Model-Based Design of the Attitude Determination and Control System on Killick-1 GNSS-R CubeSat

Heyang Zhang, Pengchen Xu, Bennett Newhook, Thomas Fitzpatrick
Memorial University of Newfoundland
Faculty of Engineering and Applied Science
230 Elizabeth Ave, St. John's, Canada A1C 5S7

***Abstract*—Being part of the Canadian CubeSat Project (CCP) hosted by Canadian Space Agency (CSA), Memorial University of Newfoundland is due to be launching Killick-1 GNSS-Reflectometry CubeSat to detect sea ice using Delayed Doppler Mapping technique. This paper presents the model-based design of the Attitude Determination and Control System (ADCS) algorithm on Killick-1. At core of ADCS, a nonlinear controller computes torque command to eliminate error between current and desired attitude, while a parallel-running momentum unloading controller prevents actuator saturation. Three-axis pointing and time-varying target tracking can be achieved jointly. A SIMULINK model is presented to validate system functionality, stability and tracking performances.**

***Keywords*—CubeSat, Attitude, Control, Simulink**

I. Background

A. Introduction

The Attitude Determination and Control System (ADCS) stabilizes and controls the orientation of the satellite in space to fulfill mission requirements, such as pointing camera to the earth, or pointing solar panels to the sun.

The ADCS estimates the satellite states using sensory information, and applies computed control commands to a set of actuators to achieve high accuracy and robustness. Typical actuators onboard CubeSats operating in Low-Earth Orbit are reaction wheels (RWs) and magnetic torquers (MTs), where RWs provide rapid, fine control by means of momentum exchange, and MTs provide external torques for unloading the excess angular momentum stored in the RW-spacecraft system.

B. Requirement

The Killick-1 CubeSat mission profile requires the satellite to maintain its angular position relative to earth's surface within ±3° (arbitrary rotation axis), and to have an instantaneous angular velocity less than ±0.25 °/s (arbitrary rotation axis) when collecting data. The strict requirement implies that an active ADCS system with three-axis stabilization and time-varying target tracking ability must be designed.

C. Attitude Control Principle

Attitude is defined by the rotation parameter, such as Quaternion or Direction Cosine Matrix (DCM), from inertial coordinate frame to satellite body-fixed coordinate frame. Attitude control is the act of driving the error between current rotation parameter and target rotation parameter to zero, where the target rotation parameter is usually being calculated by guidance portion of the flight control system.

For example, in nadir pointing mode the target attitude to be tracked is the rotation parameter from inertial coordinate system to a reference coordinate system that have one of its axes aligned to the center of the earth. The controller tracks this target in real time to achieve nadir pointing. Same principle can be applied to sun tracking or ground target tracking modes.

II. Development

A. Variable and Coordinate Frame Definition

All bolded and italicized letters refer to a column matrix, which can represent a general vector in 3-dimensional Euclidean space, or a compact form of grouping variables. An italicized letter enclosed by a square bracket denotes a matrix.

Quaternion is used as attitude parameterization due to its singularity free nature for expressing the rotation between any orientations:

$$\boldsymbol{q} = [q_0\ q_1\ q_2\ q_3]^T = [\beta\ \boldsymbol{\varepsilon}]^T. \tag{1}$$

Where $\beta = q_0$ is the scalar part, and $\boldsymbol{\varepsilon} = [q_1\ q_2\ q_3]^T$ is the vectoral part.

The vector cross-product operation is defined as the product between the skew-symmetric matrix of the first vector and the column form of the second vector:

$$\boldsymbol{a}^{\times}\boldsymbol{b} = [\tilde{a}]\boldsymbol{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}\begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}. \tag{2}$$

Which mathematically equals to $\boldsymbol{a} \times \boldsymbol{b}$.

The Earth Centered Inertial (ECI) coordinate frame is a nonrotating frame that have its origin coincides with the center of the earth's ellipsoid; The x and y axes lie on the equatorial plane, and the x-axis points to the Vernier Equinox. Although ECI frame is not truly inertial due to earth's nutation, in the case of low-earth orbiting satellites the ECI frame is sufficient to be treated as inertial.

