Solution - Session 4 Report

Anirudh Agarwal

November 23, 2023

Abstract

First things first, the files to be uploaded to the google drive for submission are:

1. Overleaf source file in a separate folder
2. PDF report generated with overleaf
3. Mendeley references pdfs/ docs/ datasheets/ other in a separate references folder
4. Python codes, inputs, and outputs, in a separate folder

1 Objectives for this Session

The objectives of this session are:

1. Complete the code and search for session 3 based on the solutions provided.
2. Find orbital speeds at various heights and ground speed for various latitudes.
3. Find encapsulant suppliers.
4. Find reaction wheel balancing services.
5. Write programs to get max torque and max angular momentum based on the given equations.

2 Search

List of things searched:

- Orbital speed and ground speed.
- Encapsulant suppliers
- Reaction wheel balancing services

For orbital speed and ground speed, there were links provided in the problem statement. The purpose of this was to familiarize yourself with how ground speed changes with latitude and how orbital speed changes with height. The more equations and visualizations you find and can put as pictures in your report, the better!

2.1 Encapsulant Suppliers

Among the many suppliers contacted, Dow Silicone has shown some promising options. The way we confirm that they are promising are by reading NASA papers about the use of silicone materials including those from Dow in space. In particular, we see this paper [1] from NASA about silicone based materials which mentions Dow silicone materials by name. These can be used as encapsulants and adhesives as they have space flight heritage aka a proven track record.

However, silicones do not fair as well against the Atomic Oxygen that exists in Low Earth Orbit and we are still looking for sealants and epoxies that can counter that. We are also looking for alternatives for encapsulants, adhesives, and coatings just in case. Please include all the suppliers and materials you found in the search.

2.2 Reaction Wheel Balancing

There is one reaction wheel balancing service in Singapore that I found called Rotech Singapore. This service has the necessary options and is available within Singapore. We will still be looking at alternative wheel balancing options in India and Singapore. Please share the information you found for other options as well.

3 Code and Algorithms

This time the code is a lot simpler, however we still have the Reader, Writer, and Calculator files for dedicated functions. This is good practice to keep track, categorize, modularize, and most importantly, debug! No complicated objects this time, however, I make a lot of extra outputs on top of what is strictly necessary. This is for a better understanding for myself and to have more granular record of each step to make excel graphs later if needed.

Other things we need to cover are:

1. What the variables mean: already covered in the problem statement
2. What each function does
3. What are the inputs and outputs
4. A brief explanation of how the code works

3.1 Inputs

The inputs for this code are not so clear. Here are the input values used:

1. Heights: go from 400-600km in height in steps of 20km. This gives 11 height values 400 and 600 inclusive. Use np.linspace() function to get these 11 height values. Launches are available between 400km and 600km up so we choose this range. We expect our orbit to be at 500.
2. Moments of Intertia: go from $1.5kgm^2$ to $6.5kgm^2$ since we want them to be centered around $4kgm^2$ as that is the expected MOI for our craft. Just like heights, we can divide these into 11 values as well separated by units of 0.5.

Other than that, we use some physical inputs:
"physical inputs.txt"

- Value Name Description Link
- 6371000 : Rearth Radius of the Earth (m) $https://en.wikipedia.org/wiki/Earth$
- 5.972168E+24 : Mearth Mass of the Earth (kg) $https://en.wikipedia.org/wiki/Earth$
- 6.6743E-11 : Gconst Gravitational Constant (SI) $https://en.wikipedia.org/wiki/Gravitational\_constant$
- 86400 : Tday Time for a day (s) $https://en.wikipedia.org/wiki/Day$

3.2 Intermediate Values

There are several intermediate values generated here for various purposes. They are given as txt files.

3.3 Output Values

There are several values here as well, however, the most important ones are the torque and angular momentum values given by changing height and MOI.

