2016

Analysis of Spacecraft Attitude Control

Shiqiao Zhu
Lehigh University

Analysis of Spacecraft Attitude Control

by

Shiqiao Zhu

A Thesis

Presented to the Graduate and Research Committee

of Lehigh University

in Candidacy for the Degree of

Master of Science

in

Mechanical Engineering and Mechanics

Lehigh University

May, 2016

The thesis is accepted and approved in partial fulfillment of requirements for the Master of Science.

Date Approved

_____

Dr. Terry Hart, Thesis Advisor

_____

Dr. D. Gary Harlow, Chairperson of Department

Acknowledgments

I want to thank my advisor, Dr. Terry Hart, for the time and effort he has expended in teaching me spacecraft dynamics control. His expertise and technical insight into spacecraft dynamics control has helped me form this research into something worthwhile. Over the last year, Dr. Terry Hart has enhanced my education through academics, research skills, and his own experiences as an engineer. He has been supportive and patient with me when aspects of the research became upsetting. For this, I'm grateful to my advisor, too. I'm also thankful to all the researchers, engineers involved in satellite design for the great effort placed for the success of the TT-1 nano-satellite mission.

Finally, I will give my special thanks to my parents for their continuous support, understanding and encouragement.

Contents

ABSTRACT

The thesis studies about repositioning the spacecraft to obtain a moving target problem. We model the spacecraft as a rigid body with axis torque controlling N axial symmetry wheels, and kinematic parameters by ADCS. Reference trajectory represented a virtual space generated by the same actual spacecraft.

Open loop reference posture, angular velocity and angular acceleration tracking order is constructed, making solar panels vector is perpendicular to the carrier tracking the movement of the sun. We developed a nonlinear feedback tracking control law, derived from the ADCS stability and control theory, the target tracking control torque. Asymptotic tracking controller make the main body frame, there are in the attitude and angular velocity error of initial reference movement. Spacecraft model, on the basis of resource manager in Tian Tuo-1 spacecraft, for demonstration in tracking the given target ADCS controller is effective.

Section 1

Introduction

Spacecraft formation flying, a developing technology has the potential ability to expand the future earth observation science missions. The development of small, low-cost spacecraft formation flying spacecraft has led several implementations of the idea of collaborative sensor. A form is defined as a set of coordinated motion of the vehicle, and it is very important of the vehicle's the relative position of each other.

The attitude tracking control is a kind of concept that as individual spacecraft control data collection, a target is selected, and the ship rigged so that the image sensor could track the place in the aim. The three main aspects of the attitude tracking control are fixed-point, angular velocity, and the desired attitude and tracking control. Airship orientation are needed to make it success in tracking control thereby the image sensor points being found directly, when it around its axis, to keep the sensor the same with the aim.

Creating the motor torque for fulfilling this maneuvering is implemented by repositioning the desired attitude and expectation of angular velocity of the control law. The spacecraft is called a three-axis stabilized spacecraft, when it is in three perpendicular axis of space vehicle control.[1] For three axis stability spacecraft, control torque also need to compensate for the environmental impact, such as air resistance, and gravity gradient torques to the direction of the spacecraft drift. These control torque can be generated from the outside, through the propeller, the internal momentum wheel, or by a combination of both. The purpose of this study is to develop the attitude tracking algorithm and control laws of three axis stabilized spacecraft to

view and track the target rotation of the earth. We first define for rigid spacecraft with momentum wheel system and the orbit model. Then we outlined the ideal posture, angular velocity, and the need to target and track the angular acceleration of the calculation of the open loop command. The target tracking trajectory by tracking the command of the sun also can make the solar cell array axis remain perpendicular to the direction of the sun, and sensor tracking target. Then, we use ADCS control theory of the development of the nonlinear feedback controller. The controller creates the necessary axial momentum wheel torque, and then makes the spacecraft body frame follow ideal open-loop trajectory by eliminating the initial tracking error tracking. Finally, the design results of ADCS are successful and suitable in using to the Tian Tuo-1. The results prove that feedback of control law drives the initial steps to zero of tracking error, and the controller can be linearized.

Section 2

Research Review

We present a brief overview of previous target tracking simulation and control investigations. 2.1 of this section discusses attitude and tracking control research that has been done for formation flying. The rest of the section deals with the tracking control problem of a single three-axis stabilized spacecraft. Our research is an extension of the control work presented in Sections 2.2–2.3.

2.1 Study of Formation Flying Spacecraft

Formation flight has not been widely studied in literature, but some simulation work in the past few years has been completed. Gramling etc.[2] studied the formation of two satellites. They discussed the vehicle navigation system for the earth observing (EO - 1)/ landsat -7 (L - 7) formation's relative navigation. Gramling etc. also outlined the EO 1 / L - 7 formation configuration and ONS is discussed how to applying Onboard Navigation System, together with the Global Positioning System, and let the formation control autonomous .They showed how the Onboard Navigation System tracking measurements from the spacecraft to spacecraft crosslink carrier signal by the Doppler frequency shift. The Onboard Navigation System's performance about tracking measurement type, quality, and frequency tracking was also investigated, and the formation of coplanar relative track geometry aspects was studied. The EO 1 / L - 7 mission's simulation results with the Doppler measurement errors and forming track and cross track errors over time are introduced.

With the extension of Gramling et al. work by Folta and Quinn[3] about the EO 1 / L - 7 mission concept. They studied using documents referred to in the formation of three types of tasks formation control.[4] Formation's drills present in using an autonomous closed-loop triaxial navigation and control algorithm for EO-1/L-7. Controller allows the spacecraft autonomous to perform complex triaxial exercises. Using the algorithm and the autonomous spacecraft simulation software simulates the ground track and maneuvers of inclination. The algorithm is considered to be in the ground track, orbital transfer control and dip Angle.

Other papers said this formation control algorithm was Ulbyshev's research[5] which involves feedback control by using a linear quadratic regulator for the position of the constellation keep. Then his derivation of the spacecraft equations' work was done by using formula of Clohessy - Wiltshire. He then deduced a linear-quadratic regulator controller for formation-keeping and put forward a double satellite constellation analysis solution. A 12-satellite constellation was simulated using the linear-quadratic regulator controller. The Control was not independent of the constellations, and needed the required data from mission control center. He stated that the controller minimized the along tracking of the displacements between orbital period displacements during the circular orbit and spacecraft.

2.2 Kinematics of Tracking Target

Tracking target includes pointing to an object, then moving to keep with the target for a given length of time. And pointing to an object needs specific posture in alignment with the target aimed at instrument aiming axis. Then the spacecraft must generate angular velocity to track a target when it is moving in its orbit. Track the simplest case is where the target is inertial fixed, and the spacecraft is originally static, then eventually the hull rates are zero.[6]

From the study, we can know that Hablani[7] developed the algorithm to generate reference trajectory as when the last angular velocity is not zero. And Hablani thought a rigid payload to connect to the spacecraft. For example, the sensor initially faces at the zenith direction. The payload direction's tracking orders were based on a 2-1-3 Euler Angle sequence. Hablani applied this sequence due to the fact that each per orbit rotation is naturally compensated by the pitch rotation on the pitch axis, which is not any coupling from the subsequent roller rotation.[8]

The spacecraft body triad $(\hat{b}_1, \hat{b}_2, \hat{b}_3)$ was initially aligned with the local-vertical-local-horizontal triad $(\hat{c}_1, \hat{c}_2, \hat{c}_3)$ before target acquisition by defining the initial boresight direction to be along the zenith direction. The desired boresight orientation was defined as the negative yaw axis $(\hat{b}_3)$. The other two spacecraft body axes, $\hat{b}_1$ and $\hat{b}_2$, were defined as the roll and pitch axis, respectively. Getting the target to the focal plane center of the instrument, the spacecraft is using a commanded pitch angle θ_{yc} to rotate about the $\hat{c}_2$ axis and then using a commanded roll angle θ_{xc} to rotate about the $\hat{c}_1$ axis acquiring the target. Because of not affect target tracking of a rotation about the sensor boresight axis, the '3' rotation was not needed. Just change the spacecraft's attitude in the other two body axes.

After that, the pitching roll Angle commands were showed by Hablani:

$$\theta_{yc} = \tan^{-1}\left[\Leftrightarrow(1\cdot\hat{c}_1)/\Leftrightarrow(1\cdot\hat{c}_3)\right]$$

The negative signs in the numerator and denominator of above equation were retained to determine the correct quadrant.

$$\theta_{xc} = \sin^{-1}\left[(1\cdot\hat{c}_2)/\square 1\square\right]$$

$$\theta_{zc} = 0$$

The yaw angle θ_{zc} and its rate $\dot{\theta}_{zc}$ were zero since the target vector was aligned with the boresight axis.

1 is from the spacecraft to the target's line-of-sight vector. Also, Hablani noted that the roll command would always be $\Leftrightarrow \pi / 2 \le \theta_{xc} \le \pi / 2$.

The position commands derived analogously the tracking angular velocity and acceleration commands. The details of these derivations could be search in reference 7 and 8. So the angular velocity commands:

$$\omega_{xc} = (\dot{1} \cdot \hat{b}_2) / \square 1 \square$$

$$\omega_{yc} = (\Leftrightarrow \dot{1} \cdot \hat{b}_1) / \square 1 \square$$

Making use of the fact that the spacecraft has three desired angular rates, Hablani got the yaw component of the angular velocity: the mean motion of the spacecraft in a circular orbit $\Leftrightarrow \omega_c \hat{c}_2$, the pitch rate command $\dot{\theta}_{yc} \hat{c}_2$, and the roll rate command $\dot{\theta}_{xc} \hat{c}_1$. The yaw component of the angular velocity became:

$$\omega_{zc} = \Leftrightarrow \omega_{yc} \tan \theta_{xc}$$

Then a second-order quantity even for a small roll angle was not found the inertial yaw rate ω_{zc}. Meanwhile, the commanded angular acceleration was proved to be equal to:

$$\dot{\omega}_{xc} = (\ddot{1} \cdot \hat{b}_2 \Leftrightarrow 2 \overset{\circ}{l} \omega_{xc}) / \square 1 \square + \omega_{yc} \omega_{zc}$$

$$\dot{\omega}_{yc} = (\ddot{1} \cdot \hat{b}_1 + 2 \overset{\circ}{l} \omega_{yc}) / \square 1 \square \Leftrightarrow \omega_{xc} \omega_{zc}$$

$$\dot{\omega}_{zc} = \Leftrightarrow \dot{\omega}_{yc} \tan \theta_{xc} \Leftrightarrow \omega_{xc} \omega_{yc} \sec^2 \theta_{xc}$$

$\mathring{l}$ is the rate of change in the line-of-sight vector in the body frame. Hablani also pointed out that the acceleration commands are useful for feedforward and/or determining the inertial resistance of the spacecraft when tracking a moving target.

It is similar for the Euler 2-1-3 sequence to the position, rate, and acceleration commands, and Hablani also derived the tracking kinematics for an Euler 1-2-3 sequence. Thus, through the roll-pitch sequence to acquire the target, the boresight axis was initially facing nadir and the spacecraft was rotated. The roll-pitch sequence was not found to be as effective for target tracking because of it being singular at 90˚ pitch angle, and acquiring a near-earth target causes the pitch angle to cross 90˚. The rate and acceleration commands for the yaw axis are also more complicated than the previous definitions because the orbital rate ω_o cannot be expressed as simply about the pitch axis $\hat{c}_2$ because it is for the pitch-roll sequence. Hablani stated that it was useful of the roll-pitch sequence, because when the pitch-roll commands are singular, the roll-pitch sequence is not. We could find the results of the roll-pitch derivation in reference 6 and 7.