The Earth Centered Earth Fixed (ECEF) coordinate frame is fixed onto earth's body and rotates with the earth. The x-axis intersects the Greenwich Meridian, and the y-axis lies within the

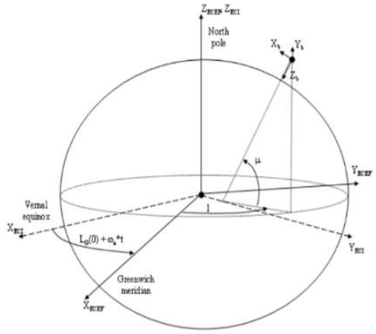

Figure 1. Illustration of ECI and ECEF coordinate frame

equatorial plane with z-axis pointing to geographic north pole. Fig. 1 illustrates ECI and ECEF frame [1].

The origin of the Body Fixed coordinate frame coincides with the Center of Mass (CoM) of the satellite; the axis direction can be assigned for convenience.

B. *Dynamic Model with Actuators*

To derive the equation of motion (EoM) of the combined RW-spacecraft system with i reaction wheels ($i \geq 3$), few key terms are needed to be defined and derived.

The orientation matrix $[G_s]$ contains the unit vector of the spin direction of each reaction wheel rotors, expressed in body frame and following the right-hand rule [2]:

$$[G_s] = [\hat{\boldsymbol{g}}_{s1} \quad \cdots \quad \hat{\boldsymbol{g}}_{si}] \tag{3}$$

The four reaction wheels on Killick-1 are being arranged in pyramidal configuration for redundancy.

The augmented spacecraft inertia tensor is being calculated by subtracting the projection of the reaction wheel rotor inertia in the spin direction from the total spacecraft inertia tensor [2]:

$$[I_{aug}] = [I_s] - \sum_1^i \left(J_{si} \hat{\boldsymbol{g}}_{si} \hat{\boldsymbol{g}}_{si}{}^T\right) \tag{4}$$

Where $[I_s]$ is the total spacecraft inertia tensor.

The angular momentum vector of the reaction wheels $\boldsymbol{h}_{RW}$ seen in the inertial frame equals to the rotor inertia J_{si} multiplied by the rotational speed. The rotation speed can be defined by the addition of the body angular velocity $\boldsymbol{\omega}$ and the relative angular velocity of the reaction wheel with respect to the satellite body Ω_i [3].

$$\boldsymbol{h}_{RW} = \begin{bmatrix} \vdots \\ J_{si}(\hat{\boldsymbol{g}}_{si}{}^T \boldsymbol{\omega} + \Omega_i) \\ \vdots \end{bmatrix} \tag{5}$$

The RW motor torque can be found by taking the derivative of (5), with $\boldsymbol{u}_{RW}$ being the torque computed by controller [3]:

$$\boldsymbol{u}_{RW} = \begin{bmatrix} \vdots \\ J_{si} \hat{\boldsymbol{g}}_{si}{}^T \dot{\boldsymbol{\omega}} + J_{si} \dot{\Omega} \\ \vdots \end{bmatrix} \tag{6}$$

The final EoM of the combined RW-spacecraft system is [3]:

$$[I_{aug}]\dot{\boldsymbol{\omega}} = -[\tilde{\omega}][I_{aug}]\boldsymbol{\omega} - [\tilde{\omega}][G_s]\boldsymbol{h}_{RW} - [G_s]\boldsymbol{u}_{RW} + \boldsymbol{L} \tag{7}$$

Where $\boldsymbol{L}$ is the sum of external torques expressed in body frame, which typically include torques induced by aerodynamic drag and gravity gradient. Notice the rows in $\boldsymbol{u}_{RW}$ and $\boldsymbol{h}_{RW}$ equals to the number of RWs.

Simulink is used to numerically integrate (7) to solve for body angular velocity. Body angular velocity is then substituted into the Quaternion Kinematic Differential Equation [4] below to solve for attitude.

$$\dot{\boldsymbol{q}} = \tfrac{1}{2} \boldsymbol{q} \otimes \boldsymbol{\omega} \tag{8}$$

Where $\otimes$ denotes quaternion multiplication.