The main purpose of this code, apart from getting data for visualizations, was to get the range of torques and angular momenta which can be used to size the ADCS system in the final session. Those outputs are given in the following file:

"Max Min TP.txt"

Torque (mNm)

Height \ MOI	MOI (kgm^2) 1.5	2	2.5	3	3.5	4	4.5	5	5.5	6	6.5
400	0.381	0.508	0.635	0.762	0.889	1.016	1.143	1.27	1.397	1.524	1.651
420	0.344	0.458	0.572	0.687	0.802	0.916	1.03	1.145	1.26	1.374	1.489
440	0.312	0.416	0.52	0.624	0.728	0.832	0.936	1.04	1.144	1.248	1.352
460	0.284	0.378	0.473	0.567	0.662	0.756	0.85	0.945	1.04	1.134	1.228
480	0.26	0.346	0.433	0.519	0.606	0.692	0.779	0.865	0.952	1.038	1.124
Height (km) 500	0.237	0.316	0.395	0.474	0.553	0.632	0.711	0.79	0.869	0.948	1.027
520	0.219	0.292	0.365	0.438	0.511	0.584	0.657	0.73	0.803	0.876	0.949
540	0.202	0.27	0.338	0.405	0.473	0.54	0.608	0.675	0.742	0.81	0.878
560	0.186	0.248	0.31	0.372	0.434	0.496	0.558	0.62	0.682	0.744	0.806
580	0.173	0.23	0.288	0.345	0.402	0.46	0.518	0.575	0.633	0.69	0.748
600	0.16	0.214	0.268	0.321	0.374	0.428	0.482	0.535	0.588	0.642	0.696

0.16	Minimum torque (mNm)
1.65	Maximum torque (mNm)
201,750.00	Minimum Angular Momentum (gcm^2 rad/s)
1,326,650.00	Maximum Angular Momentum (gcm^2 rad/s)

Angular Momentum (gcm^2 rad/s)

Height \ MOI	MOI (kgm^2) 1.5	2	2.5	3	3.5	4	4.5	5	5.5	6	6.5
400	306,150	408,200	510,250	612,300	714,350	816,400	918,450	1,020,500	1,122,550	1,224,600	1,326,650
420	291,300	388,400	485,500	582,600	679,700	776,800	873,900	971,000	1,068,100	1,165,200	1,262,300
440	277,650	370,200	462,750	555,300	647,850	740,400	832,950	925,500	1,018,050	1,110,600	1,203,150
460	265,350	353,800	442,250	530,700	619,150	707,600	796,050	884,500	972,950	1,061,400	1,149,850
480	253,950	338,600	423,250	507,900	592,550	677,200	761,850	846,500	931,150	1,015,800	1,100,450
Height (km) 500	243,450	324,600	405,750	486,900	568,050	649,200	730,350	811,500	892,650	973,800	1,054,950
520	233,850	311,800	389,750	467,700	545,650	623,600	701,550	779,500	857,450	935,400	1,013,350
540	224,850	299,800	374,750	449,700	524,650	599,600	674,550	749,500	824,450	899,400	974,350
560	216,600	288,800	361,000	433,200	505,400	577,600	649,800	722,000	794,200	866,400	938,600
580	208,950	278,600	348,250	417,900	487,550	557,200	626,850	696,500	766,150	835,800	905,450
600	201,750	269,000	336,250	403,500	470,750	538,000	605,250	672,500	739,750	807,000	874,250

Figure 1: Torque and Angular momentum values with minimum and maximum stated separately

- 0.16 : Minimum torque (mNm)
- 1.651 : Maximum torque (mNm)
- 201750.0 : Minimum Angular Momentum (gcm^2rad/s)
- 1326650.0 : Maximum Angular Momentum (gcm^2rad/s)

This is the file we use as input in the next session as one of the inputs.

3.4 Functions

The functions used by reader and writer are self explanatory as they read and write the stuff as mentioned in the function name.

The key functions in calculator are:

1. $get\_scalers()$: This uses the physical inputs and calculates the values that are only one variable for one height and not an array for instance, there is just one Vorbit for a given height but the theta, elevation angle, etc will be arrays with multiple values.
2. $get\_Walpha()$: this gives all the vectors i.e. theta[], elevationAng[], Wcraft[] and alpha[] arrays.
3. $get\_TP()$: This is to calculate Torque (T) and Angular Momentum (P) for all heights and MOIs. As the values are created, they are written on to the text files containing the torque and angular momentum values.