In conclusion, Hablani[7] presented a approach for deriving the perfect alignment and target tracking commands. Hablani's approach's problem is that these are ideal tracking commands which are constructed only for an Euler angle 2-1-3 sequence when the sensor axis initially facing the zenith direction. There is no flexibility in choosing attitude parameters or where the boresight axis is initially facing. And the algorithm didn't contain sun tracking commands. So in section 4, for determining an attitude independent reference attitude from a rotation matrix, the thesis presents an algorithm that includes sun tracking commands. Greater flexibility in the choice of spacecraft attitude parameters is provided by constructing the attitude this way.

2.3 Attitude Control Theory Overview

The section supplies a review about how the craft can be controlled following actions. The Attitude determination and control system (ADCS) is often used to those what are not easy modeled as linear systems in nonlinear system. This section provides a description of the ADCS method that is used to determine the stability of power system.

2.3.1 ADCS Control Theory

It is very important for the design of the controller to have a stability of the control system. Rouse's method can be used to determine the stability for the linear and time invariant system[9], if the system is nonlinear, or linear but the time is changing, then the stability of its type's methods do not work. The ADCS, one direct method is frequently used for determining the stability of non-linear and/or time-varying systems.

Attitude determination and control system (ADCS) plays in a satellite orbiting an indispensable part which may greatly affect the performance of the satellite. The development of micro- and nano - satellite need an attitude control system, and this system is cheap, light weight, small volume, and low power consumption. Thus, the electromagnetic coil and bias momentum wheel has been used as the most popular actuators. Three-axis magnetic coil with spacing bias momentum wheel is the popular way to realize three-axis stability control. By the momentum wheel, which is nominally spinning at a fixed rate in the pitch direction of the satellite, this method forces the attitude stabilization in roll and yaw directions. Many micro and nano satellite in orbit today by this control.

2.3.2 Controllers for Spacecraft Target Tracking

From this section, the thesis will introduce the algorithms which is using on TT - 1 to estimate and control of its attitude with the sensors and actuators. Firstly, we will introduce the satellite's

attitude dynamics equations, and then we talk about attitude determination and control algorithm which are given respectively.

2.3.2.1. Attitude Dynamics Analysis

With fly wheels, the rigid spacecraft's attitude dynamics could be described as

$$J\dot{\omega}_{bi} = -\omega_{bi}^{x}(J\omega_{bi} + H) - \dot{H} + T_c + T$$

$$\omega_{bi} = \omega_{bo} + A_{bo}\omega_{oi}$$

ω_{bi} is the angular velocity expressed in the body reference frame,

J the inertia matrix

T_d external disturbance torque vector which imposed on the satellite

T_c the control torque vector which generated by actuators

H the momentum wheel's angular momentum

A_{bo} the transformation matrix from the orbit reference frame to the body reference frame

ω_{bi}^{x} the matrix of symmetrical skew

ω_{bo} the angular velocity relative to the orbit reference frame expressed in the body reference frame

ω_{oi} the orbital angular velocity expressed in the orbit reference frame

2.3.2.2. Attitude Determination Analysis

Euler angles or a direction cosine matrix of a quaternion usually expresses Satellite attitude. [2] And quaternion field is also a popular form to express their attitude. Researchers have proposed many methods to estimate the attitude and a variety of methods have been successfully used in the satellite mission. The TT - 1 task also have many attitude determination by different methods. After the separation released, the initial state was unknown. Therefore, attitude determination

used quaternion estimator (QUEST)[7] algorithm. However, once applying into a three axis stabilized control mode satellite, the method of the QUEST will be changed by Unscented Falman Filtering (UKF) algorithm because the accuracy of the determination by QUEST was not reasonable.

2.3.2.3. QUEST Algorithm Analysis

It is the most commonly used deterministic algorithms of the QUEST algorithm and the three axis attitude determination (TRIAD) algorithm for attitude determination. They are both solutions to the Wahaba's problem.[10] The QUEST and TRIAD algorithm have been both utilized successfully on the nano-satellite. For example, The Cute - 1.7 + APD II[11] through laboratory space systems in Tokyo institute of technology's development utilized the QUEST algorithm onboard to estimate the attitude. Because of the QUEST's low computation, TT - 1 adopted it to grab the initial attitude by magnetometer and sun sensors measuring. Note that once the two vectors are paralleling, at the end of the time interval of measurement will be preserved until get the unparallel carriers.

2.3.2.4. UKF Algorithm Analysis

UKF algorithm is used for determining attitude in three-axis stability phase. The State vector X- combines attitude quaternion q_{bo} and ω_{bi} , and the System state equation of kinematics and dynamics equations could be described as follows

$$\dot{X} = \begin{bmatrix} q_{bo} \otimes Q(\omega bo)/2 \\ J^{-1}(-\omega_{bi}^{\times}(J\omega_{bi}+H)-\dot{H}+T_c+T_d) \end{bmatrix} + \begin{bmatrix} I_{3\times3} & 0 \\ 0 & I_{3\times3} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

v_1 and v_2 represent Gaussian white noise sequences.

Also, the measurement model in sunlight is described as follows

$$Z_k = \begin{bmatrix} B_{bo} \\ S_{bo} \end{bmatrix} = \begin{bmatrix} A_{bo}(q_{bo})A_{oi}A_{ig}B_g \\ A_{bo}(q_{bo})A_{oi}S_i \end{bmatrix} + \begin{bmatrix} v_b \\ v_s \end{bmatrix}$$

The measurement model is set by magnetic field vector B_{bo} and sun vector S_{bo}.

B_g and S_i are the magnetic field and sun vector described in the geographic frame and the inertial frame respectively.

v_b and v_s represent Gaussian white noise sequences.

A_{oi} : is the transformation matrix from the inertial frame to the orbit frame.

A_{ig} : is the transformation matrix from the geographic frame to the inertial frame.

At this time, only the TAM can be obtained as the measurement vector. More details of UKF algorithm are presented in Refence 12.

Section 3

Equations of Spacecraft Motion

This section describes the motion equations to the spacecraft rigid body with momentum wheels. Euler's rotational equations of the dynamics presents in Section 3.1. These equations are given for the actual spacecraft and generating a reference trajectory's virtual space vehicles. The mathematical model of the spacecraft orbit and environmental impacts were described in Section 3.2.

3.1 Spacecraft System Model

In this section, a system model is presented for use in developing tracking control algorithms. The equations of motion presented here follow the notation developed in Hughes[13].We consider a rigid spacecraft P which is shown on Figure 3.1, with N rigid axisymmetric momentum wheels $R_i, i = 1, \cdots, N$

The wheels have an arbitrary, but fixed orientation with respect to the body.

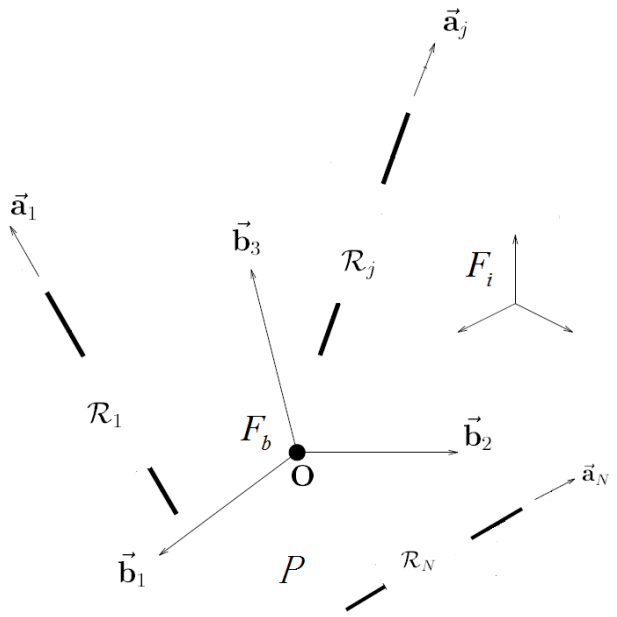

Figure 3.1: *N*-Momentum Wheels' Gyrostat Model

F_b denotes the body frame with the origin at the center of mass of the system $P + \sum_{i=1}^{N} R_i$

F_i denotes the inertial frame.

When a rigid spacecraft has one or more rigid axisymmetric wheels spinning about their axes of symmetry, then the system is known as a gyrostat.[14]

Make I on behalf of the moment of inertia of the system, including the momentum wheels, $I_s = diag\{i_{s1}, \cdot, I_{sN}\}$ denote the axial moments of inertia of the momentum wheels, and $3 \times N$ matrix $A = [a_1 \cdots a_N]$ is on behalf of the wheel axial a_j vectors. We do not assume that F_b is a principal frame. Therefore, I is not necessarily a diagonal matrix. As developed in reference 10 and 14, we use all vectors and tensors in a platform-fixed, non-principal frame, designated as the "pseudo-principal" frame. Without loss of generality, we let F_b represent the pseudo-principal frame, which is chosen so that the inertia-like matrix $J = I \Leftrightarrow AI_sA^T$ is diagonal. So the 3×1 system angular momentum vector in F_b is defined as

$$h_b = I\omega_b + AI_s\omega_s$$

The angular velocities of the body and the wheels are the ω_b and ω_s respectively. The $N \times 1$ matrix of the axial angular momenta of the wheels h_a, which is defined for the N-wheel gyrostat as

$$h_a = I_s A^T \omega_b + I_s \omega_s$$

The dynamics of the gyrostat are Euler's rotational equations of motion which come from the following equation from analytical dynamics[15]

$$\dot{v} = \overset{\circ}{v} + \omega^{\times} v$$

v represents any vector expressed in a frame with angular velocity ω.

"$\circ$" denotes differentiation of v about a moving coordinate frame.

Replacing v with h yields $\overset{\circ}{h}$, which is the rate of change of the angular momentum relative to F_b, and $\dot{h}$ which becomes the external torque acting on the system. Solving for the rate of change of the angular momentum in the body frame results in the form as

$$\dot{h}_b = \Leftrightarrow \omega_b^{\times} h_b + g_e$$
$$\dot{h}_a = g_a$$

g_e is the column vector of external torques that act on the body representing the $N \times 1$ matrix of the internal axial control torques applied by the platform to the momentum wheels.