C. *Attitude and Angular Velocity Error*

The satellite attitude is defined as the quaternion from ECI frame to body frame, denoted by $\boldsymbol{q}_{eci2b}$. The time-varying reference attitude to be tracked is the quaternion between ECI and any reference frame, denoted by $\boldsymbol{q}_{eci2r}$ [5].

The attitude and body rate error are being calculated [5]:

$$\boldsymbol{q}_{r2b} = \boldsymbol{q}_{eci2b} \otimes \boldsymbol{q}_{eci2r}{}^{-1} \tag{9}$$

$${}_{b/r}^{\;b}\boldsymbol{\delta\omega} = {}_{b/n}^{\;b}\boldsymbol{\omega} - {}_{r/n}^{\;b}\boldsymbol{\omega} \tag{10}$$

${}_{b/n}^{\;b}\boldsymbol{\omega}$ denotes "the angular velocity of body frame with respect to inertial frame, expressed in body frame"; similar notation can also be interpreted in the same way. To avoid verbose expression, ${}_{b/n}^{\;b}\boldsymbol{\omega}$ will be simplified to $\boldsymbol{\omega}$, and ${}_{r/n}^{\;b}\boldsymbol{\omega}$ will be simplified to $\boldsymbol{\omega_r}$, and ${}_{b/r}^{\;b}\boldsymbol{\delta\omega}$ will be expressed as $\boldsymbol{\delta\omega}$ [5]

the reference angular velocity is often given in reference frame coordinate, which one need to multiply by the DCM [BR] to convert it into body frame coordinate. The final angular velocity error is [5]:

$${}_{b/r}^{\;b}\boldsymbol{\delta\omega} = {}_{b/n}^{\;b}\boldsymbol{\omega} - [BR]\,{}_{r/n}^{\;r}\boldsymbol{\omega} \tag{11}$$

D. *Lyapunov's Direct Method*

Lyapunov's Direct Method is used to derive torque-level control law for the nonlinear equation in (7).

Consider the Lyapunov Function and its Lyapunov Rate [5]:

$$V(\boldsymbol{q}, \boldsymbol{\delta\omega}) = \tfrac{1}{2} \boldsymbol{\delta\omega}^T [I_{aug}] \boldsymbol{\delta\omega} + [K_P](\boldsymbol{q} - \hat{\boldsymbol{q}})^T (\boldsymbol{q} - \hat{\boldsymbol{q}}) \tag{12}$$

$$\dot{V}(\boldsymbol{q}, \boldsymbol{\delta\omega}) = \boldsymbol{\delta\omega}^T \left([I_{aug}]\, {}^{B}\frac{d}{dt}(\boldsymbol{\delta\omega}) + [K_P]\boldsymbol{\varepsilon} \right) \tag{13}$$

Where $\hat{\boldsymbol{q}} = [1\ 0\ 0\ 0]^T$ is the zero rotation of quaternion, and $[K_P] = diag(K_{P1}\ K_{P2}\ K_{P3})$ is the proportional gain matrix.

Equating Lyapunov Rate with a known negative definite function yield:

$$\boldsymbol{\delta\omega}^T \left([I_{aug}]\, {}^B\frac{d}{dt}(\boldsymbol{\delta\omega}) + [K_P]\boldsymbol{\varepsilon} \right) = -\boldsymbol{\delta\omega}^T [K_V] \boldsymbol{\delta\omega} \quad (14)$$

Where $[K_V] = diag(K_{V1}\ K_{V2}\ K_{V3})$ is the velocity gain matrix.

Expand ${}^B\frac{d}{dt}(\boldsymbol{\delta\omega})$ gives [5]:

$$[I_{aug}](\dot{\boldsymbol{\omega}} - \dot{\boldsymbol{\omega}}_{\boldsymbol{r}} + [\tilde{\boldsymbol{\omega}}]\boldsymbol{\omega}_{\boldsymbol{r}}) + [K_P]\boldsymbol{\varepsilon} + [K_V](\boldsymbol{\delta\omega}) = 0 \quad (15)$$

Eq. 15 can be used to derive torque-level control law that guarantees the system to be globally asymptotically stable.

