Techno-Science Research Journal 9(2) (2021) 52-59

I.T.C

Content list available at ITC

Techno-Science Research Journal

Journal Homepage: http://techno-srj.itc.edu.kh/

Techno-Science Research Journal

Development of Orbital Simulator for Cambodian CubeSat Mission in LEO

Penghuy Srean[1*], Morokot Sakal[2], Maximilien Berthet[3], Sarot Srang[1]

[1] *Dynamics and Control Laboratory, Institute of Technology of Cambodia, Russian Federation Blvd., P.O. Box 86, Phnom Penh, Cambodia*

[2] *Faculty of Information Technology, American University of Phnom Penh, Street 201R, PO Box 1955, Phnom Penh, Cambodia*

[3] *Graduate School of Engineering, University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, 113-8656, Tokyo, Japan*

Received: 03 November 2021; Accepted: 15 December 2021; Available online: December 2021

Abstract: *Cambodia's first CubeSat Apsara-1 is being developed through an ongoing collaboration between the Institute of Technology of Cambodia and the University of Tokyo. In the early project phase, simulation methods are required for mission design. One important component, needed to predict the mission lifetime, is an orbital simulator. For example, Apsara-1 is planned to be launched into orbit in 2024 when solar activity is near its peak, leading to higher than usual atmospheric density in low Earth orbit (LEO) which results in the reduction of the flight time. It is important to predict how long the satellite will remain in orbit, both for effective operation planning and to meet international space regulations. In this paper, an in-house orbital simulator is developed for this purpose. The simulator includes all the dominant forces needed to model the orbital dynamics at Apsara-1's planned altitude in LEO: gravitational forces due to Earth's point mass and the J2 perturbation, as well as aerodynamic drag which determines the orbital lifetime of the satellite. The simulator is verified against a publicly available orbital simulation tool. Differences in the predicted orbital decay time between the in-house simulator and the existing tool are found to be less than 8%. The results show that Apsara-1 will remain in orbit for between 75 and 255 days, depending on the solar activity and the altitude at release from the ISS. In the nominal case, the mission will last for around 140 days. Beyond orbital prediction, it is shown that the in-house modular simulator can easily be expanded to perform evaluation of various satellite subsystems, such as the onboard electrical power and attitude control systems, which depend on the orbital motion. Overall, this numerical tool is expected to be valuable not only for Apsara-1 but for the future Cambodian satellite missions in LEO.*

Keywords: Astrodynamics; Orbital decay; Simulation platform

1. INTRODUCTION

As other countries around the world, Cambodia is affected by the climate change. Every year, Cambodia has encountered natural disaster such as flooding and drought that heavily affect the large number of Cambodian people. Providing a solution for prevention and mitigation would help developing Cambodia's economy and improve the social well-being of Cambodian people. One approach is to utilize space-based data. In the recent year, many developing countries has started developing CubeSat with hope that it could later provide a space-based platform for serving the purpose of disaster response.

Since 2019, the Dynamics and Control Laboratory (DCLab), a lab-based education provider located in the Institute of Technology of Cambodia (ITC). has undertaken a collaboration with the University of Tokyo to provide training workshops to ITC students on space engineering and technology, in view of promoting space research and development activity in Cambodia [1]. Most recently, in December 2020, the DCLab and the Department of Aeronautics and Astronautics at the University of Tokyo started the UT-ITC CubeSat Design Challenge, which aims to develop a concept design of the first Cambodian 1U CubeSat, Apsara-1. Apsara-1 is a student-led project to understand how to use space technology to address socio-economic challenges of Cambodia, and to promote STEM education by providing a learning platform for students to develop their skills in systems engineering and project management [2].

Apsara-1 is planned to be launched into orbit in 2024. Before any spacecraft is released into space, it has to be carefully

* Corresponding author: Penghuy Srean
E-mail: me.penghuy@gmail.com; Tel: +855-70 649 169

developed because once in space, it is almost impossible to solve any design mistakes. Therefore, verification and validation are essential for the development of every satellite, and must be conducted on the ground beforehand. In particular, Apsara-1 will be launched when solar activity is at its peak, resulting in a shorter orbital lifetime than usual. It is crucial to know the expected lifetime for effective operation planning, and to ensure that it meets with international space debris mitigation guidelines which state that all spacecraft in low Earth orbit (LEO) must deorbit within 25 years after operation has ended [3]. Therefore, to facilitate satellite design and operation, it is important to create a space environment / orbit simulator.