$\omega_b^{\times}$ represents a skew-symmetric matrix form of a vector

$$\boldsymbol{\omega}_b = \begin{bmatrix} \omega_{1b} \\ \omega_{2b} \\ \omega_{3b} \end{bmatrix} \Leftrightarrow \quad \boldsymbol{\omega}_b^{\times} = \begin{bmatrix} 0 & \Leftrightarrow\omega_{3b} & \omega_{2b} \\ \omega_{3b} & 0 & \Leftrightarrow\omega_{1b} \\ \Leftrightarrow\omega_{2b} & \omega_{1b} & 0 \end{bmatrix}$$

$\Leftrightarrow \omega_b^{\times} h_b$ is the matrix equivalent to $\Leftrightarrow \omega_b \times h_b$. Using the definition for J, the angular momentum of the body could be written as

$$\begin{aligned} \mathbf{h}_b &= \mathbf{I}\boldsymbol{\omega}_b + \mathbf{A}(\mathbf{h}_a \Leftrightarrow \mathbf{I}_s \mathbf{A}^T \boldsymbol{\omega}_b) \\ &= \mathbf{I}\boldsymbol{\omega}_b + \mathbf{A}\mathbf{h}_a \Leftrightarrow \mathbf{A}\mathbf{I}_s \mathbf{A}^T \boldsymbol{\omega}_b \\ &= (\mathbf{J} + \mathbf{A}\mathbf{I}_s \mathbf{A}^T)\boldsymbol{\omega}_b + \mathbf{A}\mathbf{h}_a \Leftrightarrow \mathbf{A}\mathbf{I}_s \mathbf{A}^T \boldsymbol{\omega}_b \\ &= \mathbf{J}\boldsymbol{\omega}_b + \mathbf{A}\mathbf{h}_a \end{aligned}$$

So the body angular velocity can be expressed as:

$$\boldsymbol{\omega}_b = \mathbf{J}^{-1}(\mathbf{h}_b \Leftrightarrow \mathbf{A}\mathbf{h}_a)$$

Substituting ω_b into h_b yields the final expressions for the equations of motion:

$$\dot{h}_b = h_b^{\times} J^{-1}(h_b \Leftrightarrow A h_a) + g_e$$

$$\dot{h}_a = g_a$$

g_e are external torques, which are comprised of environmental torques, and possibly control torques by using thrusters or magnetic torque rods.

g_a are wheel torques, which are comprised of control torques applied by the motors, and possible friction torques.

In this thesis, assuming the gravity gradient torques, which are the only environmental torques present, and the motor torques which are the only control torques used to manuever the

spacecraft. The next part presents how $\dot{h}_b = h_b^{\times} J^{-1}(h_b \Leftrightarrow Ah_a) + g_e$ and $\dot{h}_a = g_a$ are used to define the fictitious spacecraft model and the reference control torque g_{ar}.

The desired trajectory to be tracked comes from the trajectory generated by a "virtual" spacecraft in a reference coordinate frame. Let F_r represent this reference frame which is fixed at the center of mass of this virtual spacecraft. In refence 11, the "virtual" spacecraft is assumed to be a rigid body. Here, we assume that the virtual spacecraft is a gyrostat, with the same properties as the real spacecraft.

Because the virtual spacecraft has the same inertial and wheel parameters as the real spacecraft, the reference frame dynamics are the same as in Section 3.1, except that the subscript b is replaced with r

$$\dot{h}_r = h_r^{x} J^{-1}(h_r \Leftrightarrow Ah_{ar}) + g_e$$

$$\dot{h}_{ar} = g_{ar}$$

We know that $h_r = J\omega_r + Ah_{ar}$. The external torque remains the same, but we need to solve for the virtual spacecraft's axial torque g_{ar} which is the torque that would generate the desired trajectory in the absence of initial condition errors.

The torque g_{ar} comes from first noting that $\dot{h}_r$ can also be expressed by differentiating h_r to get

$$\dot{h}_r = J\dot{\omega}_r + A\dot{h}_{ar} = J\dot{\omega}_r + Ag_{ar}$$

The above equations yield the following expression for the desired axial control torque

$$Ag_{ar} = h_r^{\times} J^{-1}(h_r \Leftrightarrow Ah_{ar}) + g_c \Leftrightarrow J\dot{\omega}_r$$

$\omega_r = J^{-1}(h_r \Leftrightarrow Ah_{ar})$ is the desired angular velocity for target tracking.

3.2 Spacecraft Orbit Model

For the sake of simplicity, we use two bodies motion equation to describe the circular orbit. Two bodies' equation is the only approximate orbital dynamics, and for the actual simulation should not be used because information. For example, the earth's flattening d disturbance from other planets, aerodynamic drag a solar radiation stress torque will be lost.

Two-body equation[12] is derived from Newton's law of universal gravitation, and contains two objects in the system, or a planet and a spacecraft. Assuming that the earth is our system's main body and the spacecraft is secondary. By defining the gravitational constant $\mu = Gm_{\oplus}$, and $m_{\oplus}$ is the mass of the earth, where the quality of the spacecraft is assuming to be spherical symmetry. As a result, both the earth and the spacecraft can be seen as a point of quality. In addition, there are no specific gravity forces along joint two institutions act as a no external or internal force on other systems. With these assumptions, the two bodies' movement orbit equation is defined as:

$$\ddot{r}_s = \frac{\mu}{\| r_s \|^3} r_s$$

r_s is the position vector from the center of the earth to the center of quality of the spacecraft in F_i.

Note that for near-earth spacecraft, the geocentric equatorial system serves as the inertial system, and for interplanetary spacecraft the heliocentric system is used as an inertial system. With changing the value of μ and defining the inertial system to be referenced from the planet's center, the two-body equation can also be used to describe satellite orbits around other planets.

In defines the track equation of motion, we now turn our attention to the external torque g_e.We assume that the only external torque at present in the equation $\dot{h}_b = h_b^{\times} J^{-1}(h_b \Leftrightarrow Ah_a) + g_e$ and $Ag_{ar} = h_r^{\times} J^{-1}(h_r \Leftrightarrow Ah_{ar}) + g_c \Leftrightarrow J\dot{\omega}_r$ is not affected by gravity gradient torque control. The

consequent variations in the specific gravitational force over a spacecraft body leads, in general, to a torque about the body mass center.[13] In contrast, if the gravitational field was uniform, then the center of mass would be at the same location as the center of gravity, and the gravitational torque about the quality of center will be zero. The following four assumptions[13] greatly simplify the gravity gradient torque expression.

(1) Only one object is mainly to be considered.

(2) Its owner has a spherically symmetric distribution of quality.

(3) Compared with the distance from the primary center of mass, a spacecraft is small.

(4) Spacecraft is a single institution.

Then with using these four assumptions, the gravity gradient torque over P, which is the center of mass shows:

$$g_e = \Leftrightarrow \mu \int_p \frac{r \times R}{R^3} dm$$

r is the vector from the spacecraft center of mass to dm

R is the vector from the center of the earth to dm

Expanding $g_e = \Leftrightarrow \mu \int_p \frac{r \times R}{R^3} dm$ and then applying the previous four assumptions, the vector form of the gravity gradient torque is found to be

$$g_e = 3\frac{\mu}{\| r_s \|^3} \hat{o}_3^{\times} I \hat{o}_3$$

$\hat{o}_3$ is a unit vector of the form

$$\hat{o}_3 = \Leftrightarrow \frac{r_s}{\| r_s \|}$$

And it is defined as a “nadir” vector which points to the location on the earth directly under the spacecraft.

The thesis has proposed equations of motion in using on circular orbit's model on spacecraft attitude dynamics. Dynamic modeling is for an actual spacecraft, and its virtual counterpart, then let *N* axisymmetric momentum wheel and attitude control. With the gravity gradient torque, orbit used movement of two bodies' equation as the only environment modeling. The two approximations are enough to produce the necessary space vehicle position and velocity vectors without the need of expensive calculation. Section 4 shows how in the aftermath of the orbital equation are used to determine the reference trajectory of virtual space vehicle.

Section 4

The Reference Trajectory

To track the moving target need a bearing body allowing some instrument fixed to the main body in the target point, and the required rotation and translation of the body to keep on moving targets. Because we are dealing with a given mobile spacecraft in orbit, we only need to track target by pointing and rotating motor.

In addition, we also ask the attitude to be constructed by the yaw control maneuver[12] to meet the requirements of solar power on the axis of sensor. We define pointing gesture and angular velocity which let the spacecraft body reference frame F_r track the target and the "reference" path of the sun. The reference trajectory is computed with open loop mode from the known location of space vehicle, speed, sensor visual axis, solar panels, and the sun vectors.

4.1 The Ideal Pointing Attitude Derivation

To point at a target needs a specific attitude making the position vector from the target to the spacecraft collinear with the instrument boresight as showed in Figure 4.1. The instrument axis can be any unit vector fixed in F_b. And it is defined to be the same in F_r. So the thesis defines the instrument, or visual axis to be along the "1" direction in F_b and F_r

$$a_b = a_r = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T$$

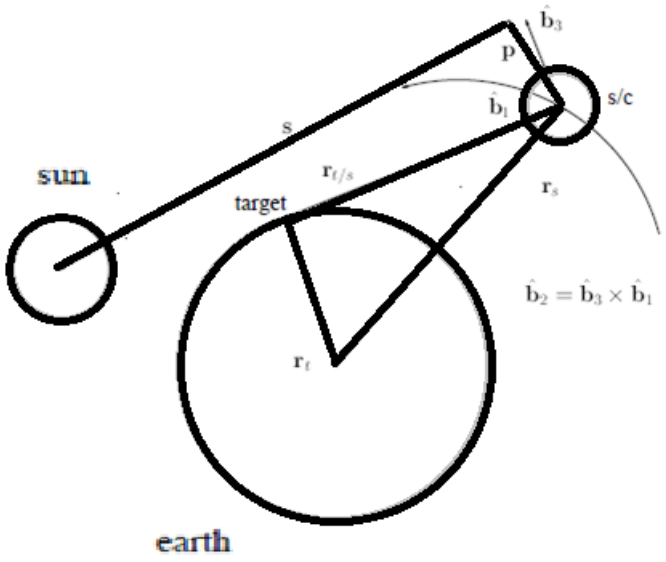

Figure 4.1: Reference attitude

In Figure 4.1, it could easily be seen that the target position vector relating to the spacecraft, $r_{t/s}$ can be written as:

$$r_{t/s} = r_t \Leftrightarrow r_s$$

r represents the position vector from the center of the earth to the spacecraft.

r_s is assumed known from orbital data

F_i expresses the target position vector as

$$r_{ti} = R_{\oplus}\left[\cos(\delta_t)\cos(\theta_{GST} + L_t)\cos(\delta_t)\sin(\theta_{GST} + L_t)\sin(\delta_t)\right]^T$$

δ_t and L_t are the latitude and longitude of the target

θ_{GST} is Greenwich sidereal time measured from a given epoch

$R_{\oplus}$ is the earth's radius.

Form Figure 4.1 shows the coordinate frames that we know these vectors. Also from the algorithm, the sun vector s is known when computes the sun unit vector in F_i from the true longitude and the obliquity of the ecliptic of date.

	F_b	F_i	F_o
$\mathbf{r}_t$	⇔	√	√
$\mathbf{r}_s$	⇔	√	√
a	√	√	√
p	√	⇔	⇔
s	⇔	√	√

Table 4.1: The vectors we know in deriving pointing attitude for use

Solar panels vector is the rotating shaft. Yaw steering operation aims to the alignment to the solar panels of the vector normal by rotating around its axis panel. This makes solar cells to generate a maximum of power for spacecraft. However, we ignore the rotation of the panels, and assume they are fixed about p to simplify the structure of the attitude of the yaw steering operation and reference. On the contrary, we rotate the craft in order to meeting the demand for electricity. We also ignore on the system any dynamic effects of flexible solar arrays. Panel carrier defines F_b and F_r as any unit vector.

Finding the required pointing attitude, we notice that r_{ti} can also be written as

$$r_{ti} = r_{xi} + R^{ir} D a_r$$

D represents the range from the spacecraft to the target.

R^{ir} yields the reference attitude, σ_r with respect to the inertial frame.