$[K_P]$ and $[K_V]$ are all diagonal matrix that contains the gains corresponding to the principle moment of inertia axis.

E. *Torque Level Control Law*

Substitute (7) into (15) and solve for motor torque gives [5]:

$$-\boldsymbol{u}_{\boldsymbol{RW}}[G_s] = [\tilde{\omega}][I_{aug}]\boldsymbol{\omega} + [\tilde{\omega}][G_s]\boldsymbol{h}_{\boldsymbol{RW}} + [I_{aug}]\dot{\boldsymbol{\omega}}_{\boldsymbol{r}} - [\tilde{\omega}][I_{aug}]\boldsymbol{\omega}_{\boldsymbol{r}} - [K_P]\boldsymbol{\varepsilon} - [K_V]\boldsymbol{\delta\omega} - \boldsymbol{L} \quad (16)$$

$\boldsymbol{u}_{\boldsymbol{RW}}[G_s]$ is the resulting torque applied to the spacecraft expressed in body frame coordinates.

An integration term is added to address the unmodelled disturbance torques. Due to the positive definiteness of the inertia tensor, such addition will not impact the stability.

The final control law is [6]:

$$-\boldsymbol{u}_{\boldsymbol{RW}}[G_s] = [\tilde{\omega}][I_{aug}]\boldsymbol{\omega} + [\tilde{\omega}][G_s]\boldsymbol{h}_{\boldsymbol{RW}} + [I_{aug}]\dot{\boldsymbol{\omega}}_{\boldsymbol{r}} - [\tilde{\omega}][I_{aug}]\boldsymbol{\omega}_{\boldsymbol{r}} - [K_P]\boldsymbol{\varepsilon} - [K_V]\boldsymbol{\delta\omega} - [K_I]\int_0^t \boldsymbol{\varepsilon} dt - \boldsymbol{L} \quad (17)$$

Where $[K_{\boldsymbol{I}}] = diag(K_{I1}\ K_{I2}\ K_{I3})$ is the integral gain matrix.

F. *Reaction Wheel Desaturation*

Unmodelled external torques will cause excess angular momentum to buildup, causing the RW to saturate. The excess momentum is being unloaded by MTs, which provides external torque in opposite direction to unload the excess angular momentum, thus bringing the wheel speed down.

First the de-spin torque based on the wheel speed feedback is obtained [6]:

$$\boldsymbol{u}_S^* = k(J_s\boldsymbol{\Omega} - J_s\boldsymbol{\Omega}_B) \quad (18)$$

Where $\boldsymbol{\Omega}_B$ is the desired rotational speed bias, and k is the proportional gain. k is being replaced by a full PID controller in simulation environment.

The de-spin torque is subtracted from $\boldsymbol{u}_{RW}$ to yield the torque command specific to reaction wheels [6]:

$$\boldsymbol{u}_{RW\_cmd} = \boldsymbol{u}_{RW} - \boldsymbol{u}_S^* \quad (19)$$

Finally, the torque command to MTs is being calculated [6]:

$$\boldsymbol{u}_{MT\_cmd} = -\boldsymbol{u}_S^* \quad (20)$$

This torque will be provided by the interaction between magnetic torquers and geomagnetic field.

III. SIMULATION

A. *Model-Based Simulation*

A Simulink model along with a virtual reality 3D animation are built to validate the functionality of the algorithm and to provide visual representation.

A complete CubeSat simulator incorporates the environment block, which includes earth's gravity, magnetic field, and atmosphere, the guidance computer, which provides the target attitude to be tracked for different pointing modes, such as nadir pointing, sun tracking and ground target tracking, and a navigation computer, which estimates spacecraft states based on sensor fusion algorithm.

