3.9/lib/python3.9/site-packages/numba/_init_.py", line 42, in <module>
    from numba.np.ufunc import (vectorize, guvectorize, threading_layer,
  File "/Applications/PyCharm.app/Contents/plugins/python/helpers/pydev/_pydev_bundle/pydev_import_hook.py", line 21, in do_import
    module = self._system_import(name, *args, **kwargs)
  File "/opt/local/Library/Frameworks/Python.framework/Versions/3.9/lib/python3.9/site-packages/numba/np/ufunc/_init_.py", line 3, in <module>
    from numba.np.ufunc.decorators import Vectorize, GUVectorize, vectorize, guvectorize
  File "/Applications/PyCharm.app/Contents/plugins/python/helpers/pydev/_pydev_bundle/pydev_import_hook.py", line 21, in do_import
    module = self._system_import(name, *args, **kwargs)
  File "/opt/local/Library/Frameworks/Python.framework/Versions/3.9/lib/python3.9/site-packages/numba/np/ufunc/decorators.py", line 3, in <module>
    from numba.np.ufunc import _internal
  File "/Applications/PyCharm.app/Contents/plugins/python/helpers/pydev/_pydev_bundle/pydev_import_hook.py", line 21, in do_import
    module = self._system_import(name, *args, **kwargs)
SystemError: initialization of _internal failed without raising an exception
```

3. I am trying to find some way to track the second body's movement at the primary body-fixed frame. The shape mode is two polyhedra moving based on gravity in space. But I can't find a correct way to define the body-fixed frame on one body. (I suppose the large one is primary, smaller one is secondary)



Two ways I tried, One use the position of primary as the origin, then choose a vertex position $\vec{x} = \vec{v}_1$ relative to the origin as the x-axis. Then choose another vertex position relative to origin \vec{v}_2 to get the z-axis $\vec{z} = \vec{v}_1 \times \vec{v}_2$, finally we can get y-axis from $\vec{y} = \vec{z} \times \vec{x}$.

But, the problem is that I can not get the vertex information from 'chipy' library in computation.py. Are there some functions that can return the all body's vertex during computation?

The other one is that I tried to use the Direction Cosine Matrix ('et la matrice de passage') provided in Line 452 of avatar.py to rotate the position vector from the inertial frame to the body-fixed frame. The stacking of eigenvalues of global inertia ('P' variable in code) should be the Direction Cosine Matrix, which can rotate the vector from the inertial frame to the body frame.

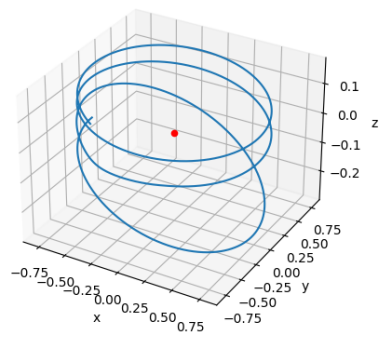
$${}^B\vec{r}_2 = {}^N\vec{R}_2 - {}^N\vec{R}_1 = P({}^N\vec{R}_2 - {}^N\vec{R}_1) \quad (1)$$

Where ${}^B\vec{r}_2$ is the secondary's position at the primary's body-fixed frame, ${}^N\vec{R}_{1/2}$ is the primary/secondary position at the inertial frame. The body-fixed frame is fixed on the primary with the same rotation.

But, The rotation matrix would be wrong. In this animation, the secondary should be stationary relative to the primary, but when I use the above equation to get the position relative to the primary, it jumps around the primary.

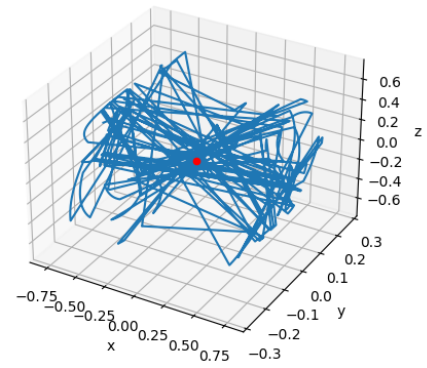


vector changes



Secondary's orbit at inertial frame

vector changes



Secondary's orbit at Primary's body-fixed frame