It is coordinate independent because the pointing attitude is in terms of a rotation matrix R^{ri}. Thus the attitude can be showed to use any set of attitude parameters. Then we need to determine R^{ri}. Because R^{oi} is known, the real problem is to determine R^{ro} while satisfying the yaw-steering condition:

$$S^T P = 0$$

From Table 4.1, we know that not every vector is known in all of the coordinate frames. If r_s and a_r were both known in F_r and F_o, this suggests using the TRIAD algorithm[17] to construct R^{ro}.

The TRIAD algorithm is used to determine an approximation of the rotation matrix from one coordinate system to another by constructing a rotation matrix from vectors in different coordinate frames. Make W_{1x} and W_{2x} on behalf of the column vectors in some coordinate system F_x, and let v_{1y} and v_{2y} denote the column vectors in some other coordinate frame F_y. Also for constructing the base vectors for F_x, the TRIAD algorithm becomes:

$$\hat{r}_{1x} = w_{1x} / \| w_{1x} \|$$

$$\hat{r}_{2x} = \frac{\hat{r}_{1x}^{\times} w_{2x}}{\| \hat{r}_{1x}^{\times} w_{2x} \|}$$

$$\hat{r}_{3x} = \hat{r}_{1x}^{\times} \hat{r}_{2x}$$

$$R^{rx} = \left[\hat{r}_{1x} \hat{r}_{2x} \hat{r}_{3x} \right] T$$

For F_y, the base unit vectors are

$$\hat{r}_{1y} = v_{1y} / \| v_{1y} \|$$

$$\hat{r}_{2y} = \frac{\hat{r}_{1y}^{\times} v_{2y}}{\| \hat{r}_{1y}^{\times} v_{2y} \|}$$

$$\hat{r}_{3y} = \hat{r}_{1y}^{\times}\hat{r}_{2y}$$

$$R^{yr} = \left[\hat{r}_{1y}\hat{r}_{2y}\hat{r}_{3y}\right]$$

r is a dummy variable on behalf of the base unit vectors in each frame. The rotation matrix R that transforms a vector from F_{x1} to F_y is constructed as

$$R^{yx} = R^{yr}R^{rx}$$

The TRIAD algorithm provides a conceptually simple way of constructing the desired rotation matrix when two vectors are known in two coordinate frames.

However, the TRIAD algorithm could not be used here to compute R^{ro}, or R^{ri} directly because r_s is not known in F_r. Any two vectors, known both in F_r and F_o, could also be used in the TRIAD algorithm to construct R^{ro}. Table 4.1 shows that there are no two vectors in these two coordinate frames. The TRIAD method can be used to construct each component leading to the rotation matrix R^{ro}. As a result, two intermediate frames F_a and F_c are defined below that allows us to use the TRIAD algorithm with the known vectors in Table 4.1. This leads to the construction of the rotation matrices R^{ao} and R^{rc}, which are used in determining R^{ro}.

When R^{ro} is known, the target pointing attitude of the virtual spacecraft is then given by the product of the following rotation matrices:

$$R^{ri} = R^{ro}R^{oi}$$

R^{oi} is constructed from the known orbit of the spacecraft and hence the inertial position and velocity vectors, r_{si} and v_{si} respectively. These vectors are used in the TRIAD algorithm to construct $R^{oi} = \left[\hat{o}_{1i}\hat{o}_{2i}\hat{o}_{3i}\right]^T$ as

$$\hat{\mathbf{o}}_{3i} = \frac{\Leftrightarrow \mathbf{r}_{si}}{\|\mathbf{r}_{si}\|}$$

$$\hat{\mathbf{o}}_{2i} = \frac{\mathbf{r}_{si}^{\times}\mathbf{v}_{si}}{\|\mathbf{r}_{si}^{\times}\mathbf{v}_{si}\|}$$

$$\hat{\mathbf{o}}_{1i} = \frac{\hat{\mathbf{o}}_{2i}^{\times}\hat{\mathbf{o}}_{3i}}{\|\hat{\mathbf{o}}_{2i}^{\times}\hat{\mathbf{o}}_{3i}\|}$$

$$R^{oi} = \left[\hat{o}_{1i}\hat{o}_{2i}\hat{o}_{3i}\right]^{T}$$

R^{oi} can also be used for elliptical orbits, and we use R^{oi} to rotate the sun vector and boresight vector into the orbital frame. So we calculate a in the inertial frame using

$$a_i = \frac{r_{t/si}}{\| r_{t/si} \|}$$

With using constructing R^{ao}, this expression is then rotated into F_o

We must define the two intermediate frames F_a and F_c before computing R^{ao} . The coordinate frame, F_a, is a "body carry" box in the center of the virtual spacecraft of the masses, which is relative to the center of the rotating track framework. Frame "a" differs from F because $\hat{a}_1$ points at the target, whereas $\hat{o}_1$ points in the direction of the spacecraft's velocity vector. Through the sun vector, s, and the boresight axis, a , frame "a" is related to orbital frame. We use them in the TRIAD method to construct R^{ao} because these two vectors are known in F_a . This feature is used to simplify the yaw-steering maneuver which relates F_a to F_c .

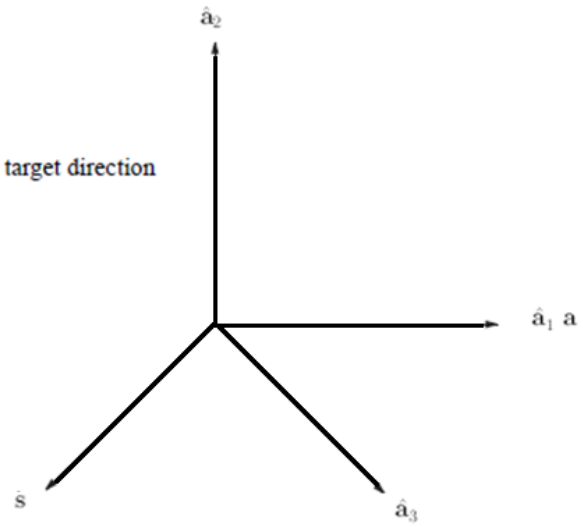

Figure 4.3: "a" the Intermediate Reference Frame

Centering at the virtual spacecraft's center of mass, the frame F_c is a body frame. Because F_c is fixed relative to F_r, it does not rotate with respect to F_r. Since a is defined to lie along $\hat{r}_1$, frame "c" differs from F_r only by a rotation about the boresight axis a. So these two frames are related by the known vectors p and a. Figure 4.4 illustrates how p and a are defined in F_a. Because in the case of the sun vector, p also lies in the 1-2 plane of F_c, which also simplifies the yaw-steering maneuver.

Along with F_b, F_i, and F_o, using F_a and F_c, allows us to use the TRIAD algorithm to construct the attitude in the following form:

$$R^{ro} = R^{rc} R^{ca} R^{ao}$$

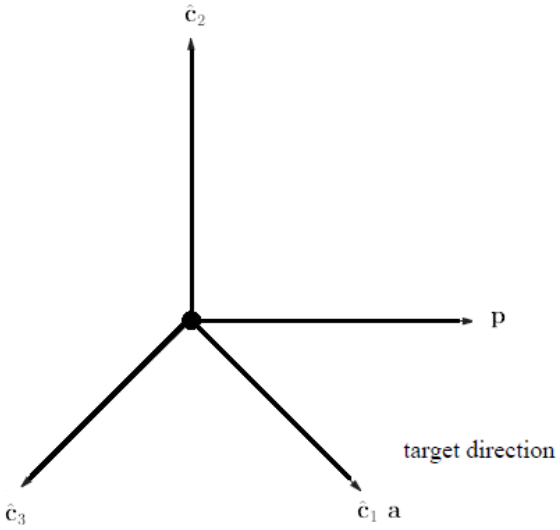

Figure 4.4: "c" the Intermediate Reference Frame

Knowing that the vectors a and p in F_r and s and a in F_o, we solve for R^{ri} with approaching the expression in $R^{ro} = R^{rc}R^{ca}R^{ao}$ from the right and the left side till we get R^{ca}. To begin on the right side, R^{ao} is built in using left equations below:

$$\hat{r}_{2x} = \frac{\hat{r}_{1x}^{\times} w_{2x}}{\| \hat{r}_{1x}^{\times} w_{2x} \|}$$
$$\hat{r}_{3x} = \hat{r}_{1x}^{\times} \hat{r}_{2x}$$
$$R^{rx} = \left[\hat{r}_{1x}\hat{r}_{2x}\hat{r}_{3x}\right]^T$$

$$\hat{a}_{1o} = a_o$$
$$\hat{a}_{3o} = \frac{a_o^{\times} s_o}{\| a_o^{\times} s_o \|}$$
$$\hat{a}_{2o} = \hat{a}_{3o}^{\times} \hat{a}_{1o}$$
$$R^{ao} = \left[\hat{a}_{1o}\hat{a}_{2o}\hat{a}_{3o}\right]^T$$

Then proceed from the left side of Equation $R^{ro} = R^{rc}R^{ca}R^{ao}$ and construct the rotation matrix R^{rc} in the same way as R^{ao}:

$$\dot{c}_{1r} = a_r$$
$$\dot{c}_{3r} = \frac{a_b^{\times} p_r}{\| a_r^{\times} p_r \|}$$
$$\dot{c}_{2r} = c_{3r}^{\times} \dot{c}_{1r}$$
$$R^{rc} = \left[\dot{c}_{1r}\dot{c}_{2r}\dot{c}_{3r}\right]$$

We now could know all of the rotation matrices in Eq. $R^{ro} = R^{rc}R^{ca}R^{ao}$ up to R^{ca}. Use R^{ca} to perform the yaw-steering maneuver thereby the sun vector being perpendicular to the solar panel axis.

Then use the prescribed orthogonality condition between the sun vector and the solar panel axis to determine the rotation matrix from F_c to F_a.

Equation $s^T p = 0$ can be expressed as

$$s_a^T R^{ac} p = 0$$

R^{ac} is not constructed using a the TRIAD method, but from a careful analysis of the kinematics that result from definition of F_c and F_a.

R^{ac} is first defined as the dot product between the base unit vectors of F_a and F_c, which can be expressed as

$$R^{ac} = \begin{bmatrix} \hat{a}_1 \cdot \hat{c}_1 & \hat{a}_1 \cdot \hat{c}_2 & \hat{a}_1 \cdot \hat{c}_3 \\ \hat{a}_2 \cdot \hat{c}_1 & \hat{a}_2 \cdot \hat{c}_2 & \hat{a}_2 \cdot \hat{c}_3 \\ \hat{a}_3 \cdot \hat{c}_1 & \hat{a}_3 \cdot \hat{c}_2 & \hat{a}_3 \cdot \hat{c}_3 \end{bmatrix}$$

Recall from Figures 4.3 and 4.4, we defined a to lie along the "1" direction. Clearly, $\hat{a}_1$ and $\hat{c}_1$ are the same vector; therefore $\hat{a}_1 \cdot \hat{c}_1 = 1$. By definition, the unit vectors of $\hat{a}_2$ and $\hat{a}_3$ are perpendicular to $\hat{a}_1$, so they are also perpendicular to $\hat{c}_1$. The same is true for $\hat{a}_1$, which is also perpendicular to $\hat{c}_2$ and $\hat{c}_3$. As a result, R^{ac} is a "1" rotation and Equation $s_a^T R^{ac} p = 0$ is

$$R^{ac} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_{ac} & \sin\theta_{ac} \\ 0 & \Leftrightarrow \sin\theta_{ac} & \cos\theta_{ac} \end{bmatrix}$$

So it could also be written as

$$s_a^T \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_{ac} & \sin\theta_{ac} \\ 0 & \Leftrightarrow \sin\theta_{ac} & \cos\theta_{ac} \end{bmatrix} p_c = 0$$

Recall that we have defined F_a and F_c so that s_{3a} and p_{3c} are zero, and thus get the further expands as

$$s_{1a} p_{1c} + s_{2a} p_{2c} \cos\theta_{ac} = 0$$

The angle θ_{ac} which satisfies the yaw-steering condition is

$$\cos\theta_{ac} = \Leftrightarrow \frac{s_{1a} p_{1c}}{s_{2a} p_{2c}}$$

If s_{3a} and p_{3c} were not zero, the above equation would be a transcendental equation involving both $\cos\theta_{ac}$ and $\sin\theta_{ac}$ terms. Solving this equation would require an iterative method, such as Newton-Rhaphson, to determine the yaw-steering angle. The rotation matrix R^{ac} is then calculated by the Equation R^{ac}. The ideal target pointing attitude R^{ri} is then constructed by multiplying together the rotation matrices found in $R^{ri} = R^{ro} R^{oi}$.