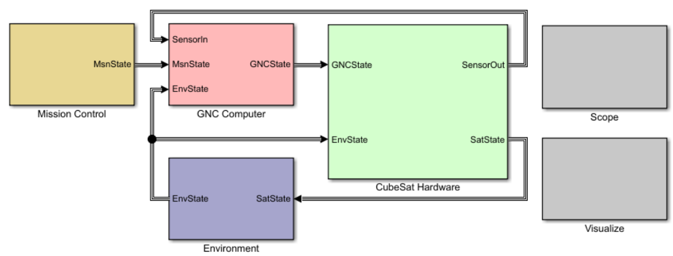

Figure 2. System level block diagram modelling of Killick-1 CubeSat

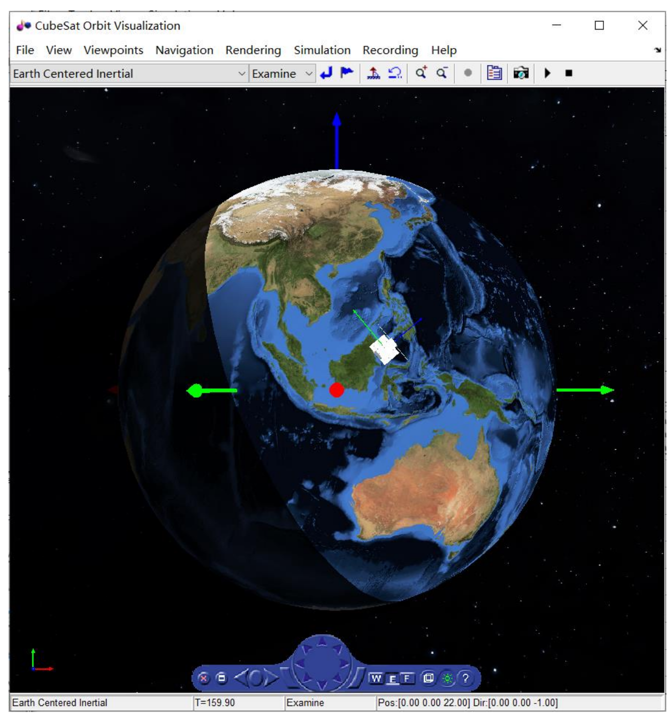

Figure 3. Virtual reality world built in VRML language to aid in visualization

B. Simulation Result

Basic parameters of Killick-1 CubeSat as well as the initial conditions are shown below:

Total inertia: $[I_s] = \begin{bmatrix} 2113 & -7.7 & 4.6 \\ -7.7 & 1590 & 0.9 \\ 4.6 & 0.9 & 1219 \end{bmatrix} \times 10^{-5}$ kg • m^2

Reaction wheel rotor inertia: $J_{s1-4} = 3500 \times 10^{-9}$ kg • m2

Initial orbital position: $\boldsymbol{x_{0\_eci}} = [6771000\ 0\ 0]^T$ m

Initial orbital velocity: $\boldsymbol{v_{0\_eci}} = [0\ 6444\ 5444]^T$ m/s

Initial attitude: $\boldsymbol{q_{0\_eci2b}} = [0.831\ 0.411\ 0.373 - 0.027]^T$

Initial angular velocity: $\boldsymbol{\omega_o} = [0.1\ 0.1\ 0.1]^T$ rad/s

Desaturation torque limit: 1×10^{-3} Nm

Reaction wheel bias: $\boldsymbol{\Omega}_B = [100\ 100 - 100 - 100]^T$ rad/s

Simulation results of a series of maneuvers are shown below:

