Satellite Orbit Simulation Physics

Writer	Francisco Borlongan

Gravity and Circular Orbit

Newton's law of gravitation gives the attractive force between an object with mass m_1 and another object with mass m_2

$$F_g = \frac{Gm_1m_2}{r^2}$$

where G is the universal gravitational constant and r is the separation between the objects. A satellite in circular orbit needs a centripetal force to stay in orbit

$$F_c = ma_c = m\frac{v^2}{r}$$

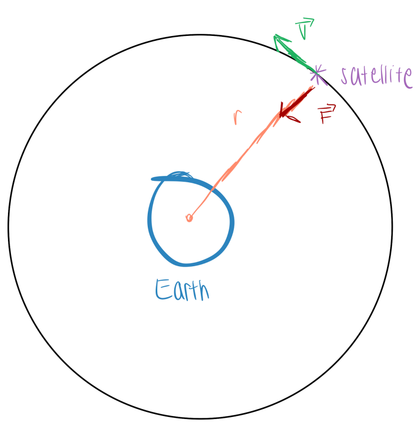
where v is the orbital speed and a_c is the centripetal acceleration. Consider a satellite of mass *m* in a circular orbit about Earth at a distance *r* from the center of Earth. It has centripetal acceleration directed toward the center of the Earth. Earth's gravity is the only force acting, so Newton's second law gives

$$\frac{GmM_e}{r^2} = ma_c = \frac{mv_{orbit}^2}{r}$$

, which is a satellite of mass *m* orbiting at radius *r* from the center of Earth. Acceleration then is given by the equation

$$a_c = \frac{GM_e}{r^2}\ or\ a_c = \frac{F_c}{m}$$

Gravitational force supplies the centripetal acceleration. The following illustrates a broader idea.



The satellite is pulled downward by Earth's gravitational pull $\vec{F}$, all while the force ties together with velocity $\vec{v}$ to maintain it on a course of orbit. We can solve for the speed of the orbit, noting that m cancels, to get the orbital speed.

$$v_{orbit} = \sqrt{\frac{GM_E}{r}}$$

Verlet Algorithm

To find the position and velocity of a satellite in orbit, Newton's equations of motion can be integrated using the Verlet algorithm. Through this, we find

$$x_{n+1} = x_n + v_n \Delta t \ + \frac{1}{2} a_n (\Delta t)^2$$
$$v_{n+1} = v_n + \frac{1}{2}(a_{n+1} + a_n)\Delta t$$

which represent the position Verlet and velocity Verlet respectively. At each time step:

1. Compute the acceleration a_n from the current position x_n
2. Update the position x_{n+1} using the position Verlet formula
3. Recompute the acceleration a_{n+1} at the new position
4. Update the velocity v_{n+1} using the velocity Verlet formula

Control Actuators

Magnetorquer

Magnetic torquing is a type of control that uses a magnetic field. They can change the magnetic strength of their electromagnetic coils, instead of relying on the constant strength of a permanent magnet. The electromagnetic coil's field is controlled by switching current flow through the coils. The magnetic dipole generated by the magnetorquer is expressed by the formula

$$m = nIA$$

where n is the number of turns of the wire, I is the current provided, and A is the vector area of the coil. The dipole interacts with the magnetic field generating a torque

$$\tau = m \times B$$

where m is the magnetic dipole vector, B is the magnetic field vector (for a spacecraft it is the Earth's magnetic field vector), and τ is the generated torque vector. Magnetorquers are useful for initial acquisition maneuvers, like detumbling and nadir-pointing.

Reaction Wheel

Reaction wheels are the most common actuator for active control. They're highly reactive and offer continuous feedback control. Reaction wheels create torque on the spacecraft by creating equal but opposite torques on the reaction wheels, which are flywheels on motors. For three axes of torque, three wheels are necessary. In an implementation, reaction wheels are usually operated around some nominal spin rate to avoid stiction effects.

Combination of Magnetorquers and Reaction Wheels

Spacecraft that stay in space typically have one internal torque source and external torque source. For smaller spacecraft, this combination is typically magnetorquers and reaction wheels. CubeSat ADCS packages use a combination of magnetorquers and reaction wheels.