There are numerous existing orbital simulation software tools, such as AGI's Satellite Tool Kit (STK) [4], a.i.solutions' FreeFlyer [5], NASA's General Mission Analysis Tool (GMAT) [6] and CNES's Semi-analytic Tool for End of Life Analysis (STELA) [7]. One of the advantages of using available tools is that they can be utilized immediately and have many features. However, one disadvantage is the cost: some tools require a paid license to use their advanced features, and the limited features of free packages may not be adequate for some applications. By contrast, developing an in-house simulator is a valuable learning exercise, and also allows a satellite development team to extend it with additional functions and improve the performance in line with the progress of a satellite project.

In this paper, we present the development of an orbital simulator that can be used to estimate the lifespan of a CubeSat in LEO. In addition, we verify and validate its performance against a publicly available orbital simulator called STELA.

2. THEORETICAL BACKGROUND

2.1 Orbital reference frame

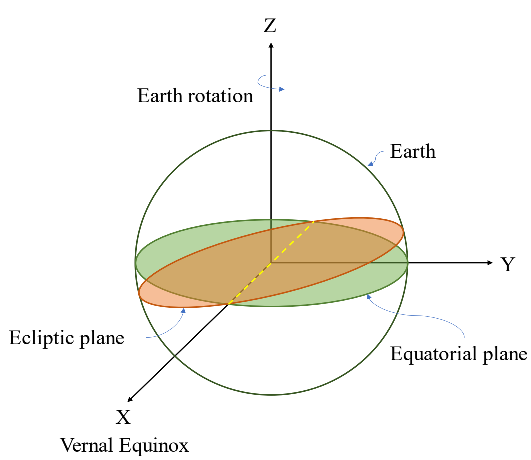

Fig. 1. ECI reference frame

In order to develop an orbital simulator, the first step is to define suitable reference frames for the analysis. The Earth Centered Inertial (ECI) frame is the choice for this paper. This reference frame is illustrated in Fig. 1.

The ECI frame is centered on the Earth and fixed with respect to inertial space. It has its origin at the center of the Earth. The X axis is pointed towards the sun at the vernal equinox, passing through the line where the equatorial plane and ecliptic plane intersect. The Z axis is directed along the Earth's axis of rotation, pointing North. The Y axis completes the coordinate system according to the right-hand rule.

2.2 State representation

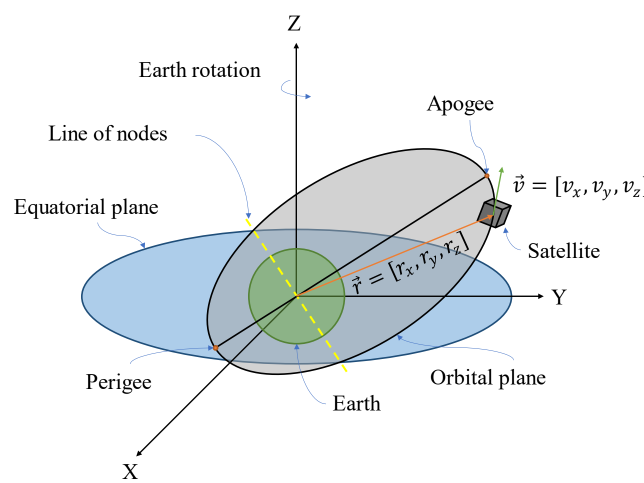

Fig. 2. Visualization of orbital state vector in the ECI frame

There are several ways to represent orbital information, and two are especially popular. Both methods completely specify the orbit, and can also be used to provide a complete set of initial conditions for predicting the orbit at future times by solving an initial value problem class of differential equations. The two representations are: the state vector, which consists of the object's position and velocity vectors; and classical orbital elements, which are also known as Keplerian orbital elements and consist of six orbital parameters. For a two-body problem with negligible perturbations, equations based on orbital elements are convenient to quickly and easily estimate an object's state at any arbitrary time. On the other hand, an analysis model based on the state vector is simpler and more useful to account for significant, arbitrary, time varying forces such as drag, which is expected to have a major impact on the orbit of the target body in this LEO study. Therefore, in this paper the state vector is used for all further orbital analysis. Note that an object's state vector can be used to compute its classical or Keplerian orbital elements and vice versa.

