CONTROLS ALGORITHM FOR A SATELLITE USING EARTH'S MAGNETIC FIELD: ORBIT MANEUVERS & ATTITUDE POSITIONING

by
KARTHIK G. GANESH
B.Tech,SASTRA University,2005

A thesis submitted in partial fulfillment of the requirements
for the degree of Master of Science in Aerospace Engineering
in the Department of Mechanical, Materials and Aerospace Engineering
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando,Florida

Fall Term
2007

Major Professor : Roger.W.Johnson

ABSTRACT

This document describes the design, analysis of Orbit Maneuvers and Attitude Control for NanoSat class satellites, which uses an electro-magnetic force controller which was proposed by the Florida Space Institute (FSI).

Orbit Maneuvering and the Attitude Control System (ACS) play a very important role for the success of this mission, as that can allow making the satellite go to the desired orbit as well do the sun pointing of the solar arrays with sufficient accuracy to achieve desired power levels. The primary mission would be to attain attitude stabilization using the torque from the coils. This is also used for pointing at the direction of the sun, for achieving desired power levels. The secondary mission would be to use the force of the magnetic field and utilize that for orbit maneuvering, and attain the desired trajectory.

This thesis gives a presentation of this detailed analysis with a simulation using Matlab/Simulink. Mathematical model of the actuators and sensors used for this satellite are designed, so that the simulation gives us results very near to the actual ones.Health Monitoring is also one of the main issues addressed in this work. This simulation helps us in understanding the mission as well as the requirements very well, and helps us know all the shortcomings.

The FUNSAT satellite is modeled as an example in Simulink together with a Kalman filter for attitude estimation based on all sensor measurements. The theory behind this, and extending the Kalman filter, is also presented.

"Each of us has cause to think with deep gratitude of those who have lighted the flame within us."

Albert Schweitzer

Dedicated to those who made this possible:

Father

Mother

Sister

Dr. R. Sridharan

Mr. Sankar Subramaniyan

ACKNOWLEDGMENTS

First and Foremost, I would like to thank my Advisor Dr. Roger. W. Johnson for his support. I am blessed to be acquainted with a person whose help, support and knowledge has made me a better person. His technical insights and adept guidance have helped me in my research work. He has been of great support to me during my crest and troughs at work. I have also been deeply inspired by his perseverance and commitment to work.

I would like to thank Dr. Kurt Lin and Dr. Chan Ham for serving in my committee, and for their invaluable guidance for completion of my thesis.

I would like to thank Florida Engineering Education Delivery System (FEEDS) now known as Center for Online and Virtual Education (COVE) for providing me financial support throughout the course of my studies. I would like to thank Dr. Alfred Ducharme, Ms. Kim Okamoto, and Ms. Naomi Morris for their support during my time in FEEDS. Time management is an invaluable asset which I gained from my experience with FEEDS.

I am grateful to Mr. Clinton Plaisted and Mr. Luis Santiago for their help with Matlab/Simulink programming.

I would like to thank my friends Anan, Anis, Bhaskar, Diala, Dipika, George, Harish, Jason, Kumar, Lin, Moorthi, Ramona, Sindhu, Tuan and Yixin for all their support and encouragement throughout my stay at UCF.

Finally, I would like to thank all my relatives and friends back home for their love and unconditional support.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER ONE: INTRODUCTION

1.1 History

The Florida University Satellite (FUNSAT) Competition was established by the Florida Space Grant Consortium in order to allow Florida University Students the opportunity to go through the design process for a pico satellite. The competition allows students to network with space professionals on a project that has significant contribution to space exploration. The winning team will be allowed to design their Flight Spacecraft and all materials costs will be covered through the Space Grant Consortium. The University of Central Florida has participated in the FUNSAT competition for two years.

1.2 Purpose

The basic premise of this Thesis is to identify and track the mission requirements of a Nano-Sat Class of Satellites and to document the design criteria by which the project is based. This Thesis serves as a supplement to design reports listing design decisions and the resulting hardware selections. The requirements identified within are considered crucial to the success of those satellites. Failure to meet any single requirement will impact the success of the overall project and will subsequently place the primary goals of Satellite in jeopardy.

The Florida University Satellite (FUNSAT) is an example of a Nanosat designed by the University of Central Florida with the intended purpose of testing electro-magnetic force controller while in a LEO. Once inserted into orbit, the FUNSAT will be a self-sufficient spacecraft complete with a power generation and distribution system, a communications system to the Earth Ground Station, and an Attitude Control System (ACS).

This Thesis concentrates on developing electro-magnetic controller for orbital maneuvering and the attitude control of a Nanosat with FUNSAT as an example. Change of orbital elements help in orbital maneuvering, and the ACS is the subsystem having the function of stabilizing the satellite and orienting it in the desired direction during the course of the mission, correcting for external perturbation torques that may be encountered. The ACS consists of torque coils, a magnetometer, an accelerometer, gyroscopes, and a solar vector, in addition to the computer hardware and software required to execute its tasks.

CHAPTER TWO: MISSION

2.1 Mission Design

The primary function of satellite mission analysis is to select the optimum orbit which best enables the satellite and payload to perform the mission. This task is performed by first analyzing the mission, payload, and satellite design requirements to determine if the mission is feasible. Providing the mission is feasible, trade-offs are performed in order to find a suitable orbit that meets the mission goals. This report includes a description of the various mission requirements, an introduction to orbital dynamics, an analysis of the effects of orbital perturbations and launch vehicle injection errors on the mission, a description of analysis performed to select the final orbit parameters, and a preliminary analysis of ground station line-of-sight (LOS) for the various phases of flight.

2.2 Mission Requirements

The objective of this thesis is to demonstrate an electro-magnetic force controller in space on a Nano-Sat Class Satellite using the parameters of the Cubesat(Eg : UCF FUNSAT Project). It is important to maximize power generation through the solar panels to provide enough power, as well as maintain normal satellite operations (e.g. ACS, Communications,

etc.). Implementing an effective ACS for such a small satellite should allow sun pointing of the solar arrays with sufficient accuracy to achieve desired power levels. To achieve a sun pointing capability, orbit design is a key element. Finally, thermal issues of the spacecraft structures can only be mitigated if a suitable orbit is analyzed.

The following is the mission sequence of events:

1. Separate from P-Pod (No power in batteries, solar panels begin charging)
2. Deploy Solar Arrays (e.g. Mechanical delay and latch release)
3. Systems turn on
    - tp = time required for enough power for system turn on (computer, actuators, sensors, etc)
    - tr = time for getting the position information from the satellite state (NORAD TLE or LV)
4. Rate Damping Mode (Safe Mode)
    - Control system attempts to null attitude rates
5. Telemetry Check
    - Communication contact (e.g. beacon signal) and link verification
    - Send the NORAD data which will be input to algorithm to propagate the trajectory
6. Sun Acquisition Mode
    - Calculate the Sun vector from solar panel intensity in flight software
    - Desired to point solar array plane normal to Sun vector (max power gen.)

- Control algorithm gives a torque command (which is converted to current commands) to the torque coils to maintain sun pointing

7. Locate the Position of the Target Satellite (SAT II) using NORAD TLE position data
8. Control Algorithm for reaching final states (Sat II) from initial states (SAT I) i.e. Orbit Maneuvering
9. Force and Torque generated, Thrust needed for satellite to orbit maneuver
    - Toggle between Orbit Maneuvering and ADCS for position change (Orbit) and Power Levels
    - Take Images at regular intervals to see how close we are to the Target Satellite.
10. De-Orbit Mode
    - Check power reserves to de-orbit
    - Command to de-orbit attitude Repeat steps 7 to 9 to make sure that the satellite is close to target satellite(affects power, position)
11. Thrust (Force) Command Final
    - Check telemetry for de-orbit confirm
12. End of mission

CHAPTER THREE: ORBITAL MANEUVERS

For the mission to be a success, first of all we need to make sure that the difference between the actual and desired states are as less as possible. Now for the difference to be as close as possible, then maneuvers need to be performed. In order to control the satellites for such maneuvers, the orbital motion need to be understood very well. The maneuvers can always be visualized using simulations and iterative methods.

3.1 Trajectory Simulation

Now as stated in Appendix A.1, orbital elements can be described using six parameters. We are representing the 2-D motion, using radial position r, the perpendicular velocity V_S, the radial velocity V_R, and the true anamoly υ. A 2-D representation of the above is given in Figure 1.

The derivations of the 4 state equations are given in Appendix B. The resultant equations which are used for simulating the Trajectory are given below.

$$\frac{dr}{dt} = V_R \tag{3.1}$$

$$\frac{dV_R}{dt} = \frac{V_S^2}{r} - \frac{\mu}{r^2} \tag{3.2}$$

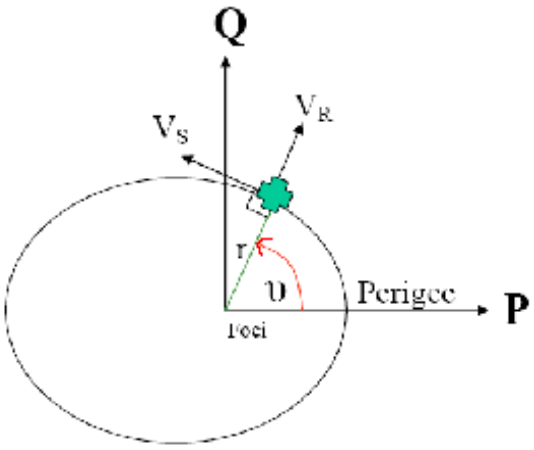

Figure 1: Orbit Representation - 2D Motion

$$\frac{dV_S}{dt} = -\frac{V_R \cdot V_S}{r} \tag{3.3}$$

$$\frac{dv}{dt} = \frac{V_S}{r} \tag{3.4}$$

Orbital maneuvers can be classified as planar and non-coplanar. Planar maneuvers manipulate the four state equations which is given by radial distance, velocity in perpendicular and radial directions, and angular position, by thrusting in the radial and perpendicular direction. Non-coplanar maneuvers manipulate the inclination and ascending node by thrusting in the out of plane direction. Out of plane maneuvers don't affect the planar orbit, but they do affect the position of the orbit within the orbital plane. Since the satellite's planar position is affected by the argument of perigee, an out of plane maneuver affects the

position of the perigee. The ascending node and inclination angle are only affected by out of plane thrusting. Planar and non-coplanar maneuvers are very different in their approach and must be dealt with as separate entities. Planar maneuvers occur solely in the radial and perpendicular direction. The state equations are used to solve for the necessary ΔV's to place the satellites in their perspective planar orbital position.

3.2 Planar Maneuvers

The Earth's oblateness and drag are natural perturbations that slightly change the orbit in time. However, some artificial perturbations can be applied to change the orbit to a desired state. These artificial perturbations, or the thrust components, are applied by the thrusters to modify the satellite's orbit. In the state equations, the thrust component is always added to the perturbations in their respective direction. To control the satellite in translation, three different thrusts are applied in the radial, perpendicular and out of plane direction.