It seems that the yaw control conditions allow tracking target satisfy the requirement of the power of the solar cell array of fully automated method. However, this is the only true when the sensor axis is perpendicular to the panel P. For some panel orientations, equation $\cos\theta_{ac} = \Leftrightarrow \frac{s_{1a} p_{1c}}{s_{2a} p_{2c}}$ has associated numerical singularity on it. The panel of vector could follow imaginary cone by yaw steering operation when the panel is not perpendicular to the optical axis vector. And when the carrier can't be perpendicular to the conical surface, the singularity occurs. Obviously, the yaw-steering maneuver can't be preformed when either *s* or *p* is aligned with *a*.

We examine $\cos\theta_{ac} = \Leftrightarrow \frac{s_{1a}p_{1c}}{s_{2a}p_{2c}}$ for the cases where $0 < \gamma \le 90^{\circ}$ in Fig. 4.5 to determine when singularities occur.

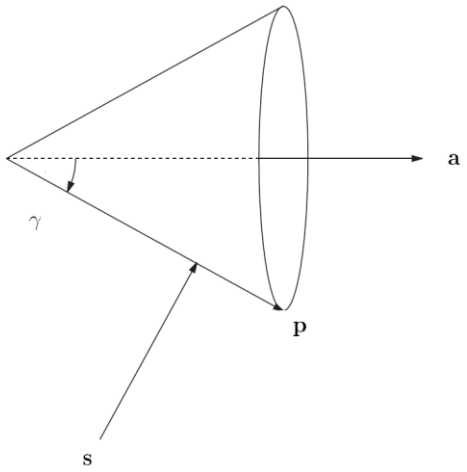

Figure 4.5: Yaw-Steering Maneuver of Graphical Illustration

Obviously, with $|\cos\theta_{ac}| = 1$, the limiting case is for $\cos\theta_{ac} = \Leftrightarrow \frac{s_{1a}p_{1c}}{s_{2a}p_{2c}}$, and singularities appear when the right side of $\cos\theta_{ac} = \Leftrightarrow \frac{s_{1a}p_{1c}}{s_{2a}p_{2c}}$ exceeds this value. If the right hand side is less than unity, no singularities will happen:

$$\left|1\frac{s_{1a}p_{1c}}{s_{2a}p_{2c}}\right| < 1$$

It could also be written as

$$\left|\frac{s_{1a}}{s_{2a}}\right| < \left|\frac{p_{1c}}{p_{2c}}\right|$$

In short, the yaw-steering maneuver can be preformed if either $s_{1a} < s_{2a}, p_{1c} < p_{2c}$, or both are satisfied. The yaw-steering maneuver can always be preformed if the a is perpendicular to p, which says that $p_{1c} = 0$. As a result, the yaw-steering maneuver is always a 90° rotation about the boresight axis.

When we know R^{ri}, we could determine the reference attitude σ_r, using

$$\begin{aligned}
\beta_0 &= \sqrt{(\text{tr}\, R + 1)}/2 \\
\sigma_1 &= (R_{2,3} \Leftrightarrow R_{3,2})/[4\beta_0(1+\beta_0)] \\
\sigma_2 &= (R_{3,1} \Leftrightarrow R_{1,3})/[4\beta_0(1+\beta_0)] \\
\sigma_3 &= (R_{1,2} \Leftrightarrow R_{2,1})/[4\beta_0(1+\beta_0)]
\end{aligned}$$

The vectors in Table 4.1 can now be expressed in all of the coordinate frames. We continue in the next section to develop needed angular velocity which is necessary to use for development of rotation matrix per frame to track the target.

4.2 The Ideal Angular Velocity

By differentiating the attitude expressions in the previous section, we develop angular velocity commands and then calculating ω_r^{ri} from the vector sum of each intermediate angular velocity expression.

So let us begin by differentiating $r_{t/s} = r_t \Leftrightarrow r_s$ to get

$$\dot{r}_{t/s} = \dot{r}_t \Leftrightarrow \dot{r}_s$$

$\dot{r}_t$ is given by $\omega_e^{\times} r_t$, where ω_e is the angular velocity of the earth

$\dot{r}_s$ is simply the known spacecraft velocity vector

Then define the angular velocity of F_o with respect to F_i. Since we are assuming that the orbit is circular, the angular velocity is known, which is just the mean motion of the orbit expressed about the negative orbit normal ($\Leftrightarrow \hat{o}_2$) as

$$w_o^{oi} = \left[0 \Leftrightarrow (\sqrt{\mu / \square r_s^T r_s \square^3}) 0 \right]^T$$

oi denotes the angular velocity of F_o with respect to F_i.

We need to calculate the angular velocity ω_a^{ao} because F_a and F_o change with time. There is not difficult to show that ω_a^{ao} can be calculated based on the differentiation of R^{ao} as[13]

$$\omega_a^{ao\times} = \Leftrightarrow \overset{\circ}{R}^{ao} R^{oa} \omega_a^{ao} F_a F_o$$

"$\circ$" denotes differentiation with respect to a moving coordinate frame. The matrix $\overset{\circ}{R}^{ao}$ is found by first rewriting Equations

$$\begin{aligned} \hat{a}_{1o} &= a_o \\ \hat{a}_{3o} &= \frac{a_o^{\times} s_o}{\| a_o^{\times} s_o \|} \\ \hat{a}_{2o} &= \hat{a}_{3o}^{\times} \hat{a}_{1o} \end{aligned} \quad \text{as} \quad \begin{aligned} D_1 \hat{a}_{1o} &= r_{t/so} \\ D_2 \hat{a}_{3o} &= r_{t/so}^{\times} s_o \\ \hat{a}_{2o} &= \hat{a}_{3o}^{\times} \hat{a}_{1o} \end{aligned}$$

D_1 and D_2 are given by $\| r_{t/so} \|$ and $\| r_{t/so}^{\times} s_o \|$ respectively.

With respect to time differentiating, the above equations results in the following

$$\begin{aligned} \dot{\hat{a}}_{1o} &= \frac{\dot{r}_{t/so} \Leftrightarrow \dot{D}_1 \hat{a}_{1o}}{D_1} \\ \dot{\hat{a}}_{2o} &= \dot{\hat{a}}_{3o}^{\times} \hat{a}_{1o} + \hat{a}_{3o}^{\times} \dot{\hat{a}}_{1o} \\ \dot{\hat{a}}_{3o} &= \frac{\dot{r}_{t/so} s_o \Leftrightarrow \dot{D}_2 \hat{a}_{3o}}{D_2} \end{aligned}$$

which have been simplified by assuming that the sun vector, slowly varies in the inertial frame, and can be considered constant in F_i.

So $\dot{s}_o$ is zero. The rates of change of D_1 and D_2 are

$$\dot{D}_1 = \frac{\dot{r}_{t/so}^T r_{t/so}}{D_1}$$

$$\dot{D}_2 = \frac{(\dot{r}_{t/so}^{\times} s_o)^T (r_{t/so}^{\times} S_o)}{D_2}$$

These derivatives have been computed with respect to F_i. We need the derivatives on behalf of the moving orbital frame, which we find using equation $\dot{v} = \overset{\circ}{v} + \omega^{\times} v$

$$\begin{aligned}
\overset{\circ}{\hat{a}}_{1o} &= \dot{\hat{a}}_{1o} \Leftrightarrow \boldsymbol{\omega}_o^{oi\times} a_{1o} \\
\overset{\circ}{\hat{a}}_{3o} &= \dot{\hat{a}}_{3o} \Leftrightarrow \boldsymbol{\omega}_o^{oi\times} a_{3o} \\
\overset{\circ}{\hat{a}}_{2o} &= \overset{\circ}{\hat{a}}_{3o}^{\times} \hat{a}_{1o} + \hat{a}_{3o}^{\times} \overset{\circ}{\hat{a}}_{1o}
\end{aligned}$$

and then R^{ao} with respect to F_o is simply

$$\overset{\circ}{R}^{ao} = \left[\overset{\circ}{\hat{a}}_{1o} \;\; \overset{\circ}{\hat{a}}_{2o} \;\; \overset{\circ}{\hat{a}}_{3o} \right]^T$$

Using $\omega_a^{ao\times} = \Leftrightarrow \overset{\circ}{R}^{ao} R^{oa}$, the angular velocity ω_a^{ai} is then found to be

$$\boldsymbol{\omega}_a^{ai} = \boldsymbol{\omega}_a^{ao} + R^{ao} \boldsymbol{\omega}_o^{oi}$$

It can be seen from

$$s_a^T \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_{ac} & \sin\theta_{ac} \\ 0 & \Leftrightarrow \sin\theta_{ac} & \cos\theta_{ac} \end{bmatrix} p_c = 0$$

that ω_a^{ca} is simply

$$\boldsymbol{\omega}_a^{ca} = \left[\Leftrightarrow \dot{\theta}_{ac} \;\; 0 \;\; 0 \right]^T$$

And $\dot{\theta}_{ac}$ is found by taking a time derivative of

$$s_a^T \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_{ac} & \sin\theta_{ac} \\ 0 & \Leftrightarrow \sin\theta_{ac} & \cos\theta_{ac} \end{bmatrix} p_c = 0$$

to yield

$$\dot{\theta}_{ac} = \frac{\dot{s}_{1a}\boldsymbol{p}_{1c} + \dot{s}_{2a}\boldsymbol{p}_{2c}\cos\theta_{ac}}{s_{2a}\boldsymbol{p}_{2c}\sin\theta_{ac}}$$

The derivative of the sun vector on behalf of F_a is given by

$$\overset{\circ}{\mathbf{s}}_a = \Leftrightarrow\boldsymbol{\omega}_a^{ai\times}\mathbf{s}_a$$

F_c has a fixed orientation with respect to F_r, so $\omega^{rc} = 0$

As a result, by adding $\omega_a^{ai} = \omega_a^{ao} + R^{ao}\omega_o^{oi}$ and $\omega_a^{ca} = \begin{bmatrix} \Leftrightarrow \dot{\theta}_{ac} & 0 & 0 \end{bmatrix}^T$ the desired tracking body rate vector ω_r^{ri} is constructed, and then rotating them into F_r

$$\boldsymbol{\omega}_r^{ri} = R^{ra}(\boldsymbol{\omega}_a^{ca} + \boldsymbol{\omega}_a^{ai})$$

R^{ra} is the rotation matrix from F_a to F_r and is found from the previous section to be

$$R^{ra} = R^{rc}[R^{ac}]^T$$

The target tracking trajectory with yaw-steering is known, when the ideal angular velocity is computed. Momentum wheels are used for maneuvering to make the virtual spacecraft follow this trajectory. The next section we talk about developing the angular acceleration commands which are used to determine the virtual spacecraft's control torque.