Fig. 2 shows the position and velocity of a satellite in orbit around the Earth in the ECI frame. The following vectors indicate the position, *r*, velocity, *v*, and acceleration, *a*, of a satellite in the ECI frame, expressed in Cartesian coordinates,

$$\vec{r}(t) = x(t)\vec{\imath} + y(t)\vec{\jmath} + z(t)\vec{k} \quad \text{(Eq. 1)}$$

$$\vec{v}(t) = \dot{x}(t)\vec{\imath} + \dot{y}(t)\vec{\jmath} + \dot{z}(t)\vec{k} \quad \text{(Eq. 2)}$$

$$\vec{a}(t) = \ddot{x}(t)\vec{\imath} + \ddot{y}(t)\vec{\jmath} + \ddot{z}(t)\vec{k} \quad \text{(Eq. 3)}$$

where the vectors $\vec{\imath}$, $\vec{\jmath}$ and $\vec{k}$ are the unit basis vectors in the X, Y and Z directions. For conciseness, matrix notation can be used as below,

$$\vec{r} = [x, y, z], \qquad \vec{v} = [\dot{x},\ \dot{y},\ \dot{z}], \qquad \vec{a} = [\ddot{x}, \ddot{y},\ \ddot{z}]$$

In astrodynamics, the orbital state vector is a formulate of position vector $\vec{r}$ and velocity vector $\vec{v}$ that together determine the trajectory of the orbiting body in space. The orbital state vector is usually denoted by $\boldsymbol{X}$,

$$\boldsymbol{X} = [\vec{r}\ \vec{v}] = [x\ y\ z\ \dot{x}\ \dot{y}\ \dot{z}]$$

2.3 Force models

To describe the motion of a satellite, it is required to define what forces are acting on it. Apsara-1 is planned to be deployed from the International Space Station (ISS) by releasing it into space with a specific initial velocity and position. Apsara-1 will then orbit under the influence of Earth's gravity, and the Earth can be represented as a point mass as a first approximation. However, there are several perturbation forces that also act on the CubeSat. The J2 perturbation is a significant perturbing gravitational force created by the oblateness of the Earth, that causes the satellite's orbit to undergo precession about the Earth's rotation axis. Atmospheric drag is another perturbation, and it is responsible for the satellite's altitude decay. In this paper, we only take these two perturbations into account, since they are the dominant effects acting on small satellites at ISS altitude. The motion of the satellite due to these forces can be described using the following equation of motion,

$$\vec{F}_{pm} + \vec{F}_{J_2} + \vec{F}_{atm} = m_{sat}\vec{a} \quad \text{(Eq. 4)}$$

where m_{sat} is the mass of the satellite [kg], F_{pm} is the gravitational force due to Earth's point mass acting on the satellite [N], F_{J_2} is the gravitational perturbation due to the oblateness of the Earth [N], and F_{atm} is the atmospheric drag force [N].

2.3.1 Gravity point mass

To derive the equation of orbital motion, Newton's second law and the universal law of gravitational are the starting points. Then, the acceleration can be written as below

$$\ddot{\vec{r}}_{pm} = \frac{G(m_{earth} + m_{sat})}{r^2}\frac{\vec{r}}{r}$$

where G is the gravitational constant [m^3·kg^{-1}·s^{-2}], and m_{earth} is the mass of the Earth [kg]. Since the mass of the satellite is much smaller than the mass of the Earth, it can be neglected, and so

$$\ddot{\vec{r}}_{pm} = \frac{Gm_{earth}}{r^2}\frac{\vec{r}}{r}$$

or

$$\vec{a}_{pm} = \ddot{\vec{r}}_{pm} = -\frac{\mu}{r^3}\vec{r} \quad \text{(Eq. 5)}$$

where $\vec{a}_{pm}$ is the acceleration of the satellite due to Earth's point mass [m/s], and Gm_{earth} is denoted by μ which is the Earth's standard gravitational parameter [m^3 s^{-2}].

2.3.2 J2 Perturbation

As the shape of the Earth is not perfectly spherical, Earth's gravity is also not spherical, and this affects the satellite's motion. The gravitational potential representing the true shape of the Earth can be described by a spherical harmonic function. The J_2 perturbation captures the majority of Earth's non-sphericity, and truncating the spherical expansion at this term is satisfactory for this study. From the aspherical potential of J_2, U_{J_2}, expressed in the ECI frame,

$$U_{J_2} = -\frac{(\mu J_2 R_{\oplus}^2)}{2r^3}\left(1 - 3\left(\frac{z}{r}\right)^2\right) \quad \text{(Eq. 6)}$$

the gradient operation is used to determine the resulting acceleration acting on the central body,

$$\nabla U_{J_2} = \vec{a}_{J_2}$$

$$a_{J_2x} = -\frac{3\mu J_2 R_{\oplus}^2}{2r^5}\left(1 - 5\left(\frac{z}{r}\right)^2 x\right) \quad \text{(Eq. 7)}$$