Now when the trajectory of the satellite is simulated, these perturbations as well as thrust need to be considered. The derivations of the J2 perturbations are given in Appendix E. The final derived equations are given below,

$$R_{J2} = \frac{-3 \cdot \mu \cdot R_{earth}^2 \cdot J_2}{2 \cdot r^4}[1 - 3(\sin(i) \cdot \sin(\omega + \upsilon))^2] \tag{3.5}$$

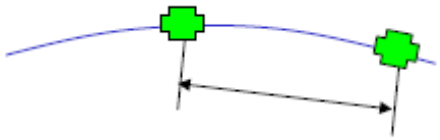

Figure 2: Planar Maneuvers

$$S_{J2} = \frac{-3 \cdot \mu \cdot R_{earth}^2 \cdot J_2}{r^4}[(\sin(i)^2 \cdot \sin(\omega + \upsilon) \cdot \cos(\omega + \upsilon)] \tag{3.6}$$

$$W_{J2} = \frac{-3 \cdot \mu \cdot R_{earth}^2 \cdot J_2}{r^4}[(\sin(i) \cdot \sin(\omega + \upsilon) \cdot \cos(i)] \tag{3.7}$$

and the drag perturbations are derived in Appendix D, and the final equations are given below,

$$Drag_R = \frac{C_d \cdot \rho \cdot A}{2m} \cdot \sqrt{V_R^2 + V_S^2 + V_W^2 \cdot V_R} \tag{3.8}$$

$$Drag_S = \frac{C_d \cdot \rho \cdot A}{2m} \cdot \sqrt{V_R^2 + V_S^2 + V_W^2 \cdot V_S} \tag{3.9}$$

$$Drag_W = \frac{C_d \cdot \rho \cdot A}{2m} \cdot \sqrt{V_R^2 + V_S^2 + V_W^2 \cdot V_W} \tag{3.10}$$

To finalize the effects of the perturbations in the equations on the satellite, we simply sum the components,

$$P_R = R_{J2} + Drag_R \tag{3.11}$$

$$P_S = S_{J2} + Drag_S \tag{3.12}$$

$$P_W = W_{J2} + Drag_W \tag{3.13}$$

Hence the state equations for the satellite become,

$$\frac{dr}{dt} = V_R \tag{3.14}$$

$$\frac{dV_R}{dt} = \frac{V_S^2}{r} - \frac{\mu}{r^2} + P_R + U_R \tag{3.15}$$

$$\frac{dV_S}{dt} = -\frac{V_R \cdot V_S}{r} + P_S + U_S \tag{3.16}$$

$$\frac{dv}{dt} = \frac{V_S}{r} \tag{3.17}$$

3.3 Non-Planar Maneuvers

Non-coplanar maneuvers are more direct and less systematic than planar maneuvers since there's only one thrust direction that affects it,because of the sines and cosines in the inclination and ascending node rates of change equations.Certain desired maneuvers are pointless in some areas of the orbit since they won't have much of an affect. These areas change places for ascending node and inclination change maneuvers. There are also areas where a desired maneuver to change one angle might have a negative effect on the other angle. One thrust direction is used to change two angular components. The inclination angle also has an affect on the rate of change of the ascending node. The closer the inclination is to zero, the higher the angular velocity of the ascending node. There is a singularity at zero inclination, where the ascending node ceases to exist and there's only a planar position at the equator. That is why there is no ascending node change at i = 0. The optimum maneuvering positions are at the nodes for inclination change, and where the summation of the true anomaly and argument of perigee are at 90° and 270° for ascending node change. On the other hand, there is no ascending node change at the nodes, and no inclination change where the summation of the argument of perigee and true anomaly are 90° and 270°. This is due to the sine in the ascending node rate being at a max at 90° and 270° and zero at the nodes. Also, the cosine in the inclination rate is at a max at the nodes and is zero at 90° and 270°. Since fuel usage is maximized in these spots, orbital maneuvers will be primarily performed close to the optimum maneuvering positions. Thrusting will not occur at the

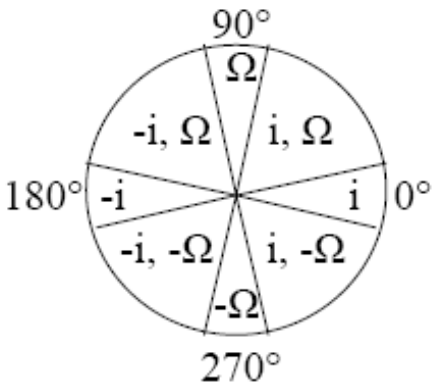

Figure 3: Positive Thrust

exact maneuvering spot, which may have an undesirable affect on the opposite orbital angle. The effect is so small, it can be easily compensated for at the other maneuvering point. All these points need to be considered, as far as singularities in the simulation is concerned.

In Non-planar maneuvers, addition to the four state equations mentioned in the planar maneuvers, two equations involving the ascending node and the inclination is used. The derivation of these equations are given in Appendix C. The final equations are given below.

$$\frac{di}{dt} = \sqrt{\frac{r}{\mu(1 + e \cdot \cos(\upsilon))}} \cos(\omega + \nu) \cdot (P_W + U_W) \tag{3.18}$$

$$\frac{d\Omega}{dt} = \sqrt{\frac{r(1 + e \cdot \cos(\upsilon))}{\mu}} \left[\frac{\sin(\omega + \nu)}{\sin(i) \cdot (1 + e(\cos(\nu))} \cdot (P_W + U_W) \right] \tag{3.19}$$

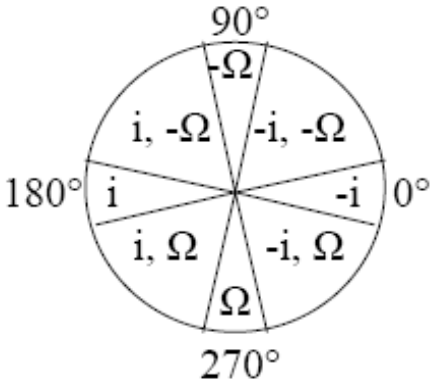

Figure 4: Negative Thrust

3.4 Controls Algorithm

The intention of the control system is to remove the error between the actual and desired states. A second order method can be constructed to more accurately reach a steady state, but still maintain simple derivations. A second order performance index is developed and a controls equations is derived from the same. The derivation is given in Appendix F and the final equation is given below.

$$x_o(K-1) = V_o wt + \frac{dV_o}{dt} \cdot \frac{wt^2}{2} \tag{3.20}$$

To apply this control theory, the orbital state equations are substituted into the performance index equation. The four state equations, the equations involving the ascending node,inclination and as well as energy as used as the desired performance index. Since the

out of plane maneuver contain the in-plane maneuver, separate controls algorithm is not required.

Now state equations used are,

$$Energy = \frac{V_S^2}{r} + \frac{V_R^2}{r} - \frac{\mu}{r} \tag{3.21}$$

$$\frac{dEnergy_o}{dt_o} = V_R \cdot (P_R + U_R) + V_S \cdot (P_S + U_S) \tag{3.22}$$

$$\frac{dr_o}{dt_o} = V_R \tag{3.23}$$

$$\frac{dV_{Ro}}{dt_o} = \frac{V_S^2}{r} - \mu\frac{1}{r^2} + P_R + U_R \tag{3.24}$$

$$\frac{d\theta_o}{dt_o} = \frac{rV_S}{\mu}\left[\begin{array}{c} \frac{\mu}{r^2} - \frac{\cos(\nu)}{e} \cdot (P_R + U_R) + \frac{\sin(\nu)(2+e\cdot\cos(\nu))}{e(1+e\cdot\cos(\nu))} \cdot (P_S + U_S) \\ + \frac{\sin(w+\nu)}{\tan(i)\cdot(1+e\cdot\cos(\nu))} \cdot (P_W + U_W) \end{array}\right] \tag{3.25}$$

$$\frac{d^2\nu_0}{dt_o^2} = -\frac{3V_R \cdot V_S}{r^2} + \frac{(P_S + U_S)}{r} \tag{3.26}$$

$$\frac{di_o}{dt_o} = \frac{\cos(w+\nu) \cdot U_W}{V_S} \tag{3.27}$$

$$\frac{d\Omega_o}{dt_o} = \frac{\sin(w+\nu)}{V_S \cdot \sin(i)} \cdot (P_W + U_W) \tag{3.28}$$

After applying the control theory for maintaining a constant orbit, the above equations become,

$$\delta(\frac{dEnergy_o}{dt_o}) = V_{R_1} \cdot (P_{R_1} + U_{R_1}) + V_{S_1} \cdot (P_{S_1} + U_{S_1}) \tag{3.29}$$

$$\delta(\frac{dr_o}{dt_o}) = V_{R_1} - V_{R_{traj}} \tag{3.30}$$

$$\delta(\frac{dV_{Ro}}{dt_o}) = \frac{V_S^2}{r_1} - \frac{V_{traj}^2}{r_{traj}} - \mu(\frac{1}{r_1^2} - \frac{1}{r_{traj}^2}) + P_{R_1} + U_{R_1} \tag{3.31}$$

$$\delta(\frac{d^2\nu_0}{dt_o^2}) = \frac{3V_{R_{traj}} \cdot V_{S_{traj}}}{r_{traj}^2} - \frac{3V_{R_1} \cdot V_{S_1}}{r_1^2} + \frac{(P_{S_1} + U_{S_1})}{r_1} \tag{3.32}$$

$$\delta(\frac{di_o}{dt_o}) = \frac{\cos(w_1 + \nu_1) \cdot U_{W_1}}{V_{S_1}} \tag{3.33}$$

$$\delta(\frac{d\Omega_o}{dt_o}) = \frac{\sin(w_1 + \nu_1)}{V_{S_1} \cdot \sin(i_1)} \cdot (P_{W_1} + U_{W_1}) \tag{3.34}$$

Now maintaining a constant orbit is just part of the mission, the actual mission would be to make sure that the satellite is as close to the one, which needs to be monitored. Now some of the equations where the controls is applied would involve not only the inputs from this satellite, but also the inputs of the other satellite, so that if not very near, at least it will maintain a constant distance, and look like a constellation.