Nadir Pointing

Nadir pointing is an attitude control method in which a satellite directs its face of the body containing onboard instruments such as a camera to the surface of the Earth. At any given moment in orbit, the satellite will adjust such that in every frame, the nadir point is maintained. Therefore, nadir pointing requires continuous computation of the desired satellite orientation and angular velocity. The orientation is crucial for Earth observation, access to solar charge, and efficient downlinks.

Building a Reference Frame

To compute the desired orientation, the vectors r and v are defined, where the former is a position vector indicating the satellite position relative to the Earth, and the latter is a velocity vector indicating the satellite velocity relative to Earth. Then, the reference frame is constructed using the two vectors.

$$\hat{z} = -\frac{r}{||r||}\hat{h} = \frac{r \times v}{||r \times v||}\hat{x} = \frac{\hat{h} \times \hat{z}}{||\hat{h} \times \hat{z}||} = \hat{z} \times \hat{x}$$

Desired Orientation (Quaternion)

The desired attitude is computed from the LVLH basis vectors.

$$R_{desired} = [\hat{x}\ \hat{y}\ \hat{z}]\ q_{desired}$$

In code, cast $R_{desired}$ to a quaternion then normalize it. This is for interpolation and error computation.

Attitude Error

The orientation error is computed using quaternion algebra.

$$q_{err} = q_{desired}q_{current}^{-1}(e_{axis}, \theta) = \mathrm{AxisAngle}(q_{err})$$

The error quaternion is converted to an axis-angle pair, where e_{axis} is the rotation axis and θ is the magnitude of the error.

Desired Angular Velocity

To track orbital motion, the CubeSat must rotate at the orbital angular rate.

$$\omega_{orb} = \sqrt{\frac{\mu}{r^3}} \omega_{des} = R_{desired} \begin{bmatrix} 0 \\ w_{orb} \\ 0 \end{bmatrix}$$

Transforming the angular velocity vector into the body frame ensures correct rotational tracking. The angular velocity error can also then be computed.

$$e_\omega = \omega_{actual} - \omega_{desired}$$

The difference between actual and desired angular velocity gives the rate error for derivative control.

Control Law (PD Controller)

A proportional-derivative (PD) controller generates the required control torque.

$$K_p = I(\omega_n^2), K_d = I(2\zeta\omega_n)\tau = -(K_p\theta e_{axis} + K_d e_\omega)$$

In code, for large attitude errors, derivative control ignores $\omega_{desired}$ to prioritize stopping rotation.

Actuator Saturation

Torque is limited to reaction wheel capability.

$$\tau_{final} = \mathrm{clamp}(\tau, -0.002, 0.002)$$

References

Milankovitch orbital cycles and their role in Earth’s climate. (n.d.). NASA Science. https://science.nasa.gov/science-research/earth-science/milankovitch-orbital-cycles-and-their-role-in-earths-climate/

Nikolic, B. (n.d.). Numerical solution of ODEs. University of Delaware. https://www.physics.udel.edu/~bnikolic/teaching/phys660/numerical_ode/node5.html

Nadir pointing. (n.d.). Spirepedia. https://spire.com/spirepedia/nadir-pointing/

Physics Classroom. (n.d.). Mathematics of satellite motion. https://www.physicsclassroom.com/class/circles/lesson-4/mathematics-of-satellite-motion

ResearchGate. (2017). Nadir pointing and sun pointing mode of the satellite [Figure]. https://www.researchgate.net/figure/a-Nadir-pointing-and-b-sun-pointing-mode-of-the-satellite_fig3_318677736

Satellite orbits and energy. (n.d.). Pressbooks BCCampus. https://pressbooks.bccampus.ca/universityphysicssandbox/chapter/satellite-orbits-and-energy/

Satellite orbits and energy. (n.d.). LibreTexts Physics. https://phys.libretexts.org/Workbench/PH_245_Textbook_V2/05%3A_Module_4_-_Special_Applications_of_Classical_Mechanics/5.01%3A_Objective_4.a./5.1.04%3A_Satellite_Orbits_and_Energy

University of Hawaii. (n.d.). Energy in satellite motion. Pressbooks. https://pressbooks-dev.oer.hawaii.edu/epet302/chapter/7-7/

What are Smallsats and CubeSats? (n.d.). NASA. https://www.nasa.gov/what-are-smallsats-and-cubesats/