$$a_{J_2y} = -\frac{3\mu J_2 R_{\oplus}^2}{2r^5}\left(1 - 5\left(\frac{z}{r}\right)^2 y\right) \quad \text{(Eq. 8)}$$

$$a_{J_2z} = -\frac{3\mu J_2 R_{\oplus}^2}{2r^5}\left(3 - 5\left(\frac{z}{r}\right)^2 z\right) \quad \text{(Eq. 9)}$$

$$\vec{a}_{J_2} = \left[a_{J_2x}, a_{J_2y}, a_{J_2z}\right] \quad \text{(Eq. 10)}$$

where $\vec{a}_{J_2}$ is the acceleration due to J_2 [m/s], J_2 is the second zonal geopotential harmonic coefficient [dimensionless], $R_{\oplus}$ is the radius of the Earth [m], and $\vec{a}_{J_2x}$, $\vec{a}_{J_2y}$ and $\vec{a}_{J_2z}$ are the components of $\vec{a}_{J_2}$ along the axes of the ECI frame.

2.3.3 Atmospheric density model

In LEO, at ISS altitude, the effect of atmospheric drag on a satellite cannot be neglected. Air drag causes the satellite's trajectory to decay down to re-entry. The acceleration due to atmospheric drag $\vec{a}_{atm}$ can be determined by the equation

$$\vec{a}_{atm} = -\frac{1}{2} C_D \frac{A}{m_{sat}} \rho v_{rel}^2 \left(\frac{\vec{v}_{rel}}{|\vec{v}_{rel}|}\right) \qquad \text{(Eq. 11)}$$

where C_D is the drag coefficient [dimensionless], A is the satellite's effective cross-sectional area [m^2], ρ is the atmospheric density [kg/m^3], and v_{rel} is the speed of the atmosphere relative to the satellite [m/s]. As a first approximation, it can be assumed that the atmosphere rotates at the same rate as Earth. Thus, the relative velocity can be found by

$$\vec{v}_{rel} = \vec{v} - \vec{\omega}_{\oplus} \times \vec{r}$$

$$\vec{v}_{rel} = [\dot{x} + \vec{\omega}_{\oplus}\ y, \dot{y} + \vec{\omega}_{\oplus}\ x, \dot{z}] \qquad \text{(Eq. 12)}$$

where $\omega_{\oplus}$ is the angular velocity of the Earth about its axis [rad/s].

In the above two equations, the atmospheric density is the most challenging parameter to be determined. There are numerous density models that have been developed to determine the density. In particular, the NRLMSISE-00 model has become a standard for orbital dynamics modelling in LEO. This model also takes solar activity into account, which is important given the high solar activity during the expected mission period of Apsara-1.

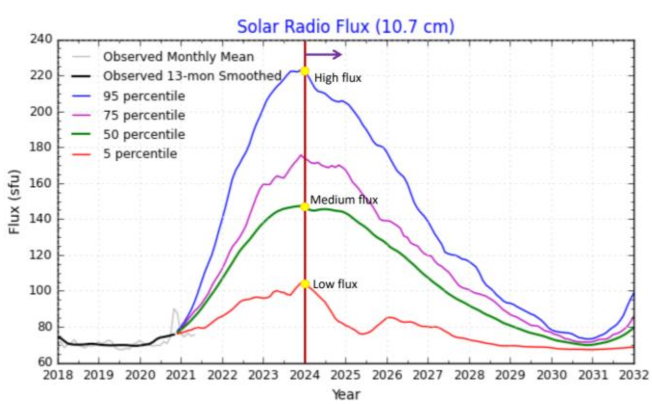

Fig. 3 F10.7 solar activity prediction (adapted from [8])

Solar activity predictions describe the impact of space weather on the atmosphere, and allow its effect on the expected orbital lifetime of a satellite to be determined. The F10.7 (solar radio flux) and Ap (geomagnetic planetary activity) indices can be used as inputs to an upper atmosphere (also called thermosphere) density model to calculate the atmospheric density. Higher solar activity leads to higher density at a given altitude, and hence a shorter mission lifetime. Solar activity predictions over around the next 10 years are shown in Fig. 3. The solar cycle has a period of 11 years. It can be seen that the target launch date of Apsara-1, January 2024, coincides with the solar maximum.