For achieving the above, the above equations become,

$$\delta(\frac{dEnergy_o}{dt_o}) = V_{R_1} \cdot (P_{R_1} + U_{R_1}) + V_{S_1} \cdot (P_{S_1} + U_{S_1}) \tag{3.35}$$

$$\delta(\frac{dr_o}{dt_o}) = V_{R_2} - V_{R_1} \tag{3.36}$$

$$\delta(\frac{dV_{Ro}}{dt_o}) = \frac{V_{S_2}^2}{r_2} - \frac{V_{S_1}^2}{r_1} - \mu(\frac{1}{r_2^2} - \frac{1}{r_1^2}) + P_{R_2} + U_{R_2} - P_{R_1} - U_{R_1} \tag{3.37}$$

$$\frac{d\theta_o}{dt_o} = \frac{rV_{S_2}}{\mu} \begin{bmatrix} \frac{\mu}{r_2^2} - \frac{\cos(\nu_2)\cdot(P_{R_2}+U_{R_2})}{e_2} + \\ \frac{\sin(\nu_2)(2+e_2\cdot\cos(\nu_2))\cdot(P_{S_2}+U_{S_2})}{e_2(1+e_2\cdot\cos(\nu_2))} \\ + \frac{\sin(w_2+\nu_2)\cdot(P_{W_2}+U_{W_2})}{\tan(i_2)\cdot(1+e_2\cdot\cos(\nu_2))} \end{bmatrix} - \frac{rV_{S_1}}{\mu} \begin{bmatrix} \frac{\mu}{r_1^2} - \frac{\cos(\nu_1)\cdot(P_{R_1}+U_{R_1})}{e1} \\ + \frac{\sin(\nu_1)(2+e_1\cdot\cos(\nu_1))\cdot(P_{S_1}+U_{S_1})}{e_1(1+e_1\cdot\cos(\nu_1))} \\ + \frac{\sin(w_1+\nu_1)\cdot(P_{W_1}+U_{W_1})}{\tan(i_1)\cdot(1+e_1\cdot\cos(\nu_1))} \end{bmatrix} \tag{3.38}$$

$$\delta(\frac{d^2\nu_0}{dt_o^2}) = \frac{3V_{R_2} \cdot V_{S_2}}{r_2^2} - \frac{3V_{R_1} \cdot V_{S_1}}{r_1^2} + \frac{(P_{S_2} + U_{S_2})}{r_2} - \frac{(P_{S_1} + U_{S_1})}{r_1} \tag{3.39}$$

$$\delta(\frac{di_o}{dt_o}) = \frac{\cos(w_2 + \nu_2) \cdot U_{W_2}}{V_{S_2}} \tag{3.40}$$

$$\delta(\frac{d\Omega_o}{dt_o}) = \frac{\sin(w_2 + \nu_2)}{V_{S_2} \cdot \sin(i_2)} \cdot (P_{W_2} + U_{W_2}) \tag{3.41}$$

CHAPTER FOUR: SYSTEM CONFIGURATION

A system of solar cells, NORAD TLE, IMU with a magnetometer,torque coils as well as an on board computer are to be used to complete the Attitude Determination and Control System (ADCS) system.

The solar cells are used to determine a sun vector that would give the position of the sun relative to the satellite. This would be done by measuring the current through each of the solar cells. The maximum value that the cells can obtain by being directly in sunlight would be known. If they are not in direct sunlight, the current would be a percentage of this maximum value and from these percentages a sun vector can be calculated. These do not give the highest accuracy but it would still fall within the requirements needed.Since these would also be providing power, the solar cells would be being used to their highest potential.

Once we get maximum power, we can get maximum current, and then using this current, we get the Force using the Earth's Magnetic Field. Now depending on the position of the satellite, we have to make sure that we not only have the correct attitude, but also the Force for Orbit Maneuvers. We have to make sure that both these are done in an optimum manner. It can also sometimes happen that we might have to compromise one for the other, depending on the position of the satellite.

Celeriac elements given by NORAD (North American Aerospace Defense Command) in a two-line element (TLE) set format would be used to gather information of the satellite's

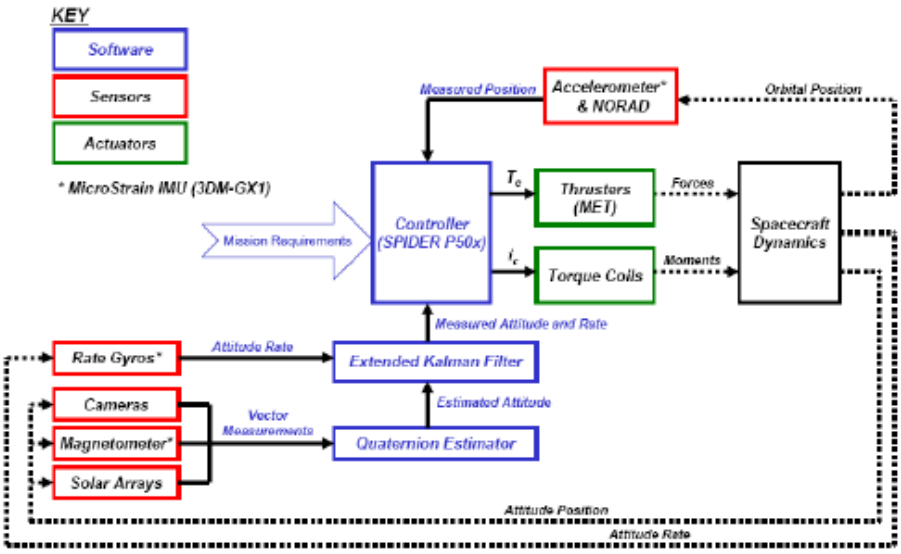

Figure 5: ADCS Block Diagram

position. These lines of data can be run through a program, also provided by NORAD, to give the satellite's latitude, longitude, altitude, velocity, look angles, right ascension, and declination.

The inertial measuring unit (IMU) would consist of a 3-axis accelerometer to measure the acceleration and a 3-axis gyroscope to measure the rotation (pitch, yaw, and roll) rates of the satellite. From these inputs, the deviation from the desired position would be calculated. A 3-axis magnetometer would also be added to this unit to measure the magnetic field in a particular direction. Torque coils would then be used as an active control system for the satellite. Three torque coils would be needed in order to have control of all three axes. Once they are in a magnetic field, the current that would be induced in the coils would push the coil to align with the magnetic field.

4.1 NORAD

NORAD (North American Aerospace Defense Command) will be used to track the satellite once in orbit. This is a free program and the data is given in a two-line element format. These lines of data consist of only certain characters including the numbers 0-9, letters A-Z, periods, spaces, and the plus and minus signs. These lines of data are run through a program, also provided free by NORAD, to give the satellite's latitude, longitude, altitude, velocity, look angles, right ascension, and declination.This allows for an affordable way to track the satellite.

4.2 Torque Coils

The key features that made torque coils the best option for attitude control of the satellite were size, weight and cost. Torque Coils allow full control of motion without the constraints and power requirements of external thrust. Several options of coils that meet our requirements were explored, and these will offer the best means for control to all aspects of movement for the satellite, as well as stability. By utilizing three coils, located on the X, Y, and Z axis, we will have complete control of mobility. Two Torque Coils will be mounted on each axis equidistant form the center of the mass but 180 degrees apart. Control current,both magnitude and direction will dictate the following control actions :

(a) Current in the opposite directions will be used for attitude control.

(b) Current in the same direction will be used for transition maneuvers.

The torque coils will be an integral aspect of ensuring the success of the satellite's mission. A list of the actions to be performed and use of the torque coils are as follows:

- Launch
- De-tumble - reduce rotation rates to near zero (from separation, fault)
- Attitude Acquisition - Find sun, Earth, Stars, etc. by sweeping
- Flight - normal operation such as pointing for science and monitoring thrust
- Formation Flight - propulsive maneuver for relative position change
- Communication - periodic pointing of antenna at Earth
- Safe - response to a fault, stable state in which to wait for commands

The addition of a safe mode is a great asset in the case of loss of communication was to occur. The default would automatically position the satellite

4.3 Solar Cells

The solar cells will be used to generate power; therefore, reception of sun light is important for positioning and maximizing power generation. The location of the sun vector will be determined by using the current outputs from the solar cells and a computer algorithm. The current outputs need to be coming from at least two perpendicular sources (cells) to determine the exact direction and angle the sunlight is hitting the satellite. This will require cooperation between Power and ADCS since the solar panels and current sensors will be used by ADCS to calculate the location of the sun.

When the sun hits directly perpendicular to a solar cell, the current reading would be at the maximum value (which will be known based on the cell's characteristics and the solar intensity). Thus, if the sun is not straight-on, the current received from each cell will be a percentage of the maximum value. This value will be used to determine an angle that the sun is from the cells normal vector. However, using data from only one cell will not determine the direction of the sun, only its angle from the normal vector. In 3-D space this creates a cone of possible sun vectors for each solar cell. When two such cones are derived for perpendicular solar cells, there will be only one pair of vectors (one vector from each cell) which are parallel.

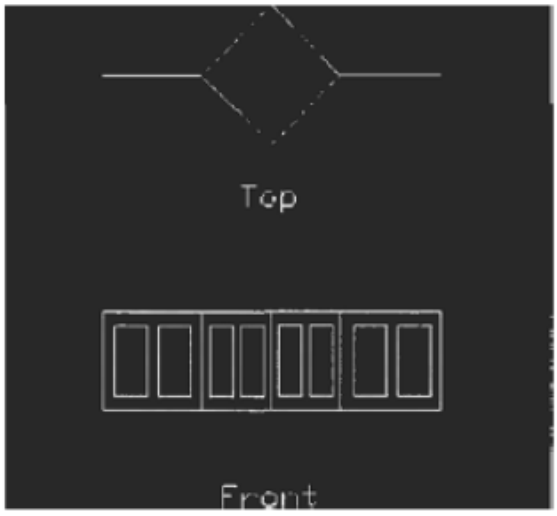

Figure 6: Solar Cells

ADCS does not require any additional parts to determine the sun vector since it will be using components that the Power subsystem is already implementing into the satellite. There will be a current sensor from each of the sides and ADCS will take these inputs and run them through the above algorithm to determine the sun vector location. The algorithm will be written in a language which will have ease of use and adaptability.

CHAPTER FIVE: MATHEMATICAL MODELING

5.1 Satellite Model

5.1.1 Dynamics

A satellite can be regarded as an ideal rigid body. The dynamic model of the satellite is derived using a Newton-Euler formulation, where the applied torque change is related to the angular momentum. The satellite model is:

$$I\dot{\omega} + S(\omega_{ib}^{b})(I\omega_{ib}^{b}) = \tau_{grav}^{b} + \tau_{m}^{b} \tag{5.1}$$

where I is the moment of inertia matrix, ω_{ib}^{b} is the angular velocity of the body frame with respect to the inertial frame and τ^{b} are the torques acting on the satellite also decomposed in body frame, $S(\omega_{ib}^{b})$ is the skew matrix of ω_{ib}^{b}, τ_{grav}^{b} is the gravitational torque working on the satellite body. τ_{m}^{b} is the torque applied by the magnetic coils.

The angular velocity of the satellite relative to the inertial frame is expressed in the body frame according to (5.2).

$$\omega_{ib}^{b} = \omega_{io}^{b} + \omega_{ob}^{b} = R_{o}^{b}\omega_{io}^{o} + \omega_{ob}^{b} \tag{5.2}$$

Where ω_{io} is the angular velocity in orbital frame with respect to the inertial frame,ω_{ob} is the angular velocity in body frame with respect to the orbital frame,ω_{ib} is the angular velocity in body frame with respect to the inertial frame,R_{o}^{b} is the rotation matrix.