4.3 The Ideal Angular Acceleration

We compute the desired angular accelerations when the angular velocities are known in each of the coordinate frames. The accelerations are needed to compute the reference axial wheel torque g_{ar} to generate the desired trajectory. The angular acceleration $\dot{\omega}_r^{ri}$ is constructed analogously to the angular velocity ω_r^{ri}. The acceleration commands are found by taking a time derivative of Equation

$$\ddot{r}_{t/s} = \ddot{r}_t \Leftrightarrow \ddot{r}$$

In section 3 we define $\ddot{r}_s$ is simply the two-body equation of motion. The inertial acceleration of the target $\ddot{r}_{ti}$ is

$$\ddot{r}_{ti} = \omega_e^{\times}\dot{r}_{ti}$$

$\overset{\circ\circ}{r}_{ti}$ is zero because F_i does not rotate.

We start determining what the angular accelerations of the coordinate frames after defining the accelerations of the position vectors. The angular acceleration $\dot{\omega}_o^{oi}$ is found by differentiating

$$\omega_o^{oi} = \begin{bmatrix} 0 & \Leftrightarrow(\sqrt{\mu/\square r_s^T r_s \square^3}) & 0 \end{bmatrix}^T \text{ to get } \dot{\omega}_o^{oi} = \begin{bmatrix} 0 & 1.5(\dot{r}_s^T r_s + r_s^T \dot{r}_s)(\sqrt{\mu/\square r_s^T r_s \square^5}) & 0 \end{bmatrix}^T$$

Only when the orbit is elliptical, the equation is only true because r_s varies when the spacecraft moves in an elliptical orbit. However, we use a circular orbit for this thesis, so the above equation is zero because $\square r_s \square$ is constant.

The next component that is needed is the angular acceleration of F_a with respect to F_o. The angular acceleration $\overset{\circ}{\omega}_a^{ao}$ is found by differentiating equation $\omega_a^{ao\times} = \Leftrightarrow \overset{\circ}{R}^{ao} R^{oa}$ which yields:

$$\omega_a^{ao\times}\ \overset{\circ}{\omega}_a^{ao\times} = \Leftrightarrow \overset{\circ\circ}{R}^{ao} R^{oa} \Leftrightarrow \overset{\circ}{R}^{ao}\ \overset{\circ}{R}^{oa}$$

The second derivative of R^{ao} is found by differentiating

$$\begin{aligned} \overset{\circ}{\hat{a}}_{1o} &= \dot{\hat{a}}_{1o} \Leftrightarrow \boldsymbol{\omega}_o^{oi\times} a_{1o} \\ \overset{\circ}{\hat{a}}_{3o} &= \dot{\hat{a}}_{3o} \Leftrightarrow \boldsymbol{\omega}_o^{oi\times} a_{3o} \\ \overset{\circ}{\hat{a}}_{2o} &= \overset{\circ\times}{\hat{a}}_{3o}\ \hat{a}_{1o} + \hat{a}_{3o}^{\times}\ \overset{\circ}{\hat{a}}_{1o} \end{aligned}$$

to get

$$
\begin{aligned}
\overset{\circ\circ}{\hat{a}}_{1o} &= \ddot{\hat{a}}_{1o} \Leftrightarrow 2\boldsymbol{\omega}_o^{oi\times}\dot{a}_{1o} \Leftrightarrow \boldsymbol{\omega}_o^{oi\times}(\boldsymbol{\omega}_o^{oi\times}a_{1o}) \Leftrightarrow \dot{\boldsymbol{\omega}}_o^{oi\times}a_{1o} \\
a & \\
\overset{\circ\circ}{\hat{a}}_{3o} &= \ddot{\hat{a}}_{3o} \Leftrightarrow 2\boldsymbol{\omega}_o^{oi\times}\dot{\mathbf{a}}_{3o} \Leftrightarrow \boldsymbol{\omega}_o^{oi\times}(\boldsymbol{\omega}_o^{oi\times}a_{3o}) \Leftrightarrow \dot{\boldsymbol{\omega}}_o^{oi\times}a_{3o} \\
\overset{\circ\circ}{\hat{a}}_{2o} &= \overset{\circ\circ\times}{\hat{a}}_{3o}\hat{a}_{1o} + 2\,\overset{\circ\times}{\hat{a}}_{3o}\,\overset{\circ}{\hat{a}}_{1o} + \hat{a}_{3o}^{\times}\overset{\circ\circ}{\hat{a}}_{1o} \\
\overset{\circ\circ}{\mathbf{R}}^{ao} &= \left[\overset{\circ\circ}{\hat{a}}_{1o}\;\; \overset{\circ\circ}{\hat{a}}_{2o}\;\; \overset{\circ\circ}{\hat{a}}_{3o}\right]^T
\end{aligned}
$$

and the inertial derivatives of the unit vectors are found by differentiating

$$
\dot{\hat{a}}_{1o} = \frac{\dot{r}_{t/so} \Leftrightarrow \dot{D}_1\hat{a}_{1o}}{D_1}
$$

$$
\dot{\hat{a}}_{3o} = \frac{\dot{r}_{t/so}s_o \Leftrightarrow \dot{D}_2\hat{a}_{3o}}{D_2}
$$

so we get

$$
\begin{aligned}
\ddot{\hat{a}}_{1o} &= \frac{\ddot{\mathbf{r}}_{t/so} \Leftrightarrow 2\dot{D}_1\dot{\hat{a}}_{1o} \Leftrightarrow \ddot{D}_1\hat{a}_{1o}}{D_1} \\
\ddot{\hat{a}}_{3o} &= \frac{\ddot{\mathbf{r}}_{t/so}^{\times}\mathbf{s}_o \Leftrightarrow 2\dot{D}_2\dot{\hat{a}}_{3o} \Leftrightarrow \ddot{D}_2\hat{a}_{3o}}{D_2} \\
\ddot{D}_1 &= \frac{(\ddot{\mathbf{r}}_{t/so}\cdot\mathbf{r}_{t/so}) + (\dot{\mathbf{r}}_{t/so}\cdot\dot{\mathbf{r}}_{t/so}) \Leftrightarrow \dot{D}_1^2}{D_1} \\
\ddot{D}_2 &= \frac{(\ddot{\mathbf{r}}_{t/so}^{\times}\mathbf{s}_o)\cdot(\mathbf{r}_{t/so}^{\times}\mathbf{s}_o) + (\dot{\mathbf{r}}_{t/so}^{\times}\mathbf{s}_o)\cdot(\dot{\mathbf{r}}_{t/so}^{\times}\mathbf{s}_o) \Leftrightarrow \dot{D}_2^2}{D_2}
\end{aligned}
$$

and then $\dot{\omega}_a^{ai}$ becomes

$$
\dot{\omega}_a^{ai} = \dot{\omega}_a^{ao} + R^{ao}\dot{\omega}_o^{oi}
$$

Meanwhile, $\overset{\circ}{\omega}_a^{ca}$ is found by differentiating the expression in $\omega_a^{ca} = \left[\Leftrightarrow \dot{\theta}_{ac} \quad 0 \quad 0\right]^T$ where

$$
\ddot{\theta}_{ac} = \frac{\ddot{s}_{1a}p_{1c} \Leftrightarrow 2\dot{\theta}_{ac}\dot{s}_{2a}p_{2c}\sin\theta_{ac} \Leftrightarrow \dot{\theta}_{ac}^2 s_{2a}p_{2c}\cos\theta_{ac} + \ddot{s}_{2a}p_{2c}\cos\theta_{ac}}{s_{2a}p_{2c}\sin\theta_{ac}}
$$

The acceleration of the sun vector on behalf of $\mathcal{F}_a$ is given by

$$
\overset{\circ\circ}{\mathbf{s}}_a = \Leftrightarrow 2\boldsymbol{\omega}_a^{ai\times}\,\overset{\circ}{\mathbf{s}}_a \Leftrightarrow \boldsymbol{\omega}_a^{ai\times}(\boldsymbol{\omega}_a^{ai\times}\mathbf{s}_a)
$$

The desired angular acceleration becomes

$$\dot{\omega}^{ri} = R^{ra}(\overset{\circ}{\omega}_a^{ca} + \dot{\omega}_a^{ai})$$

R^{ra} is the rotation matrix from F_a to F_r.

With F_r using $R^{ra} = R^{rc}\left[R^{ac}\right]^T$, the desired acceleration vector $\dot{\omega}^{ri}$ is constructed from knowing $\overset{\circ}{\omega}_a^{ca}$ and $\dot{\omega}_a^{ai}$. So when get $R^{ri}\omega^{ri}$ and $\dot{\omega}^{ri}$, we can completely describe the desired trajectory that the real spacecraft needs to obtain in order to track a target. According to the next section, we show how this open loop reference trajectory is applied in the control law's derivation that will asymptotic let any initial tracking errors be zero in the body frame.

Section 5

Spacecraft Control Laws

The thesis develops a nonlinear feedback control law to track a goal, which let the actual craft present a rotational motion in using momentum wheel track attitude and angular velocity proposed in section 4. To perform the tracking maneuver, the spacecraft uses momentum wheels generating control torques. Stability and control theory of the control law is derived with using the ADCS. The definitions of these errors in section 5.1 would be used with the equation of state in the development of the controller. In section 5.2 we study the ADCS's some standard equations. The tracking errors in the attitude and angular velocity are stabilized by this controller asymptotically.

5.1 The Error Kinematics Problem

It is different to the actual body frame and the reference frame at the beginning of a rotational tracking maneuver. This difference is called the tracking error between the two frames. To track errors between F_b and F_r could be calculated by the attitude, angular velocity, angular momenta, angular acceleration, and control torques. Thus, the thesis only considers the attitude error and the angular velocity error for the derivation of our control law. Construct the attitude tracking error from the rotation matrix which is assembled from the actual body frame attitude vector and the rotation matrix which is assembled from the ideal pointing attitude. The attitude error is

$$R^{br}(\delta\sigma) = R^{bi}(\sigma_b)R^{ir}(\sigma_r)$$

$R^{br}(\delta\sigma)$ is the rotation matrix from the reference frame F_r to the body frame F_b

$\delta\sigma$ is the error in the attitude between the frames F_b and F_r

The error in the angular velocity is simply the difference of the actual and virtual spacecraft's angular velocities. The tracking error of the angular velocity is represented in F_b as

$$\delta\omega = \omega_b \Leftrightarrow R^{br}(\delta\sigma)\omega_r$$

Using $\dot{\sigma} = G(\sigma)\omega$ references from 11 and 18, for the error kinematics the differential equation becomes

$$\delta\dot{\sigma} = G(\delta\sigma)\delta\omega$$

In the actual implementation, the computing expenses are reduced by using $R^{br}(\delta\sigma) = R^{bi}R^{ir}(\sigma_r)$ instead of $\delta\dot{\sigma} = G(\delta\sigma)\delta\omega$ to determine $\delta\sigma$.