For simplicity, in this paper an online density calculator [19] was used to calculate atmospheric density data from initial altitude to altitude at the end of life of Apsara-1. The calculator's graphic interface can be seen in Fig. 4. While the atmospheric density varies with respect to altitude, it also varies as a function of latitude and longitude at the same particular altitude. To simplify, it can be assumed that the density at various latitudes and longitudes at a particular altitude has a constant value that is equal to the average at that altitude. The output of this method is a table of density data that consists of atmospheric density as a function of altitude. This table is used in the in-house simulator for the purpose of Apsara1's lifetime simulation.

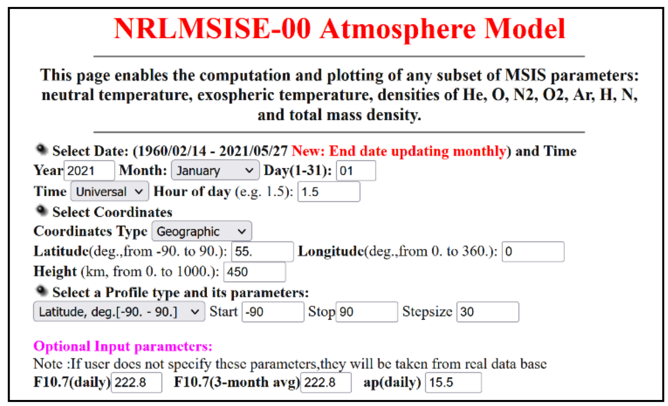

Fig. 4. Atmospheric density online calculator (adapted from Rastaetter [9]

To obtain an atmospheric density table to be used in simulation, the steps are as follows. Step 1 is to find in Fig. 3 the F10.7 and Ap indices during time of the mission for three solar flux conditions: low, medium, and high. In step 2, on the online atmosphere calculator website, the following tasks should be performed: set "coordinate type" to geographic, set "longitude" to 0 and leave the "latitude" at default, set "altitude" to 450km, and set "profile type" to latitude with limits -90 and 90 and step 30 degrees. For "optional input parameters", the values found in step 1 should be input for one solar flux case (e.g., high), and the density values should be generated and stored. In step 3, step 2 should be repeated for longitude equal to 30 degrees then 60 and so on until 330 degrees, while other settings are unchanged. By the end of step 3, a table of the Earth's atmospheric density at 450 km altitude is created, with resolution 30 degrees in longitude and latitude, and the average density at that altitude can be calculated to reduce the dimension of the table to a single value. In other words, the density no longer varies with respect to longitude and latitude, but only changes as a function of altitude. In step 4, steps 2 and 3 are repeated for altitudes of 400km, 350km and so on until 150km. By the end of this step, a density table ranging from 150km to 450km with 50km resolution for a given solar condition is obtained, covering the expected orbital altitude range during the Apsara-1 mission. In step 5, steps 2, 3 and 4 are repeated for the other two solar flux conditions. Finally, the result is a density table consisting of the mean density averaged over longitude and latitude at altitudes from 450km to 150km in 50km increments. Values are stored for three different solar activity conditions: low, medium, and high.

3. ORBITAL SIMULATION DEVELOPMENT

3.1 Orbital simulator architecture

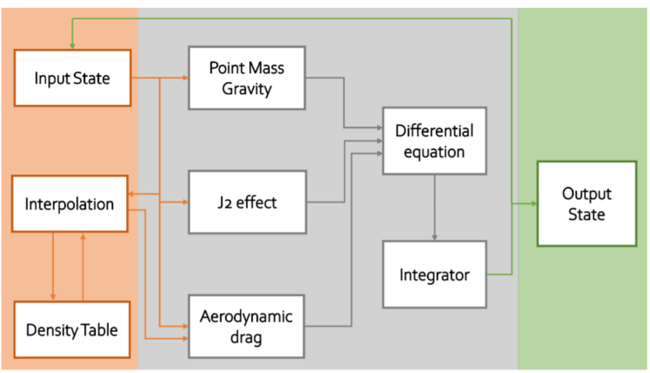

Fig. 5. Orbital simulator architecture

There are three main orbital propagation techniques that can be used to propagate a satellite along its orbit from an initial position until the end of life: analytical technique, semi-analytical technique, and numerical technique. In this paper, the numerical technique is used. It requires an orbital dynamic equation, which is a differential equation, solved by numerical integration with a specified initial condition. The in-house orbital simulator was developed step by step, module by module using the MATLAB environment with the equation introduced in section 2. Yet, the simulator has no user interface and which is a feature to be further developed in the future.