5.1.2 Kinematics

The kinematics describes the satellite's orientation in space and is derived by integration of the angular velocity. The angular velocity of the satellite model can be described by unit quaternions and is formulated in (5.3).Detail Derivation is given in Appendix A.7.

$$\dot{\mathbf{q}} = \frac{1}{2}\begin{bmatrix} \eta I_{3x3} + S(\boldsymbol{\varepsilon}) \\ -\boldsymbol{\varepsilon}^{T} \end{bmatrix} \boldsymbol{\omega}_{ob}^{b} \tag{5.3}$$

To find the rotation velocity for the body frame relative to the orbit frame, see equation (5.4)

$$\boldsymbol{\omega}_{ob}^{b} = \boldsymbol{\omega}_{ib}^{b} - R_{o}^{b}\boldsymbol{\omega}_{io}^{b} \tag{5.4}$$

5.2 Magnetic Torque

The actuator used on FUNSAT is three pairs of magnetic coils. The basic idea behind the magnetic coils is based on reacting together with the Earth's magnetic field. The magnetic coil produces a magnetic dipole when currents flow through it's windings, which is proportional to the ampere-turns and the area enclosed by the coil. The actuator effect of the magnetic torquers will then have as purpose to react with the Earth's magnetic field to place the satellite in selected attitude and maneuver direction. The measurements of the geomagnetic field is done by magnetometers. The torque generated by the magnetic coils can be modeled as:

$$\tau_m^b = m^b B^b \tag{5.5}$$

the magnetic dipole moment generated by the coils m^b is shown in (5.5) and $B^b = \begin{bmatrix} B_x^b & B_y^b & B_z^b \end{bmatrix}^T$ is the local geomagnetic field vector, relative to the satellite,

$$m^b = m_x^b + m_y^b + m_z^b = \begin{bmatrix} N_x i_x A_x \\ N_y i_y A_y \\ N_z i_z A_z \end{bmatrix} = \begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} \tag{5.6}$$

where N_k is number of windings in the magnetic coil, i_k is the coil current and A_k is the span area of the coil. An easy way of representing the magnetic torque is then:

$$\tau_m^b = S(m^b)B^b = \begin{bmatrix} B_z^b m_y - B_y^b m_z \\ B_x^b m_z - B_z^b m_x \\ B_y^b m_x - B_x^b m_y \end{bmatrix} \tag{5.7}$$

One of the effects of using magnetic torques is that they will contribute to the measurements of the Earth's magnetic field.

5.3 Gravitational Torque

A satellite orbiting Earth is affected by the gravitational field (actually it is affected by every object in the universe but only the Earth's gravitational field is taken into consideration). The torque is derived is used for passive stabilization with a gravity boom that will utilize this effect. When assuming homogeneous mass distribution of the Earth, the gravity gradient is derived as:

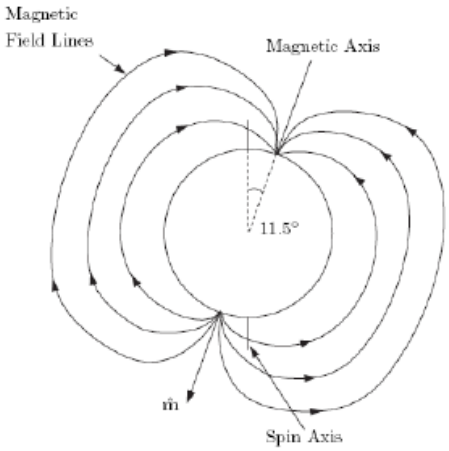

Figure 7: Earth's Magnetic Field

$$\tau_{grav} = 3 \cdot \omega_o^2 \cdot c_3^b \cdot I \cdot c_3^b \tag{5.8}$$

where,

$$\omega_o^2 = \frac{\mu}{R_o^3} \tag{5.9}$$

5.4 Earth's Magnetic Field

The Earth's magnetic field generally resembles the field around a magnetized sphere,

or a tilted dipole seen in Figure 7. As of 1999, the dipole axis was tilted approximately 11.5 from the spin axis, and drifting approximately 0.2 /yr. Its strength at the Earth's surface varies from approximately 30000nT near the equator to 60000nT near the poles. Further, there exists a low magnetic intensity field at approximately 25 S and 45 W known as the Brazilian Anomaly. A high exists at 10 N and 100 E, and the two of these together suggest that not only is the dipole axis tilted, but it does not quite pass through the center of the Earth.

5.4.1 Modeling the magnetic field

The accepted model for Earth's magnetic field is the International Geomagnetic Reference Field(IGRF), put forth by the International Association of Geomagnetism and Aeronomy (IAGA).

The IGRF is essentially a set of Gaussian coefficients, g_n^m and h_n^m, that are put forth every 5 years by IAGA for use in a spherical harmonic model. At each of these epoch years, the group considers several proposals and typically adopts a compromise that best fits the data available. The coefficients for a given epoch year are referred to by IGRF and then the year, as in IGRF2000. The model includes both the coefficients for the epoch year and secular variation variables, which track the change of these coefficients in nanoTessla per year. These secular variation coefficients are used to extrapolate the Gaussian coefficients to the date in question. Once data becomes available about the actual magnetic field for a given epoch year, the model is adjusted and becomes the Definitive Geomagnetic Reference Field, or DGRF.2

Typically the IGRF consists of 120 coefficients for each epoch year, with 80 secular variation coefficients. However, due to unprecedented geomagnetic data available, the IAGA released a new set of values for IGRF2000 in July 2003 in the 9th-generation IGRF. This new model expanded to increase the precision of the coefficients to one-tenth of a nanoTessla (up from one nanoTessla), and increased the number of coefficients to degree 13.

5.4.2 Random variation in the magnetic field

The many variations in the magnetic field around the Earth that can not be accurately modeled due to their random nature, but a brief overview is provided here. It is important to be aware of these variations when considering spacecraft design. First, there are temporal variations that occur about every 27 days when the active solar area of the Sun faces the Earth. These variations last between seconds and days, and are particularly bad when the Earth is near equinox in March-April and September-October.

The second type of variation is diurnal variations which occur due to partical movement within the ionosphere. These polar (auroral) and equatorial electrojets can have significant impact on the magnetic field. The auroral electrojet can cause changes on the order of 1000nT to 1500nT at the Earth's surface, while the equatorial electrojets cause disturbances on the order of 220nT between the altitudes of 96 and 130 kilometers.

The last variation is a result of magnetic storms, which occur during solar flares. Solar flares are very closely related to Sun activity and so magnetic storms generally follow the same 27 day pattern seen in the general temporal variations, although they occur less frequently.

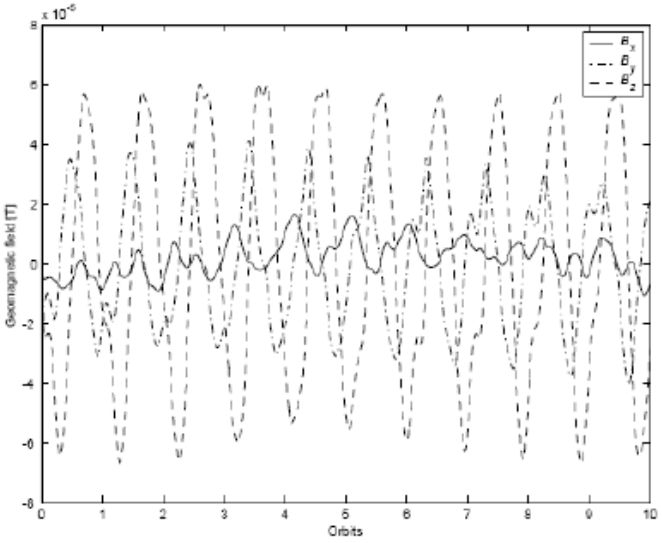

Figure 8: Local geomagnetic field Bo, the IGRF2000 coeffcients

During the first phase of the magnetic storm the effect is around 50nT, however, during the main phase, which lasts several hours, the variation is as much as 400nT.

5.4.3 Mathematics of the IGRF

According to the physics, the magnetic field, B, is defined as the negative gradient of the scalar potential function V, such that

$$B = -devV \tag{5.10}$$

Although a simple dipole model gives a good approximation of the geomagnetic field, it can be modeled more closely using a spherical harmonic model of the scalar potential as given in Equation (5.11). This is the equation about which the IGRF is based.

$$V(r,\theta,\phi) = a\sum(\frac{a}{r})^{n+1}\sum(g_n^m\cos m\phi + h_n^m\sin m\phi)P_n^m(\theta) \tag{5.11}$$

Here a, is the reference radius of the Earth (a = 6371.2 km), r, θ,and ϕ are the geocentric coordinates (r is the radius in kilometers, θ is the co-latitude and ϕ is the longitude. The

coefficients g_n^m and h_n^m are the gaussian coefficients put forth by the IAGA for the IGRF, and $P_n^m(\theta)$ represents the Schmidt quasi-normalized associated Legendre functions of degree n and order m. The input to this function is actually cosθ, rather than θ, but this has been dropped for abrevity.

5.4.3.1 Derivatives of the scalar potential function

The magnetic field strength is calculated by taking the partial derivative of V, and the following equations are derived using the same,

$$B_r = -\frac{V}{r} = \sum (\frac{a}{r})^{n+2}(n+1) \sum (g_n^m \cos m\phi + h_n^m \sin m\phi) P_n^m(\theta) \tag{5.12}$$

$$B_\theta = -\frac{1}{r}(\frac{V}{\theta}) = -\sum (\frac{a}{r})^{n+2} \sum (g_n^m \cos m\phi + h_n^m \sin m\phi) \frac{P_n^m(\theta)}{\theta} \tag{5.13}$$

$$B_r = -\frac{1}{r \sin\theta}(\frac{V}{\phi}) = \sum (\frac{a}{r})^{n+2}(n+1) \sum m(-g_n^m \sin m\phi + h_n^m \cos m\phi) P_n^m(\theta) \tag{5.14}$$

where B_r, B_θ and B_ϕ represent the field strength in local tangential coordinates, and the other variables defined as before.

5.4.3.2 Legendre polynomials

In order to calculate the magnetic field, one must first calculate the associated Legendre polynomials. Legendre polynomials are a set of orthogonal polynomials that also satisfy the zero mean condition. The following equations for the Legendre polynomials and associated Legendre polynomials come from Schaub.