5.2 The ADCS Controller

5.2.1. Attitude Control

Three magnetic coils with distance bias momentum wheel are joint design of satellite attitude control In TT - 1. In this section, both damping control and tri-axial stability control algorithm in detail.

5.2.2. Control of Damping

After separation, the satellite is assumed in the random initial state at large angular velocity. At this stage, only the magnetic force could be used as a sensor, and the magnetic coils are used for attitude damping control as the main actuators. The main purpose of this phase is to reduce the angular velocity and magnetic coils.

B - point method[19] this stage is the most popular algorithms because of its fast convergence and low computational cost. However, in the method of point *B* is seriously affected by the

magnetometer measurement noise. For improving the control precision, the angle between component vector in Y-axis and B_{bo} is firstly defined as follows:

$$\alpha_y = \arccos \frac{B_y}{\sqrt{B_x^2 + B_y^2 + B_z^2}}$$

The component vectors of B_{bo} in X-axis, Y-axis and Z-axis respectively are B_x, B_y and B_z.

The Magnetic measurement calculates time derivate of xy at each sample time interval:

$$\dot{\alpha}_y(t_k) = \frac{\alpha_y(t_k) - \alpha_y(t_{k-1})}{t_k - t_{k-1}}$$

The damping control is divided into two phases.

Stage 1: With magnetic coils in Y-axis, dump the angular velocity of X-axis and Y-axis, and establish the Y-Thomson steady state. Likewise, Y-axis is driven to the normal direction of orbit plane.

The component vector of dipole moment M in Y-axis is showed as follows:

$$M_y(t_k) = -k_y \dot{\alpha}_y(t_k)$$

k_y is the control gain

Stage 2: Let us dump the angular velocity of Y-axis with Z-axis or X-axis magnetic coil to a reference angular velocity ω_{ref}. So the component vector of M in X-axis or Z-axis is showed as follows:

$$\begin{cases} M_x = k_x(\omega_y - \omega_{ref})\,\mathrm{sgn}(B_z) \\ M_z = k_z(\omega_y - \omega_{ref})\,\mathrm{sgn}(B_x) \end{cases}$$

ω_y is the angular velocity of Y-axis

k_x and k_z are the control gains

5.2.3. Control of Three-axis Stabilization

After the stage of damping control, the proportional-derivative control law is adopted to adjust the satellite to the desired orientation. The control law of the magnetic coils is

$$T_c = -K_p\Theta - K_q\omega_{bo}$$

K_p, K_q are coefficient matrix of the control gains

$\Theta = [\phi, \quad \theta, \quad \psi]^T$ represents the attitude angle

ϕ, ϕ and ψ represent respectively roll, pitch and yaw angle

We know that the dipole moment M of magnetic coils interacts with geomagnetic field B to produce the control torque:

$$T_c = M \times B$$

Dictated by the magnetic coils, the constraint is that T_c only generates in the orthogonal direction with magnetic field B. Therefore on the magnetic dipole moment, the best control law applied is

$$M = (B \times T_c)/\square B\square^2$$

Section 6

On-orbit Tracking Data Analysis

In this section, by ground station, the data received is used to analyze the ADCS on-orbit TT-1performance. With the altitude of 480 km, the satellite operates in a sun synchronous orbit, the eccentricity of 0.000454 and the inclination angle of 97.3˚. So we analyze respectively on the performances of damping phase and stabilization phase. So we could get the result that the nadir pointing accuracy could be limited within ±5˚ and the accuracy of three-axis control is about ±10˚.

6.1. Damping Phase Analysis

This phase's objective is to reduce the angular velocity by damping control strategy. From study other research we get the Figure 6.1 which clearly shows the attitude angular velocity after successful launch separation. Also from above Figure 6.1, we could see that the attitude angular velocity after separation was within 1.5 (˚)/s. So it is very clear that the separation design is very successful. After separation, the momentum wheel speeded up rapidly and steady at a fixed rate in10 s. Figure 6.2 shows the control state after separation. From Figure 6.2, from damping control to proportional-derivative control, it could be seen that the control mode was successfully changed. The damping control was from 07:19:52 to 07:20:13. It was lasted for 21s. Because of the small angular velocity after separation, the damping phase is short. Then the attitude control state turned into stabilization control. Figure 6.3 shows the attitude angular velocity in the second orbit. Obviously, the attitude angular velocity had been constrained within 0.12 (˚)/s. Moreover,

the attitude angular velocity xx in X-axis and the attitude angular velocity xz in Z-axis had converged to the range of ±0.06 (˚)/s. This shows that it is suitable for TT-1 nano-satellite with the damping control algorithm.

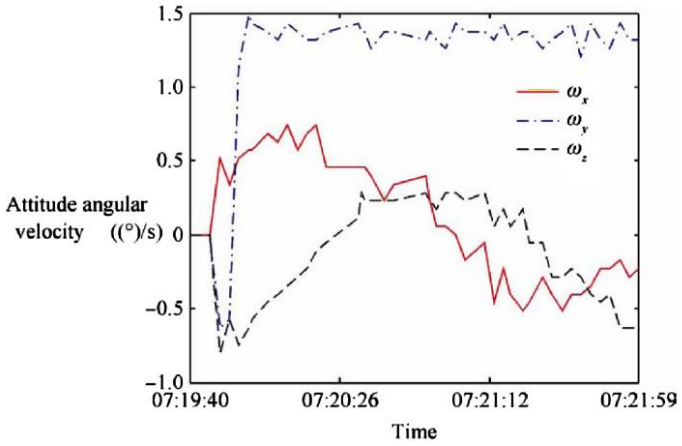

Figure 6.1: After separating, the changing of the attitude angular velocity (data from the website)[20]

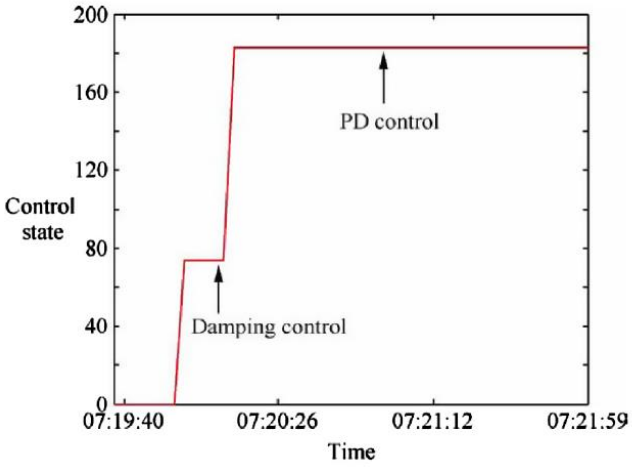

Figure 6.2: After separation, the changing of the control state (data from the website)

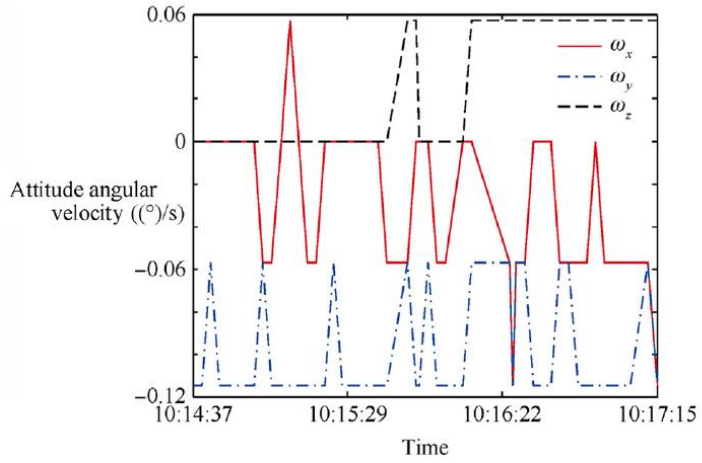

Figure 6.3: In the second orbit, the changing of the attitude angular velocity (data from website).

6.2. Analysis of Three-axis Stabilization Phase

When finishing the damp control phase, the satellite would turn into the three-axis stabilization control mode. First, we could analyze the performance of this phase in sunlight area. From Figure 6.4 to Figure 6.6 showing that the time period is from 06:50:18 to 06:54:29. From Figure 6.4, when the pitch angle was constrained within 3˚, we could know that three-axis attitude angle had been constrained within ±6˚. Furthermore, it is clearly that the pitch angle was more stable than the roll angle and the yaw angle. Thus, the result is that the momentum wheel control in pitch axis is effective. From Figure 6.5, we could see that the attitude angular velocity of roll and yaw axis had converged to the range of ±0.06 (˚)/s, and the pitch angular velocity had been constrained within ±0.12 (˚)/s. At the same time, the wheel speed was steady at about 6100 r/min shown in Figure 6.6. Therefore, we could know that the performance of ADCS is satisfying in sunlight area.

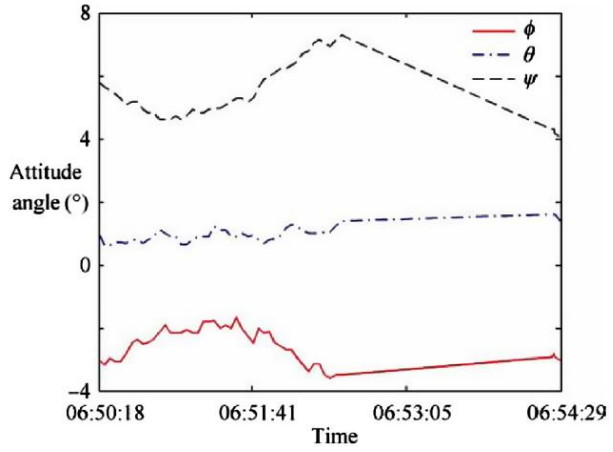

Figure 6.4: The changing of the attitude angle in sunlight (data from the website)

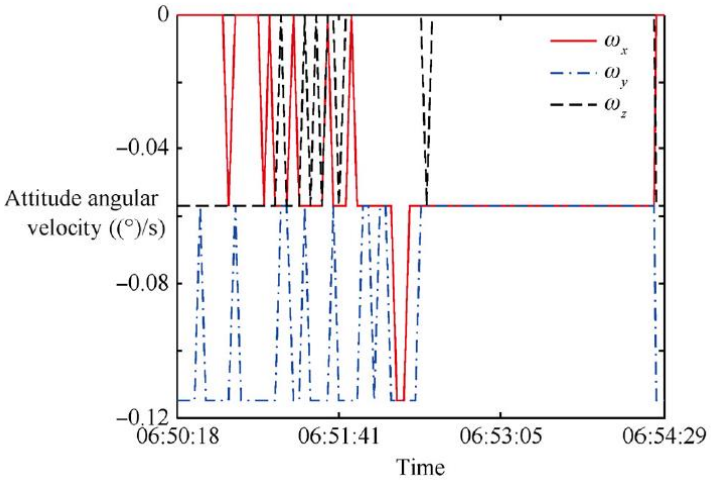

Figure 6.5: The changing of the attitude angular velocity in sunlight (data from the website)

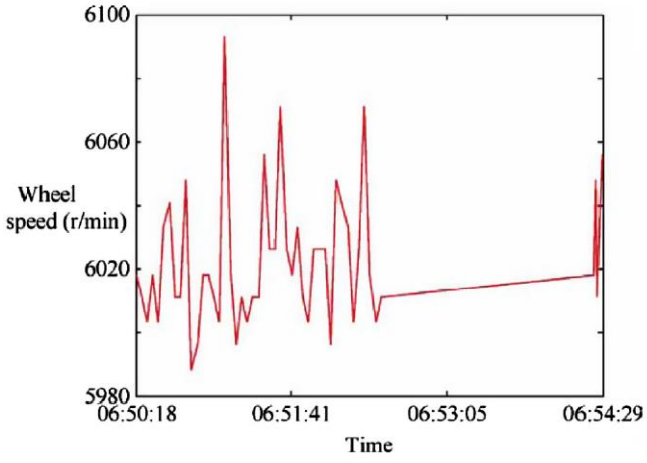

Figure 6.6: The changing of the momentum wheel speed in sunlight (data from the website)

ADCS in Eclipse performance is shown in Figure 6.7 to Figure 6.9.The time period is from 17:36:14 to 17:37:22.At that time, the satellite into space is more than 10 days. In eclipse, sun sensors cannot make effective response to determine attitude information, and could only get magnetic measurement vector. Therefore, in Eclipse ADCS performance will be worse than in the sunshine areas. From Figure 6.7 to Figure 6.9, it can show that all three position angle errors' absolute value were limited in 20 ˚, and pitch attitude angle error was about 5 ˚ slowly changing. At the same time, the angular velocity of yawing and rolling shaft were limited in 0.1 (˚)/s, and the velocity of the pitch attitude was at the rate in 0.12(˚)/s. Therefore, we can conclude that the ADCS can also work effectively even if in Eclipse, and satellite is tri-axial stability. However, compared with the sunlight region performance, the control precision is much better than that in eclipse.