Fig. 5 shows the architecture of the orbital simulator and how each component interacts with the others. Overall, the architecture is separated into three areas, distinguished by color. The orange part refers to the input, the gray part represents the computation part, and the green part represents the output. The main part in this architecture is the “computation part” that include differential equation and numerical integration. The 56differential equation is formed by combination of three dominant force models include the point mass gravity, the j2 effect and the aerodynamic drag (atmospheric drag). The computation part requires input from “input part” that contains initial condition such as state vector and atmospheric density. The “output part” is generated from the “computation part” that contain the propagated state vectors. The mathematical detail is shown in Fig. 6.

A description of Fig. 5 is presented in the above three tables. Table 1 describes the blocks of the input area that contain the input state, density table, and interpolation. Even if the interpolation block performs computation to calculate the density at a specific altitude, it is considered to be in the input area since it provides input data for the computation area. The block “input state” refers to the current state vector to input for propagating the next state. The other input parameters such as the satellite properties, Earth’s properties, etc., that are unchanged throughout the simulation time are declared into relevant blocks of the computation area. For instance, the aerodynamic drag calculation requires a drag coefficient of the satellite, thus it is declared there.

Table 1 Input area of the architecture

Block	Description
Input state	The initial position and velocity vector of the satellite.
Density table	A table that consists of atmospheric density data as a function of satellite altitude in the range from 150km to 450km.
Interpolation	A numerical technique to calculate the density based on the closest two values in the table.

Table 2 describes the blocks of the computation area consisting of a differential equation made up of the three force models for point mass gravity, the J2 effect and atmospheric drag, and an integrator that is used to solve the differential equation.

Table 2 Computation area of the architecture

Block	Description
Differential equation	An equation that represents orbital motion of the satellite, including the three dominant forces.
Point mass gravity	Gravity force from the two-body problem.
J2 effect	Perturbation due to oblateness of the Earth.
Atmospheric drag	Drag force due to friction of the satellite with the atmosphere.
Integrator	The numerical scheme used to solve the differential equation numerically.

Table 3 provides the description of the output area which has only one block which is the output state. After each step of numerical integration, the output state becomes the input state to the subsequent iteration.

Table 3 Output area of the architecture

Block	Description
Output state	The propagated orbital state vector of the satellite.

3.2 Mathematical architecture of the orbital simulator

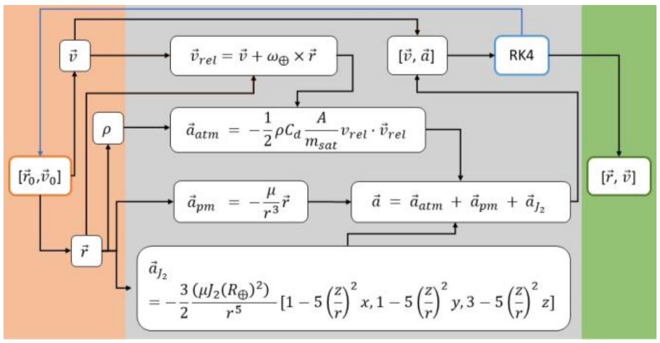

Fig. 6. Mathematical architecture of the orbital simulator

Fig. 6 displays the mathematical architecture of the orbital simulator. It has the input area, computation area and output area. The input area contains the initial orbital state vector and a look up table of density. The density is a function of altitude, thus based on the look up table, a linear interpolation method is used to determine the density value at a specific satellite position. The computation area is where the differential equation is formed and computed by numerical integration. Based on performance and convenience, the fourth order Runge Kutta method (RK4) is chosen for numerical integration. The output area is the propagated orbital state vector.

4. RESULTS AND DISCUSSION

Now that the simulator has been developed, it is used to generate orbital decay trajectories of the 1U CubeSat Apsara-1, to estimate the mission lifetime.

4.1 Problem setting

This section lists the parameters that are needed for the simulation. It is assumed that Apsara-1 will be deployed from the ISS on 1 January 2024. The altitude of the satellite when deployed from the ISS will be in the range between 380km and 420km. For this simulation, the average value of 400km is chosen.

Table 4 provides the necessary orbital parameters that are used to determine the initial orbital state vector. Inclination defines the orbital plane orientation with respect to the equatorial plane, and altitude defines the height from the surface of the Earth to the satellite. Eccentricity is assumed to be zero due to the near circular orbit of the ISS, and the argument of perigee and right ascension of ascending node (which define the orientation of the orbital plane and of the orbital skewness) and the true anomaly (which defines the position of a satellite along its orbit) are assumed to be zero for simplicity.