Regular Legendre polynomials $\mathrm{P}_n(v)$ are calculated to satisfy the following equations :

$$(1 - 2vx^2 + x^2)^{-1/2} = \sum P_n(v)x^n \tag{5.15}$$

when solved, this becomes Rodrigues' formula:

$$P_n(v) = \frac{1}{2^n n!}(\frac{d}{dv})^n(v^2 - 1) \tag{5.16}$$

For reference, this equation yields the following first four Legendre polynomials:

$$P_0(v) = 1 \tag{5.17}$$

$$P_1(v) = v \tag{5.18}$$

$$P_2(v) = (3v^2 - 1)/2 \tag{5.19}$$

$$P_3(v) = (5v^3 - 3v)/2 \tag{5.20}$$

5.4.3.3 Associated Legendre polynomials

The above Legendre polynomials are related to the associated Legendre polynomials through the following equation:

$$P_{n,m}(v) = (1 - v^2)^{1/2m} \frac{d^m}{dv^m}(P_n(v)) \tag{5.21}$$

Note that for all m greater than n, the associated Legendre polynomial is equal to zero. Following are the associated Legendre polynomials through the third degree, which can be verified using equations (5.22), (5.23) and (5.24).

$$P_{1,1}(v) = \sqrt{1 - v^2}\ P_{2,1}(v) = 3v\sqrt{1 - v^2}\ P_{3,1}(v) = \frac{3}{2}\sqrt{1 - v^2}(5v^2 - 1) \tag{5.22}$$

$$P_{2,2}(v) = 3(1 - v^2) \qquad P_{3,2}(v) = 15v^2(1 - v^2) \tag{5.23}$$

$$P_{3,3}(v) = 15v(1-v^2)^{3/2} \tag{5.24}$$

In case of the magnetic field model, v is replaced by cosθ, such that the above equations become,

$$P_{1,1}(v) = \sin\theta \; P_{2,1}(v) = 3\cos\theta\sin\theta \; P_{3,1}(v) = \frac{3}{2}\sin\theta(5\cos^2\theta - 1) \tag{5.25}$$

$$P_{2,2}(v) = 3\sin^2\theta \qquad\qquad P_{3,2}(v) = 15\cos\theta\sin\theta \tag{5.26}$$

$$P_{3,3}(v) = 15\sin^3\theta \tag{5.27}$$

The formulas in equations (5.25), (5.26) and (5.27) represent traditional associated Legendre polynomials that have not been normalized in any way, and they are represented by $\mathrm{P}_{n,m}$. There are two commonly used normalizations that must also be accounted for as described by Jacobs. The first is the Gaussian normalized associated Legendre polynomials, $\mathrm{P}^{n,m}$. These are related to the non-normalized set by the equation

$$P^{n,m} = \frac{2^n!(n-m)!}{(2n!)}P_{n,m} \tag{5.28}$$

In magnetic field modeling, the Schmidt quasi (semi) - normalized form, P_n^m is used, related by the equation

$$P_n^m = \left[\frac{2(n-m)!}{(n+m)!}\right]^{1/2} P_{n,m} \tag{5.29}$$

The recursive formulas for the Gaussian normalized associated Legendre polynomials are used in this model, and the relationship between the Gaussian normalized and Schmidt quasi-normalized polynomials is explained in detail in the next section.

5.4.3.4 Schmidt quasi-normalization

The relationship between the Gaussian normalized associated Legendre polynomials and the Schmidt quasi-normalized is given as

$$P_n^m = S_{n,m} P^{n,m} \tag{5.30}$$

where $P^{n,m}$ is the Gaussian associated Legendre polynomial, and $S_{n,m}$ is defined by

$$S_{n,m} = \left[\frac{(2-\delta_m^0)(n-m)!}{(n+m)!}\right]^{1/2} \frac{(2n-1)!!}{(n-m)!} \tag{5.31}$$

The Kronecker delta is defined as $\delta_i^j = 1$ if $i = j$ and $\delta_i^j = 0$ otherwise. Further, (2n-1)!! $=1\cdot3\cdot5\cdot\cdots(2n\text{-}1)$.

Due to the fact that that these normalization values can be calculated irrespective of the value of θ at which the associated Legendre polynomials are calculated, it is much simpler to instead normalize the model coefficients, g_n^m and h_n^m, recognizing that

$$g^{n,m} = S_{n,m} g_n^m \tag{5.32}$$

$$h^{n,m} = S_{n,m} h_n^m \tag{5.33}$$

This allows the normalization to only occur once, and then use those new coefficients can be used to compute the field strength at whatever location is desired.

5.4.3.5 Final Result

In order for the results of equations (5.12), (5.13) and (5.14) to be effective in satellite work, they must be converted to geocentric inertial components, using the following equation:

$$B_x = (B_r \cos\delta + B_\theta \sin\delta)\cos\alpha - B_\phi \sin\alpha \tag{5.34}$$

$$B_y = (B_r \cos\delta + B_\theta \sin\delta)\sin\alpha + B_\phi \cos\alpha \tag{5.35}$$

$$B_z = (B_r \sin\delta + B_\theta \cos\delta) \tag{5.36}$$

Where δ is the latitude measured positive North from the equator, and α is the local sidereal time of the location in question.

5.5 Sensors

There are many types of sensors available for observing the attitude of a satellite. In this thesis the sensors used are two star trackers, a sun sensor and a magnetometer.

5.5.1 Sun Sensor

A sun sensor is an instrument which measures the direction from the satellite to the sun. The direction to the sun can be measured in two different ways, both of them relying on photocells. The rst one, the analog sun sensor, measures the intensity of the sun, and the second one, the digital sun sensor, uses a pattern where different photocells is exposed depending on the direction of the sun.

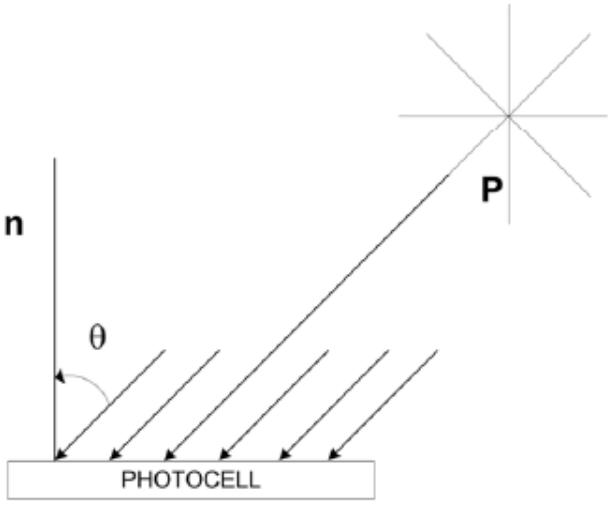

Figure 9: Analog Sun Sensor

5.5.1.1 Analog sun sensor

As mentioned above the analog sun sensor, also called cosine sensor, measures the sun's intensity. More precisely it measures the energy flux through the surface area of photocell. The energy flux through a photocell is given by :

$$E = P \cdot ndA \tag{5.37}$$

where P is the pointing vector against the sun, n is a vector perpendicular to the cell, dA is the surface area of the cell and E is the energy flux. Equation (5.37) can be rewritten

into :

$$E = \|P\| \, \|n\| \cos\theta dA \tag{5.38}$$

where £ is the angle of the pointing vector P, and therefore is the angle from the perpendicular vector n toward the sun. The energy flux, E, is proportional to the electric current generated in the photocell. Therefore the easiest way to measure the angle of the pointing vector is to measure the current with :

$$I_c = I_{\max} \cos\theta \tag{5.39}$$

where I_c is the measured current, and $I_{\max}$ is the maximum current generated in the photocell.

5.5.1.2 Digital sun sensor

A digital sun sensor is built up of a pattern of photocells. The photocells are placed inside an installation that restricts which photocells that are illuminated and make this depended on the direction of the sun. This is illustrated in Figure 10. The photocells can be put into many different patterns. One example of this is a binary pattern, but it is more usual to use a Gray-coded pattern. From both of these patterns the angle of the sun can be found in the output of the sensors, since the photocells that is illuminated generate a higher energy

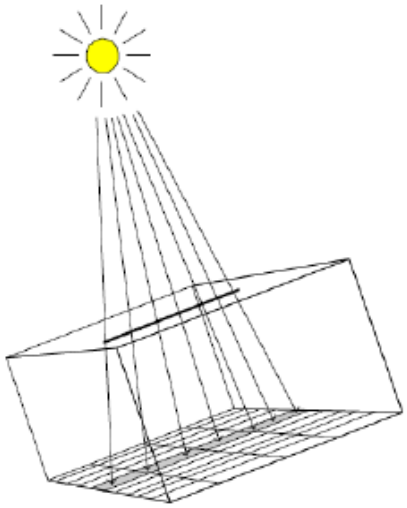

Figure 10: Digital Sun Sensor

level, either higher voltage or higher current, than the sensors that is in the shadow. This can in turn be converted into digital ones and zeros, which gives the angle of the sun either in binary or Gray scale.

5.5.1.3 The measured sun vector

The sun sensor measures the sun vector in body frame. Without noise the sun vector will be given by (5.40). With noise the measured sun vector can be given as

$$v_s^b = R_o^b v_s^o + w \tag{5.40}$$

where w is an additive noise term, or it can be given by

$$v_s^b = R_o^b(q_{w/noise})v_s^o \tag{5.41}$$

where $q_{w/noise}$ is the quaternion with noise. The rotation $R_o^b(q_{w/noise})$ can be replaced with $R_o^b(\theta_{w/noise})$ if the attitude is given in Euler angles. Whether to use equation (5.40) or (5.41), depends on how the noise is given.

5.5.2 Magnetometer

A magnetometer is an instrument which measures the flux density of the magnetic field it is placed in. A three axis magnetometer placed inside a satellite, will measures the geomagnetic intensity and direction surrounding the satellite. In a low orbiting satellite this can be used as a low cost, low weight, and reliable attitude sensors, with an accuracy of 0.5 to 3 degrees and a weight of 0.3 to 1.2 kg. It can also be used to calculate the control input when using coils as actuator, and to estimate the orbit.