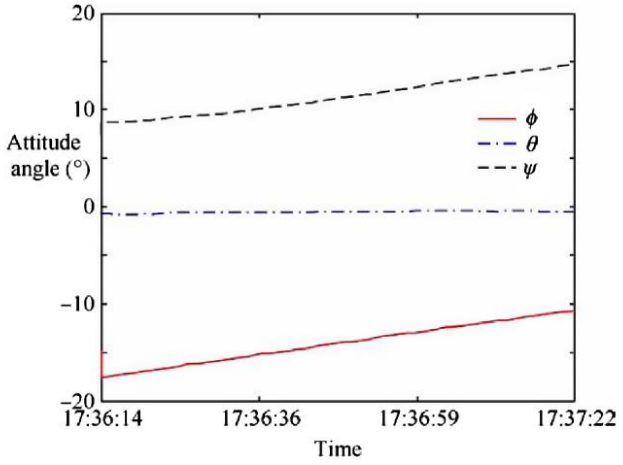

Figure 6.7: The changing of the attitude angle in eclipse (data from the website)

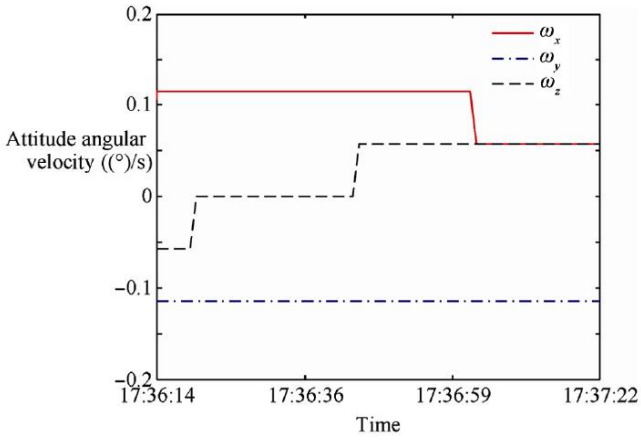

Figure 6.8: The changing of the attitude angular velocity in eclipse (data from the website)

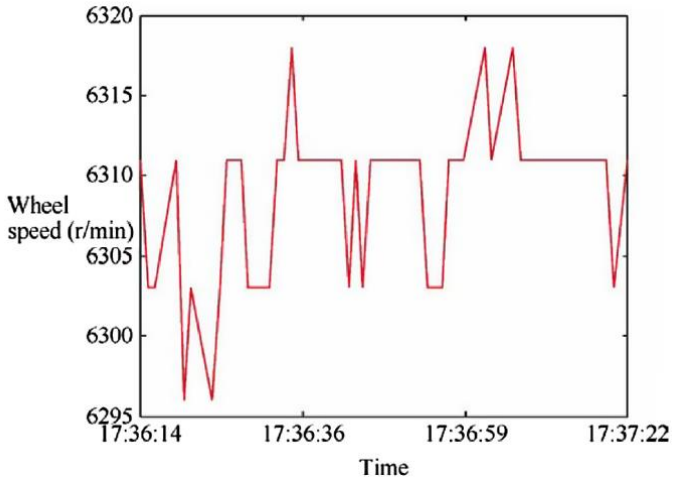

Figure 6.9: The changing of the momentum wheel speed in eclipse (data from the website)

Section 7

Summary and Conclusions

For computing a multi-axis target tracking trajectory, method is developed that also allows the solar panel vector to remain perpendicular to the sun vector during a tracking maneuver. The nano-satellite TT-1 spacecraft is made to track this reference trajectory through the ADCS control law driving the initial tracking errors asymptotically to zero. For tracking rotational maneuvers, the control law generates internal torques provided by momentum wheels.

Because only known vectors are needed, constructing the reference trajectory in a TRIAD-like manner is advantageous. We construct the reference motion only use the spacecraft position, velocity, sensor, panel, and sun vectors. The reference trajectory is unique for each tracking problem due to the fact that the position vectors are specific to a particular target. Also, the reference attitude is constructed in the form of rotation matrix, which makes it coordinate independent. It gives greater flexibility for spacecraft maneuver design by an attitude coordinate independent reference attitude. From the research, we could know that other authors have developed similar algorithms, but they were designed for a particular type of attitude parameters and rotational maneuver and lack flexibility for use in various mission situations.

By differentiating the attitude to compute the benchmark angular velocity and acceleration is a direct way. However, it will involve some algebraic manipulation to compute the derivatives on behalf of the rotating coordinate frames.

Moreover, we know that TT-1 has been on orbit for over one year which far exceeds one month lifespan, and it has accomplished all the missions successfully. As the first single-board

nano-satellite in China, TT-1 also verifies the single-board structure and actualizes no cable connection.

In the thesis, the design of ADCS subsystem is showed in detail and its on-orbit performance is analyzed. We can make a conclusion that the initial design of ADCS for TT-1 is feasible and suitable according to the original telemetry data. In sunlight area, the attitude nadir pointing accuracy is within 5˚ which is better than the designed accuracy of 10˚. In eclipse, ADCS can also work in expect.

In the further study, according to the on-orbit performance analysis of TT-1 mission, we can get more useful suggestions and conclusions. Also, gaining a precise attitude control, variable speed momentum wheel should be considered.

Bibliography

[1] Wertz, J. R., editor, *Spacecraft Attitude Determination and Control*, D. Reidel, Dordrecht, Holland, 1978.

[2] Gramling, C. J., Lee, T., Niklewski, D. J., and Long, A. C., "Relative Navigation For Autonomous Formation Flying Of Spacecraft," In *Proceedings of the AAS/AIAA 1997 Astrodynamics Specialist Conference*, pp. 1–16, August 1997.

[3] Folta, D. C. and Quinn, D. A., "An Algorithm For Enhanced Formation Flying of Satellites In Low Earth Orbit," In *Proceedings of the American Institute of Physics Space Technology and Applications International Forum*, pp. 803–822, 1998.

[4] Folta, D., Bordi, F., and Scolese, C., "Considerations On Formation Flying Separations For Earth Observing Satellite Missions," In *Advances in Astronautical Sciences*, Vol. 79, No. 2, 1992, pp. 803–822.

[5] Ulybyshev, Y., "Long-Term Formation Keeping of Satellite Constellation Using Linear-Quadratic Controller," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 1, 1998, pp. 109–115.

[6] Hablani, H. B., "Design of a Payload Pointing Control System for TrackingMoving Objects,"*Journal of Guidance, Control, and Dynamics*, Vol. 12, No. 3, 1989, pp. 365–374.

[7] Hablani, H. B., "Multiaxis Tracking and Attitude Control of Flexible Spacecraft with Reaction Jets," *Journal of Guidance, Control, and Dynamics*, Vol. 17, No. 4, 1994, pp. 831–839.

[8] Hablani, H. B., "Target Acquisition, Tracking, Spacecraft Attitude Control, and Vibration Suppression with IPFM Reaction Jet Controllers," In *Proceedings of the AIAA Guidance, Navigation and Control Conference*, pp. 1118–1137, Hilton Head, SC, August 1992.

[9] Ogata, K. , *Modern Control Engineering*, Prentice-Hall, Englewood Cliffs, New Jersey, third edition, 1991.

[10] Hall, C. D., "Escape From Gyrostat Trap States," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 3, 1998, pp. 421–426.

[11] Hall, C. D., Tsiotras, P., and Shen, H., "Tracking Rigid Body Motion Using Thrusters and Momentum Wheels," In *1998 AIAA/AAS Astrodynamics Conference*, pp. 302–307, August 1998.

[12] Kalweit, C. C., "Optimum YawMotion for Satellites with a Nadir-Pointing Payload," *Journal of Guidance, Control, and Dynamics*, Vol. 6, No. 1, 1983, pp. 47–52.

[13] Hughes, P. C., *Spacecraft Attitude Dynamics*, John Wiley & Sons, New York, 1986.

[14] Hall, C. D., "Spinup Dynamics of Gyrostats," *Journal of Guidance, Control, and Dynamics*, Vol. 18, No. 5, 1995, pp. 1177–1183.

[15] Meirovitch, L., *Methods of Analytical Dynamics*, McGraw-Hill, New York, 1970.

[16] Vallado, D. A., *Fundamentals of Astrodynamics and Applications*, McGraw-Hill, New York,1997.

[17] Bar-Itzhack, I. Y. and Harman, R. R., "Optimized TRIAD Algorithm for Attitude Determination,"*Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 1, 1996, pp. 208–211.

[18] Schaub, H., Robinett, R. D., and Junkins, J. L., "Globally Stable Feedback Laws for Near-Minimum-Fuel and Near-Minimum-Time Pointing Maneuvers for a Landmark-Tracking Spacecraft," *Journal of the Astronautical Sciences*, Vol. 44, No. 4, 1996, pp. 443–466.

[19] Meng T, Wang H, Jin ZH, Han K. *Attitude stabilization of a picosatelliteby momentum wheel and magnetic coils*. J Zhejiang Univ Sci A 2009;10(11):1617 - 23.

[20]Ran Dechao, Sheng Tao *, Cao Lu, Chen Xiaoqian, Zhao Yong *Attitude control system design and on-orbitperformance analysis of nano-satellite – – "Tian Tuo 1"* Chinese Journal of Aeronautics, (2014),27(3): 593 – 601.

Vita

Shiqiao Zhu was born on July 8, 1991, in the city of Harbin, China. He studied Mechanical Design, Manufacturing, and Automation from 2010 to 2014 at Shandong University, China. Shiqiao Zhu completed his Bachelor of Engineering degree, along with a dual degree in Finance, in June 2014. He began his graduate studies in Mechanical Engineering and Mechanics at Lehigh University as of the Fall 2014 semester. He is pursuing his Master of Science degree under the guidance of Dr. Terry Hart.