Table 4 Initial orbital parameters configuration

Parameter	Name	Value	Unit
i	Inclination	51.6	°
h	Altitude	400	km
e	Eccentricity	0	-
ω	Argument of perigee	0	°
Ω	Right ascension of ascending node	0	°
ν	True anomaly	0	°

Table 5 Properties of Apsara-1

Parameter	Name	Value	Unit
m	Mass	1.33	kg
A	Area	0.015	m^2
C_D	Drag coefficient	2	-

Table 5 contains the properties of Apsara-1, namely mass, frontal cross section area and drag coefficient. Assumed constants are presented in Table 6. These constant values are also required for the simulation.

Table 6 Other constant value

Symbol	Name	Value	Unit
μ	Earth standard gravitational parameter	398600.441×10^9	m^3/s^2
J_2	J2 coefficient	$1.08262668 \times 10^{-3}$	-
$R_{\oplus}$	Earth's equatorial radius	6,378,137	m

4.2 Results and validation

The decay trajectories are simulated with three cases: one nominal case and two off-nominal cases with different predicted solar flux activity. The detailed description of each case is shown in Table 7 below.

The orbital decay trajectories produced by the orbital simulator are shown in Fig. 7, for the three previously mentioned solar activity conditions, giving a total of three possible trajectories. The 'low', 'med' and 'high' solar activity values are respectively the 5, 50 and 95 percentiles for the solar flux predictions in 2024-25. For the nominal condition, Apsara-1 will be in orbit for around 144 days. The off-nominal case is divided into 2 cases. With high solar flux, Apsara-1 would stay in orbit for about 72 days. With low solar flux, Apsara-1 would stay in orbit for around 255 days.

Table 7 Assumed solar flux condition

Solar flux condition	Description
Medium (50% solar flux)	Average solar flux for the mission time.
High (95% solar flux)	High predicted solar flux for the mission time.
Low (5% solar flux)	Low predicted solar flux for the mission time.

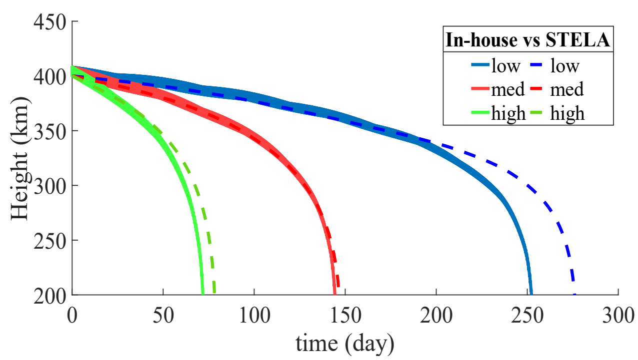

Fig. 7. Orbital decay trajectories result produced by STELA compared to in-house orbital simulator

STELA is the tool that is selected to verify the newly developed in-house simulator. It is a freely available tool that has been designed by CNES (the French Space Agency). To verify the result produced by the in-house simulator, STELA was used to produce the decay trajectory with the same setup. Fig. 7 also shows the result produced by STELA. Based on the result from STELA, the shortest flight is about 78 days while the longest flight time is about 275 days, and the normal flight duration is around 146 days.

4.3 Discussion

As the result shows, Apsara-1's lifespan has been estimated: in the nominal case, it will remain in orbit for approximately five months. However, to ensure that Apsara-1 meets with mission objectives, we need to consider the worst scenario where Apsara-1 has the shortest flight time, three months. Compared to the general lifetime of a CubeSat in LEO, Apsara-1's lifetime is very short. Based on the prediction of the lifetime of Apsara-1, the operation team shall create operation planning that reflects this short flight duration and also conduct ground operation training beforehand to ensure that Apsara-1 can finish all its planned experiments to meet with the defined mission objectives. In addition, Apsara-1 also meets international space debris regulation guidelines because in the worst (i.e., longest) case it will deorbit within approximately nine months with the minimum amount of solar activity.

By comparing the results from both simulators, the in-house simulator produces good results with a difference of overall approximately 8% for the decay duration compared to the publicly available tool. One reason for the small variation between both results is the fact that both simulators are built differently. In particular, for the in-house orbital simulator, the density data obtained for decay estimation is simplified by making the assumption that the density at various latitudes and longitudes at a particular altitude has a constant value which is equal to the average at that altitude. On the other hand, STELA uses a more accurate atmospheric density model, which accounts for the changes in density with latitude and longitude.

As mention earlier, the most beneficial of the in-house development of the simulator is that it is editable. While the publicly available tools such as STELA has limited fixed feature that do not allow the user to make any change as they needed. The in-house simulator can be further improved and expend to support the next the satellite development phases.