The most common magnetometer used in space is the flux-gate magnetometer, where each axis has a sensor. Each sensor consists of a transformer wound around a core of high-permeability material, illustrated in Figure 11. By exciting the primary winding with a high frequency , this will induce a frequency on the secondary winding, where the amplitude and phase of the even harmonics are linearly proportional to the ambient magnetic field. For one

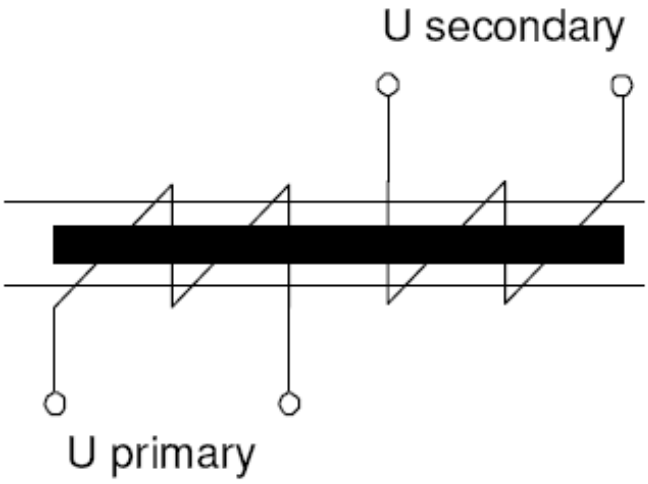

Figure 11: Magnetometer

axis this can give:

$$b\alpha\hat{U}_{\text{sec}} \tag{5.42}$$

where $\hat{\text{U}}_{\text{sec}}$ is the voltage amplitude generated on the secondary winding. This can be rewritten into

$$b\alpha\ K_{prop}\hat{U}_{\text{sec}} \tag{5.43}$$

where K_{prop} is the proportionality constant, and b is the ambient magnetic field. K_{prop} is determined by the material in the core, and the number of windings on the primary and

secondary winding. When the three sensors are mounted perpendicular to each other, it will be possible to use (5.43), to calculate the magnetic field vector:

$$B^b_{meas} = \begin{bmatrix} b^b_1 \\ b^b_2 \\ b^b_3 \end{bmatrix} = \begin{bmatrix} K_{prop}\hat{U}_{\sec 1} \\ K_{prop}\hat{U}_{\sec 2} \\ K_{prop}\hat{U}_{\sec 3} \end{bmatrix} \tag{5.44}$$

where b^b_1 is the ambient magnetic field of sensor 1 measured in body frame (it has been assumed that the sensor are aligned with the body of the satellite), and $\hat{U}_{\sec 1}$ is the voltage amplitude on the secondary winding of the rst sensor. Mathematically B^b_{meas} will, in a noise free environment, be equal to

$$B^b_{meas} = R^b_o(q)B^o \tag{5.45}$$

where B^o is the magnetic field given in orbit frame. With noise B^o will be equal to

$$B^b_{meas} = R^b_o(q)B^o + w \tag{5.46}$$

where w is the noise vector.

CHAPTER SIX: KALMAN FILTER

A major characteristic of differential correction and sequential-batch correction is that the converged state estimate and covariance matrix are based on processing a batch of data, spread over sometime interval which may be minutes, hours,days or even weeks. These techniques do not lend themselves to problems in which forces are incompletely modeled. For example, variable atmospheric drag due to changing solar flux and geomagnetic activity causes slow and fast variations in the orbit, respectively. A second characteristic is that the estimate is always associated with a particular epoch. Thus you may have to predict from the epoch state to a new time. A more significant issue is how to propagate the state and covariance matrix over the time interval to provide the state accurate information at the new epoch. Covariance propagation is one application in which the quality of that process-noise model. Kalman filter solve some of these problems but also introduce new ones.

The Kalman filter is essentially a set of mathematical equations that implement a predictor-corrector type estimator that is optimal in the sense that it minimizes the estimated error covariance—when some presumed conditions are met. Since the time of its introduction, the Kalman filter has been the subject of extensive research and application, particularly in the area of autonomous or assisted navigation. This is likely due in large part to advances in digital computing that made the use of the filter practical, but also to the relative simplicity and robust nature of the filter itself. Rarely do the conditions necessary

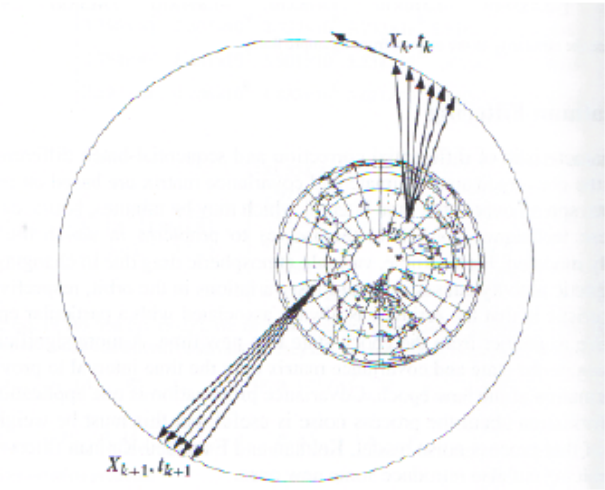

Figure 12: Observations from multiple passes

for optimality actually exist, and yet the filter apparently works well for many applications in spite of this situation.

The Kalman filter has been proven to be extremely useful for problems in which data streams continuously, such as in as an attitude-estimation system or with continuous observations from a space-based sensor, but these techniques have been difficult to use for orbit determination. Many investigators conclude that filters are difficult to "tune"; thus they often ignore new data and diverge-a process known as smugness.

In the Figure 12, we want to use new observations at time t_{k+1} and we have already calculated the best estimate of the converged state and the covariance ($\hat{X}_k$ and $\hat{P}_k$) from data at the original epoch, t_k.

This next section describes the filter in its original formulation (Kalman 1960) where the

measurements occur and the state is estimated at discrete points in time.

6.1 Estimation of the Process

The Kalman filter addresses the general problem of trying to estimate the state x ε R^n of a discrete-time controlled process that is governed by the linear stochastic difference equation

$$x_k = Ax_{k-1} + Bu_k + w_{k-1} \tag{6.1}$$

with the measurement z ε R^n that is

$$z_k = Hx_k + v_k \tag{6.2}$$

The random variables w_k and v_k represent the process and measurement noise (respectively). They are assumed to be independent (of each other), white, and with normal probability distributions

$$p(w) \sim N(0, Q) \tag{6.3}$$

$$p(w) \sim N(0, R) \tag{6.4}$$

In practice, the process noise covariance and measurement noise covariance matrices might change with each time step or measurement, however here we assume they are constant.

The $n\times$ n matrix A in the difference equation equation (6.1) relates the state at the previous time step $k-1$ to the state at the current step k, in the absence of either a driving function or process noise. Note that in practice A might change with each time step, but here we assume it is constant. The $n \times l$ matrix B relates the optional control input u ε R^n to the state x. The mxn matrix H in the measurement equation (6.2) relates the state to the measurement z_k. In practice might change with each time step or measurement, but here we assume it is constant

6.2 Computational Origins of the Filter

We define $\hat{x}_k^- \varepsilon R^n$ (note the "super minus") to be our a priori state estimate at step k given knowledge of the process prior to step k, and to be our a posteriori state estimate at step k given measurement z_k. We can then define a priori and a posteriori estimate errors as

$\mathrm{e}_k^- \equiv x_k - \hat{x}_k^-$

$\mathrm{e}_k \equiv x_k - \hat{x}_k$

The a priori estimate error covariance is then

$$P_k^- = E\left[e_k^- e_k^{-T}\right] \tag{6.5}$$

and the a posteriori estimate error covariance is

$$P_k = E\left[e_k e_k^T\right] \tag{6.6}$$

In deriving the equations for the Kalman filter, we begin with the goal of finding an equation that computes an a posteriori state estimate $\hat{x}_k$ as a linear combination of an a priori estimate $\hat{x}_k^-$ and a weighted difference between an actual measurement z_k and a measurement prediction $H\hat{x}_k^-$ as shown below in equation (6.7). Some justification for equation (6.7) is given in "The Probabilistic Origins of the Filter" found below

$$\hat{x}_k = \hat{x}_k^- + K(z_k - H\hat{x}_k^-) \tag{6.7}$$

The difference $(z_k - H\hat{x}_k^-)$ in equation (6.7) is called the measurement innovation, or the residual. The residual reflects the discrepancy between the predicted measurement $H\hat{x}_k^-$ and the actual measurement z_k . A residual of zero means that the two are in complete agreement.

The nxm matrix K in equation (6.7) is chosen to be the gain or blending factor that minimizes the a posteriori error covariance equation (6.6). This minimization can be accomplished by first substituting equation (6.7) into the above definition for e^k, substituting that into equation (6.6), performing the indicated expectations, taking the derivative of the trace of the result with respect to K, setting that result equal to zero, and then solving for K. One form of the resulting K that minimizes equation (6.6) is given by

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \tag{6.8}$$

$$K_k = \frac{P_k^- H^T}{(H P_k^- H^T + R)} \tag{6.9}$$

Looking at equation (6.8) we see that as the measurement error covariance approaches zero, the gain K weights the residual more heavily. Specifically,

$$\lim K_k = H^{-1} \tag{6.10}$$

On the other hand, as the a priori estimate error covariance approaches zero, the gain K weights the residual less heavily. Specifically,

$$\lim K_k = 0 \tag{6.11}$$

Another way of thinking about the weighting by K is that as the measurement error covariance approaches zero, the actual measurement z_k is "trusted" more and more, while the predicted measurement $H\hat{x}_k^-$ is trusted less and less. On the other hand, as the a priori estimate error covariance P_k^- approaches zero the actual measurement z_k is trusted less and less, while the predicted measurement $H\hat{x}_k^-$ is trusted more and more.

6.3 Probabilistic Origins of the Filter

The justification for equation (6.7) is rooted in the probability of the a priori estimate $\hat{x}_k^-$ conditioned on all prior measurements z_k (Bayes' rule). For now let it suffice to point out that the Kalman filter maintains the first two moments of the state distribution,

$$E[x_k] = \hat{x}_k \tag{6.12}$$

$$E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T] = P_k \tag{6.13}$$

The a posteriori state estimate equation (6.7) reflects the mean (the first moment) of the state distribution— it is normally distributed if the conditions of equation (6.3) and equation (6.4) are met. The a posteriori estimate error covariance equation (6.7) reflects the variance of the state distribution (the second non-central moment). In other words,

$$p(x_k \mid z_k) \tilde{} N(E[x_k], E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T]) \tag{6.14}$$

$$p(x_k \mid z_k) = N(\hat{x}_k, P_k) \tag{6.15}$$

6.4 Discrete Kalman Filter Algorithm

We will begin this section with a broad overview, covering the "high-level" operation of one form of the discrete Kalman filter. After presenting this high-level view, we will narrow the focus to the specific equations and their use in this version of the filter.

The Kalman filter estimates a process by using a form of feedback control: the filter estimates the process state at some time and then obtains feedback in the form of (noisy) measurements. As such, the equations for the Kalman filter fall into two groups: time update equations and measurement update equations. The time update equations are responsible for projecting forward (in time) the current state and error covariance estimates to obtain the a priori estimates for the next time step. The measurement update equations are responsible for the feedback i.e. for incorporating a new measurement into the a priori estimate to obtain an improved a posteriori estimate.

The time update equations can also be thought of as predictor equations, while the measurement update equations can be thought of as corrector equations. Indeed the final estimation algorithm resembles that of a predictor-corrector algorithm for solving numerical problems as shown below

The specific equations for the time and measurement updates are presented below in 6.16 and 6.17.

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_k \tag{6.16}$$

$$P_k^- = AP_{k-1}A^T + Q \tag{6.17}$$

Again notice how the time update equations (6.16) project the state and covariance estimates forward from time step $k-1$ to step k. A and B are from equation (6.1), while Q is from equation (6.3).