3.5 Output plots

4 Learning and Insights

The data structure for this new problem statement was not clearly defined. There are a lot of different things I wanted out of this, thus I made txt files storing a lot of data however, that was not clearly articulated in the problem statement. I will be more mindful of clarifying the requirements going forward. This also shows how this is a live project and we are still in the process of discovery. The best way is to complete the problem statement after the solution is also made, such that concessions for such data requirements can be made.

Another major thing to note is the syntax for the 3D plot and how the variables need to be defined in a certain way:

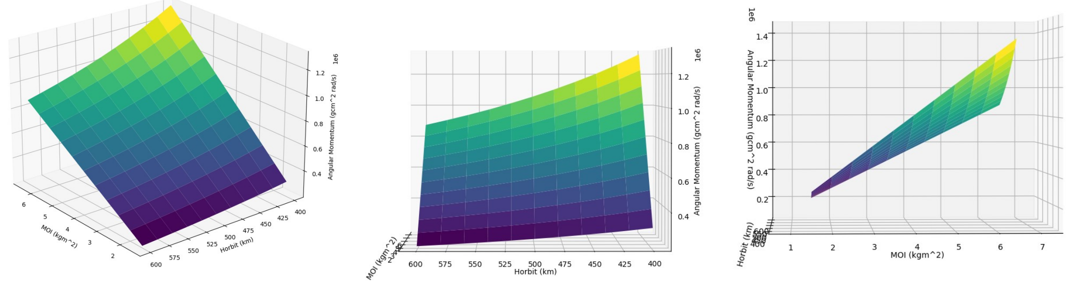

Figure 2: Different views of a 3D plot showing Angular momentum (gcm^2rad/s) vs MOI and Orbital height

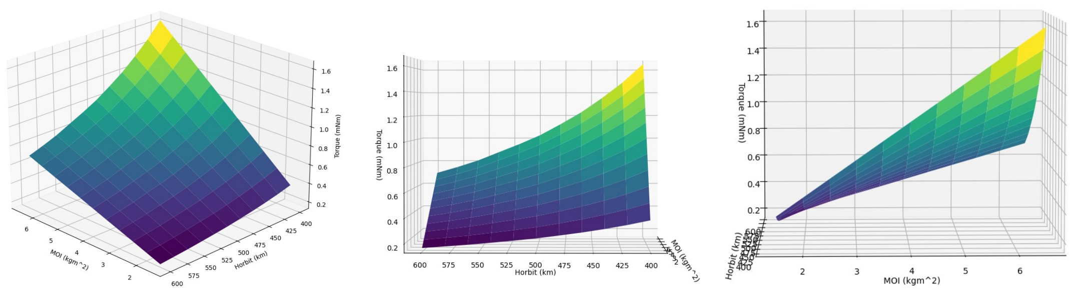

Figure 3: Different views of a 3D plot showing Torque (mNm) vs MOI and Orbital height

- Horbit and moi need to be defined with np.linspace()
- torque and Pcraft need to be remade using the np.array(¡var¿).reshape((row, col)) function.
- To preserve the values of the variables in previous elements, each row of the array must be individually declared inside the for loop as done in the *calculator.get_tp*() and the *reader.read_tp*() functions.
- Also note that to write values into a text file, everything must be converted to string. Upon reading the values from a text file, they must be converted to float/ int or another number type to make them work in math equations.

Please add any other tips or suggestions you found from your repeated struggle with Python!

Apart from code related learning and insights, I believe that giving search problems that are close to what we are looking for is most meaningful. However, since students may not be able to contact suppliers, pricing and other information may be hard to come by. Fortunately, for the materials for instance, the best source of information was the NASA paper evaluating the performance of various materials! Going forward, we can shift the focus from looking for suppliers to more general and research information, which may still yield unexpectedly accurate and helpful information.

References

[1] K. B. K. . P. H. G. Dursch, H. W. Evaluation of adhesive materials used on the long duration exposure facility. *NASA Contractor Report*, 1995.