To upgrade and improve the in-house simulator, a more realistic density atmosphere model should be used. Thus, the decay estimation will be more realistic. Moreover, instead of using only the J2 perturbation, additional higher degree and order geopotential terms could be included. This will improve the orbit determination to be more accurate, although it is not expected to affect the performance of the decay estimation. Apart from generating orbit trajectories for lifetime estimation, the simulator could easily be extended to become a supporting tool for the design and verification of other satellite subsystems. For example, during the concept design phase of Apsara-1, the simulator has been used to help design the attitude controller of Apsara-1. Moreover, it also provides data on solar illumination and eclipse conditions necessary to conduct thermal analysis for the thermal subsystem and power budget analysis for the electrical power system.

In addition, the development of the in-house orbital simulator will also be useful in the future in DCLab. As the Apsara-1 project progresses from one phase to the next, the simulator will be needed to be used along the way. As this tool is developed with modularity and reusability concept, it is easier to add or remove more sophisticated physical effects and simulation capabilities as the mission needs change and the students' abilities mature. In terms of a learning point of view, the present simulator is hoped to become a tool for DCLab members to learn and practice since it is required to be continuously developed, with more new features added for the next phase of Apsara-1 development. After the end of the Apsara-1 project, the in-house simulator will remain a tool that can support future satellite or rocket projects of DCLab.

5. CONCLUSIONS

In this paper, we developed an orbital simulator as a supporting tool for mission design and analysis in the early phase of a LEO CubeSat project. The simulator was used to generate the orbital decay trajectories of the 1U CubeSat Apsara-1 and predict its lifetime in orbit. To verify the simulator, the results were compared with those obtained from the publicly available orbit simulation software STELA. The orbital lifetimes obtained using the two methods differed by around 8%, which is acceptable for usage during the early design phase of the Apsara-1 CubeSat. The contribution of the simulator is the lifetime perdition that facilitate the development team in mission design and also for space debris mitigation demonstration. Moreover, the simulator also facilitates for subsystem design such as electrical power and thermal subsystem which required orbital information as inputs. As the simulator need further improvement along the satellite development phases, for the next step of development, we plan to increase the accuracy by upgrading the perturbation model to a higher precision model. For instance, the perturbation due to the non-spherical shape of the Earth can be replaced by a full spherical harmonic function. In addition, we plan to turn the orbital simulator into a sophisticated environment/satellite simulator that is capable of being used for the analysis of various satellite subsystems in a single platform. Lastly, we also plan to make it more user-friendly by developing a graphic user interface.

ACKNOWLEDGMENTS

The authors would like to thank the Dynamics and Control Laboratory and the Department of Aeronautics and Astronautics of the University of Tokyo for creating the opportunity for Cambodian students to be able to learn about space technology and participate in the Apsara-1 project.

REFERENCES

[1] Berthet, M. (October 2020). Student-Led Policy and Technical Capacity Building Program: The Road to Cambodia's First CubeSat. *71st International Astronautical Congress, Paper IAC20-E1.7.x57649*.

[2] Berthet, M. (2021). International Collaboration Towards Cambodia’s First Small Satellite Education Program: Lessons Learnt. *Small Satellite Conference*, 11.

[3] United Nations Office for Outer Space Affairs. (2010). *Space Debris Mitigation Guidelines of the Committee on the Peaceful Uses of Outer Space.* United Nations publication

[4] Analytical Graphics, i. (2021). *Systems Tool Kit (STK)*. Retrieved on 17 October 2021 from https://www.agi.com/products/stk

[5] a.i.solution. (n.d.). *FreeFlyer Astrodynamics software*. Retrieved on 17 October 2021 from https://ai-solutions.com/freeflyer-astrodynamic-software/

[6] National Aeronautics and Space Administration. (2018). *design and integration tools*. Retrieved on 17 October 2021 from https://software.nasa.gov/software/GSC-18094-1

[7] The French Space Agency. (n.d.). *Semi-analytic Tool for End of Life Analysis*. Retrieved on 17 October 2021 from https://logiciels.cnes.fr/en/content/stela

[8] NASA. (2021). *Solar Cycle Progression and Forecast*. Retrieved on 17 October 2021 from https://www.nasa.gov/msfcsolar

[9] Rastaetter, L. (2021). *NRLMSISE-00 Atmosphere Model*. Retrieved on 17 October 2021 from https://ccmc.gsfc.nasa.gov/modelweb/models/nrlmsise00.php