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} \tag{6.18}$$

$$\hat{x}_k = \hat{x}_k^- + K(z_k - H\hat{x}_k^-) \tag{6.19}$$

$$P_k = (1 - K_k H)P_k^- \tag{6.20}$$

The first task during the measurement update is to compute the Kalman gain, K_k. Notice that the equation given here as equation (6.11) is the same as equation (6.8). The next step is to actually measure the process to obtain z_k, and then to generate an a posteriori state estimate by incorporating the measurement as in equation (6.12). Again equation (6.12) is simply equation (6.7) repeated here for completeness. The final step is to obtain an a posteriori error covariance estimate via equation (6.13).

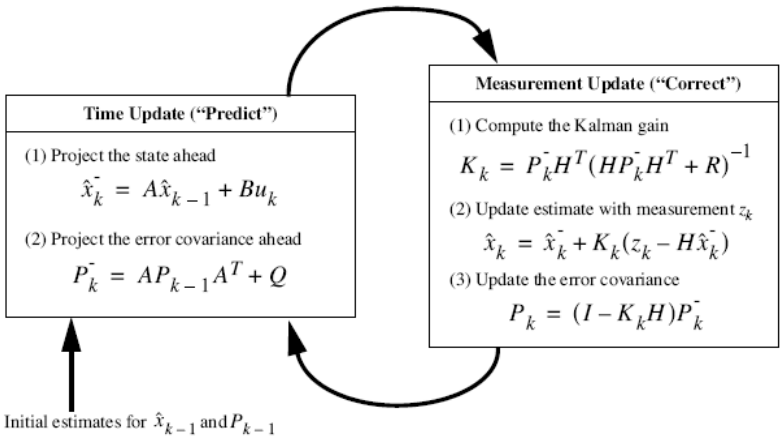

Figure 13: Complete Picture Working of the Kalman Filter

After each time and measurement update pair, the process is repeated with the previous a posteriori estimates used to project or predict the new a priori estimates. This recursive nature is one of the very appealing features of the Kalman filter—it makes practical implementations much more feasible than (for example) an implementation of a Wiener filter (Brown and Hwang 1996) which is designed to operate on all of the data directly for each estimate. The Kalman filter instead recursively conditions the current estimate on all of the past measurements. Figure 13 offers a complete picture of the operation of the filter, combining the high-level diagram of Figure 12 with the equations 6.1 and 6.2.

CHAPTER SEVEN: CONTROLLER DESIGN

7.1 LQR Problem Statement

The basic optimization problem considered is finding state-feedback control law of the form $u = -Kx$ that minimizes a performance measure of the form

$$V = \int (x'Qx + u'Ru)dt + x'(T)Mx(t) \tag{7.1}$$

where the system dynamics are given by

$$x = Ax + Bu \tag{7.2}$$

and Q and M are typically positive-semidefinite matrices, R is a positive-definite matrix, x is an n-dimensional state vector, and u is an m-dimensional input vector. In this problem we consider the final time T fixed, but the final state $x(T)$ free.

It is assumed that the state equations have been written so that the state x represents an incremental state. Thus, the control goal is to keep the state x as close to the state $x_r = 0$ as possible. The term $x'Qx$ in 7.1 is then measure of control accuracy, the term $u'Ru$ is a measure of control effort, and the term $x'(T)Mx(T)$ is a measure of terminal control accuracy. A control problem where the object is to maintain the state close to the

zero state is referred to as a regulator problem, hence the optimization problem is called as linear-quadratic-regulator problem.

7.2 LQR Solution :Matrix Ricati Solution

For the LQR problem we have the $l(x,u,t) = x'Qx + u'Ru$, $f(x,u,t) = Ax + Bu$, and $m(x) = x'Mx$. The matrices A,B,Q,R and M constitute the input data to the LQR problem and in general may all be time-varying i.e. depend explicitly on t. For convenience, this dependency is suppressed in the sequel. Most of the results in this section are valid for the general time-varying case. Indeed, this is an important advantage of the dynamic-programming solution to linear-quadratic problems, as opposed to frequency-domain approaches. Results are valid only in the time-invariant case will be explicitly noted.

The first step in obtaining a solution to the LQR problem is the minimization of

$$x'Qx + u'Ru + \left[\frac{\partial V^*}{\partial x}\right]' (Ax + Bu) \tag{7.3}$$

with respect to u. This minimization may be done by setting the gradient to the zero vector. When R is positive definite, this necessary condition for a minimal point is also a sufficient condition. Setting the gradient equal to zero and solving for u yields

$$u^* = \frac{1}{2} R^{-1} B' \frac{\partial V^*}{\partial x} \tag{7.4}$$

Now it is known that an integral-quadratic form evaluated for a linear system is a quadratic form in the initial state of the system, hence, it is reasonable to assume that

$$V^*(x,t) = x'P(t)x \tag{7.5}$$

where $P(t)$ is symmetric. The gradient of V* is then $2P(t)x$. If V^* and its gradient are substituted back in the equation, we obtain after some matrix manipulations

$$-x'Px = x'[A'P + PA + Q - PBR^{-1}B'P]x \tag{7.6}$$

with the boundary condition

$$V^*(x,t) = x'P(t)x = x'Mx \tag{7.7}$$

Now we have used the matrix identity $2x'PAx = x'(A'P + PA)x$ in deriving the above equation. Since the above equations are true for all x,we obtain the following matrix Ricati Equation and the final boundary value for $P(t)$

$$-P = A'P + PA + Q - PBR^{-1}B'P \tag{7.8}$$

$$P(T) = M \tag{7.9}$$

Finally the optimal state-feedback control law is given by

$$u^* = -K(t)x \tag{7.10}$$

$$K(t) = R^{-1}B'P(t) \tag{7.11}$$

7.3 Linearization of the Satellite Equations

We need to linearize all the equations obtained from the mathematical modeling, which makes the satellite easier to analyze, as well as implement the linear controller.

7.3.1 Kinematics

The kinematics of the satellite is described in as

$$\dot{\mathbf{q}} = \frac{1}{2}\begin{bmatrix} \eta I_{3x3} + S(\boldsymbol{\varepsilon}) \\ -\boldsymbol{\varepsilon}^T \end{bmatrix} \boldsymbol{\omega}_{ob}^b \tag{7.12}$$

We linearize the system around the points $\eta = 1$ and $\boldsymbol{\varepsilon} = 0$ which results in the system :

$$\dot{\mathbf{q}} = \begin{bmatrix} \frac{1}{2}\boldsymbol{\omega}_{ob}^b \\ 0 \end{bmatrix}, \qquad \boldsymbol{\omega}_{ob}^b = 2\dot{\boldsymbol{\varepsilon}} \tag{7.13}$$

7.3.2 Dynamics

7.3.2.1 Rotation matrix

The linearization of the rotation matrix between body and orbit frame around the points $\eta = 1$ and $\varepsilon = 0$ results in the linearized rotation matrix:

$$R_b^o = I_{3x3} + 2S(\boldsymbol{\varepsilon}) \tag{7.14}$$

7.3.2.2 Angular velocity

Now we know that,

$$R_o^b = (R_b^o)^T \tag{7.15}$$

Now putting together the two equations we get the linearized expression for R_o^b becomes

$$R_b^o = I_{3x3} - 2S(\boldsymbol{\varepsilon}) \tag{7.16}$$

which means that,

$$R_o^b = \begin{bmatrix} \frac{1}{2} & \varepsilon_3 & -\varepsilon_2 \\ -\varepsilon_3 & \frac{1}{2} & \varepsilon_1 \\ \varepsilon_2 & -\varepsilon_1 & \frac{1}{2} \end{bmatrix} \tag{7.17}$$

Now from the mathematical modeling of the dynamics, we know that

$$\boldsymbol{\omega}_{ib}^{b} = \boldsymbol{\omega}_{ob}^{b} + R_{o}^{b}\boldsymbol{\omega}_{io}^{b} \tag{7.18}$$

Now substituting the above two equations, into equation, we get,

$$\boldsymbol{\omega}_{ib}^{b} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} 2\varepsilon_1 - 2\omega_o\varepsilon_3 \\ 2\varepsilon_2 - \omega_o \\ 2\varepsilon_3 + 2\omega_o\varepsilon_1 \end{bmatrix} \tag{7.19}$$

the time derivative of $\boldsymbol{\omega}_{ib}^{b}$ is then,

$$\boldsymbol{\omega}_{ib}^{b} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} 2\varepsilon_1 - 2\omega_o\varepsilon_3 \\ 2\varepsilon \\ 2\varepsilon_3 + 2\omega_o\varepsilon_1 \end{bmatrix} \tag{7.20}$$

7.3.3 Gravitational Torque

Now the gravitational torque can be expressed by quaternions in body frame as follows

$$\boldsymbol{\tau}_{grav}^{b} = 3\omega_o^2 \begin{bmatrix} 2(I_z - I_y)(\varepsilon_2\varepsilon_3 + \eta\varepsilon_1)(1 - 2(\varepsilon_1^2 + \varepsilon_2^2)) \\ 2(I_x - I_z)(\varepsilon_1\varepsilon_3 + \eta\varepsilon_2)(1 - 2(\varepsilon_1^2 + \varepsilon_2^2)) \\ 4(I_y - I_x)(\varepsilon_1\varepsilon_2 + \eta\varepsilon_3)(\varepsilon_2\varepsilon_3 + \eta\varepsilon_1) \end{bmatrix} \tag{7.21}$$

Linearized around $\eta = 1$ and $\varepsilon = 0$,

$$\boldsymbol{\tau}^b_{grav} = 3\omega_o^2 \begin{bmatrix} (I_z - I_y)\varepsilon_1 \\ (I_z - I_x)\varepsilon_2 \\ 0 \end{bmatrix} \tag{7.22}$$

7.3.4 Magnetometer Torque

The magnetometer torque is expressed in (7.23) can be represented in orbit frame as follows:

$$\boldsymbol{\tau}^b_m = S(m^b)B^b = S(R^o_b)B^o = S(m^b)[I_{3x3} - 2\eta S(\boldsymbol{\varepsilon}) + 2S^2(\boldsymbol{\varepsilon})]B^o \tag{7.23}$$

linearization around $\eta = 1$ and $\varepsilon = 0$ gives,

$$\boldsymbol{\tau}^b_m = S(m^b)B^b = \begin{bmatrix} B^o_z m_y - B^o_y m_z \\ B^o_x m_z - B^o_z m_x \\ B^o_y m_x - B^o_x m_y \end{bmatrix} \tag{7.24}$$