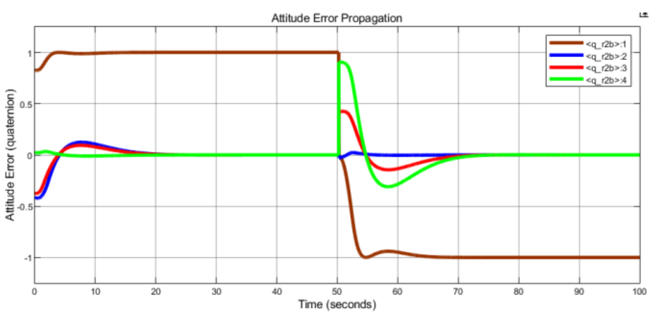

Figure 4. Attitude error propagation during stabilization and re-orientation

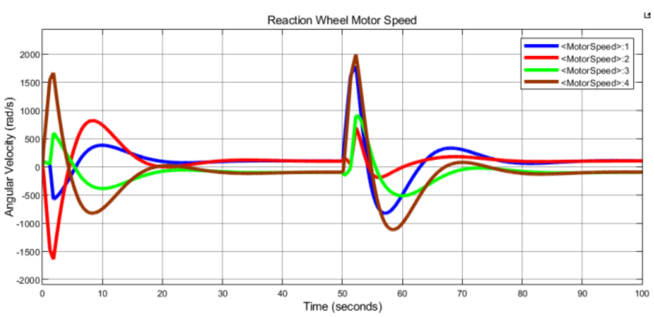

Figure 5. Reaction wheel motor speed evolvement

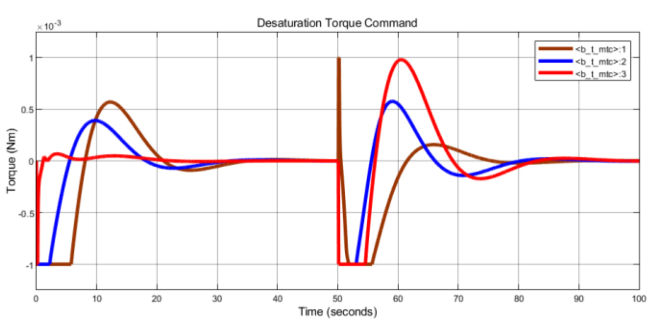

Figure 6. Computed desaturation torque command

Fig. 4 shows the attitude error during 100 second simulation, where a re-orientation command is given at 50 second after the satellite had stabilized itself from the initial tumble. It is clear that the attitude error quaternion converges to either [1 0 0 0] or [-1 0 0 0], which indicates no rotation.

Fig. 5 showed how the reaction wheel speed change during simulation. The first group of spikes is caused by the reaction wheel absorbing initial angular momentum, and the second group of spikes indicates momentum transfer between the wheel rotor and spacecraft body.

Fig. 5 also showed the reduction of reaction wheel speed towards a desired bias of $[100\ 100 - 100 - 100]^T$ rad/s after each maneuver. The de-saturation is being conducted by applying an external torque through magnetic torquers; the computed torque can be seed in fig. 6; notice that although the torques occasionally reach maximum limit, the de-saturation functionality has not been affected.

ACKNOWLEDGMENT

This project was supported by Memorial University of Newfoundland and C-CORE. We thank our colleagues from other subsystems of the Killick-1 CubeSat team who provided input and inspirations on the design aspect of ADCS.

We would like to especially show gratitude to our supervisor Dr. Oscar De Silva for sharing his of wisdom with us and for guiding us through the various stages of research. We also thanks Mr. Desmond Power for providing us insight and expertise in the aerospace industry that greatly assisted our design process as well as helped us avoid inefficient decisions.

We hereby express our deepest appreciation to everyone who had helped us on this project.

REFERENCES

[1] "Implement quaternion representation of six-degrees-of-freedom equations of motion in Earth-centered Earth-fixed (ECEF) coordinates - Simulink", Mathworks.com, 2019. [Online]. Available: https://www.mathworks.com/help/aeroblks/6dofecefquaternion.html. [Accessed: 27- Oct- 2019].

[2] H. Schaub, "Kinematics: Describing the Motions of Spacecraft", coursera.org, 2019.

[3] H. Schaub, "Kinetics: Studying Spacecraft Motion", coursera.org, 2019.

[4] A. Narayan, "How to Integrate Quaternions | Ashwin Narayan", Ashwin Narayan, 2019. [Online]. [Accessed: 27- Oct- 2019].

[5] H. Schaub, "Control of Nonlinear Spacecraft Attitude Motion", coursera.org, 2019.

[6] E. Hogan and H. Schaub, "Three-Axis Attitude Control Using Redundant Reaction Wheels with Continuous Momentum Dumping", Journal of Guidance, Control, and Dynamics, vol. 38, no. 10, pp. 1865-1871, 2015. Available: 10.2514/1.g000812.