7.3.5 Linearization of the satellite model

The satellite model is expressed as :

$$I\omega^b_{ib} = -\omega^b_{ib}(I\omega^b_{ib}) + \tau^b_{grav} + S(m^b)B^o \tag{7.25}$$

substituting the above equations, we get,

$$I_x(2\varepsilon_1 - 2\omega_o\varepsilon_3) = (I_y - I_z)(2\omega_o\varepsilon_3 + 8\omega_o^2\varepsilon_1) + (B_z^o m_y - B_y^o m_z) \tag{7.26}$$

$$I_y(2\varepsilon_2) = -6(I_x - I_z)\omega_o^2\varepsilon^2 + (B_x^o m_z - B_z^o m_x) \tag{7.27}$$

$$I_z(2\varepsilon_3 - 2\omega_o\varepsilon_1) = (I_y - I_x)(2\omega_o\varepsilon_3 + 2\omega_o^2\varepsilon_1) - (B_y^o m_x - B_x^o m_y) \tag{7.28}$$

then using that,we get

$$k_x = \frac{I_y - I_z}{I_x} \tag{7.29}$$

$$k_y = \frac{I_x - I_z}{I_y} \tag{7.30}$$

$$k_z = \frac{I_y - I_z}{I_z} \tag{7.31}$$

results in

$$\varepsilon_1 = (1 - k_x)\omega_o\varepsilon_3 - 4k_x\omega_o^2\varepsilon_1 + \frac{1}{2I_x}(B_z^o m_y - B_y^o m_z) \tag{7.32}$$

$$\varepsilon_2 = -3k_x\omega_o^2\varepsilon_2 + \frac{1}{2I_y}(B_x^o m_z - B_z^o m_x) \tag{7.33}$$

$$\varepsilon_3 = -(1-k_z)\omega_o\varepsilon_1 - k_z\omega_o^2\varepsilon_3 + \frac{1}{2I_x}(B_y^o m_x - B_x^o m_y) \tag{7.34}$$

Now the state vector is

$$x = \begin{bmatrix} \varepsilon_1 & \varepsilon_1 & \varepsilon_2 & \varepsilon_2 & \varepsilon_3 & \varepsilon_3 \end{bmatrix} \tag{7.35}$$

$$u = \begin{bmatrix} m_x & m_y & m_z \end{bmatrix}^T \tag{7.36}$$

the model can be written a

$$x(t) = Ax(t) + B(t)u(t) \tag{7.37}$$

where A and B are matrices

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -4k_x\omega_o^2 & 0 & 0 & 0 & 0 & (1-k_x)\omega_o \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -3k_y\omega_o^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & -(1-k_z)\omega_o & 0 & 0 & -k_z\omega_o^2 & 0 \end{bmatrix} \tag{7.38}$$

$$B(t) = \begin{bmatrix} \frac{B_x^o}{I_x} & 0 & 0 \\ 0 & \frac{B_y^o}{I_y} & 0 \\ 0 & 0 & \frac{B_z^o}{I_z} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{7.39}$$

CHAPTER EIGHT: RESULTS

The number of solar arrays depends on the amount of current needed, as well as the size of the satellite, and also very important, the time for the mission undertaken. Now, different scenarios were simulated, and using the controls, different currents were produced and the solar arrays needed were calculated.

Table 1: Overview of Power Generated

Current	Force	No of arrays		Power Generated					
(mA)	(mN)	Cell	Interface	Sustained (mW)			Maximum (mW)		
-	-	-	-	@90	@45	Max	@90	@45	Max
140	0.4	2	2	624.4	401.8	2052.4	635.6	409.0	2089.2
250	1	2	2	1115.0	717.5	3664.9	1135.0	730.3	3730.7
1400	3.5	4	4	6244.0	4018.0	20524.0	6356.0	4090.0	20892.0
6000	15	15	17	26760	17219	87958	27240	17528	89536
12500	45	30	30	55750	35870	183250	56750	36520	186530

Table 1 gives us a fair idea as to how much solar arrays are needed for obtaining the desired current and power for our mission. Now from the table, we take the the thrust of 15 mN, and then see the the orbit maneuvers and the time taken for the satellite to reach the desired trajectory.

Table 2: Current per Standard Size Cell

Area per cell	6.88cm x 3.81cm-0.33cm^2 = 24.55cm^2
Current Density per cell	16.0 mA/cm^2
Current per cell	24.55cm^2 x 16.0 mA/cm^2 = 392.8 mA

Table 3: Current per Interface Size Cell

Area per cell	6.88cm x 3.3cm-0.33cm^2 = 22.61cm^2
Current Density per cell	16.0 mA/cm^2
Current per cell	22.61cm^2 x 16.0 mA/cm^2 = 361.76 mA

Table 4: Total Current from Arrays

Current of 2 panels (with sun incident angle @ 90 deg)	392.8 mA
Current of 2 panels (with sun incident angle @ 45 deg)	361.76 mA
Total Current	754.56 mA

Table 5: Power per panel (with sun incident angle @ 90 deg)

Sustained Power	4.46 V x 392.8 mA=1751.9 mW
Maximum Power	4.54 V x 392.8 mA=1783.3 mW

Table 6: Power per panel (with sun incident angle @ 45 deg)

Sustained Power	1751.9 mW x cos(45) = 1138 mW
Maximum Power	1783.3 mW x cos(45) = 1158.91 mW

Table 7: Total Power from the Array

Sustained Power	2*1751.9 mW + 2*1138 mW=5779.8 mW
Maximum Power	2*1783.3 mW + 2*1158.91 mW = 5884.42 mW

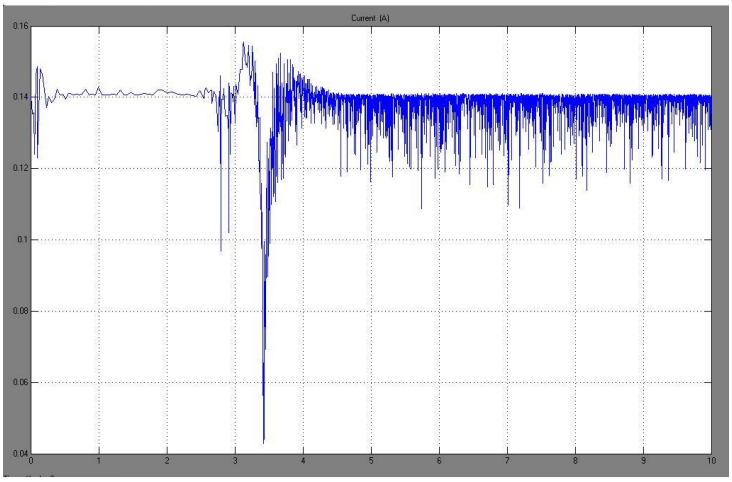

Figure 14: Current : 0.140 A

A series of tables(Table 2 to Table 7) were created to find out, how much current and power is being produced by each individual cell, and that helps us in calculating the number of cells required to produce the current in the different scenarios as given in Table 1.

The Figures from 14 to 19 gives us the amount of Current and Force produced for different scenarios.

The thrust profile shown in the Figures 20 to 23 above are the different components of thrust in each direction. These thrust profiles are put in the controls algorithm for orbit maneuvers, and the results of those maneuvers are shown in the Figures below.

From the Figure 25, we can see that it takes about 6-7 orbits to reach the desired trajectory, as far as change in radius is concerned.

Figure 26 shows the 3-D image of the planar maneuvering.

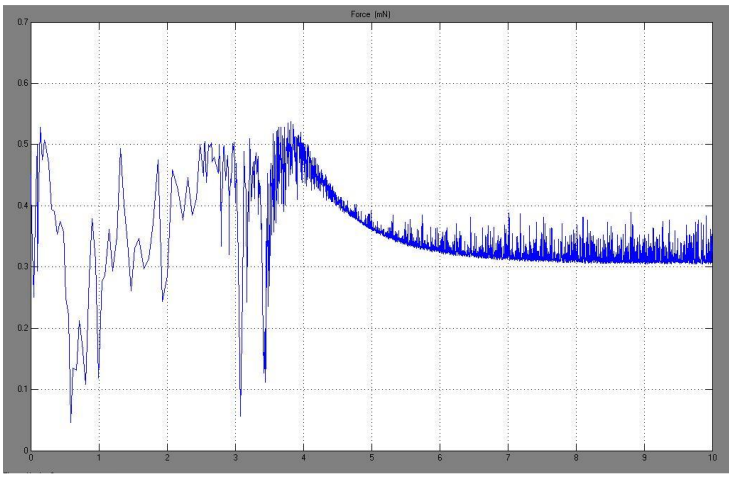

Figure 15: Force produced for current of 0.140 A

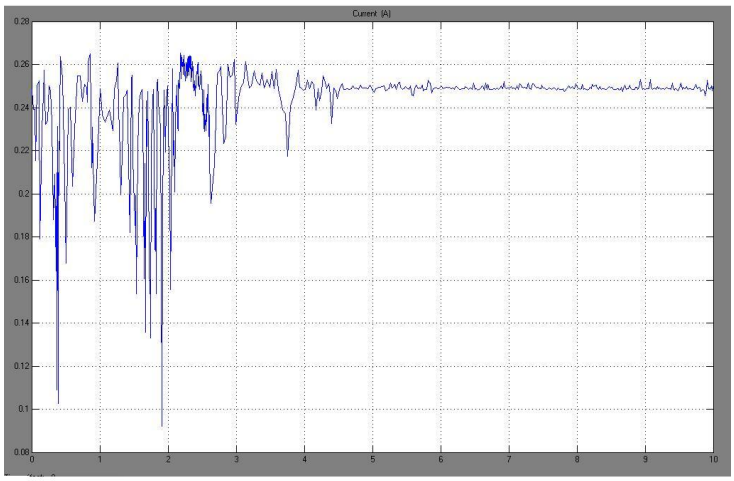

Figure 16: Current Produced : 0.250 A

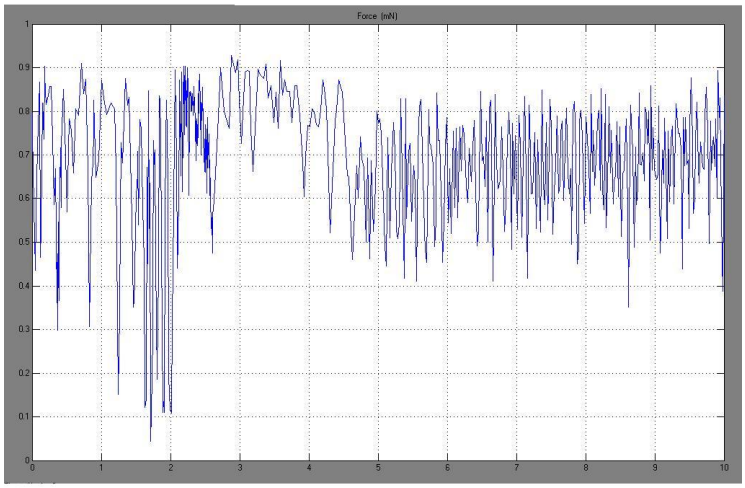

Figure 17: Force produced for current of 0.250 A

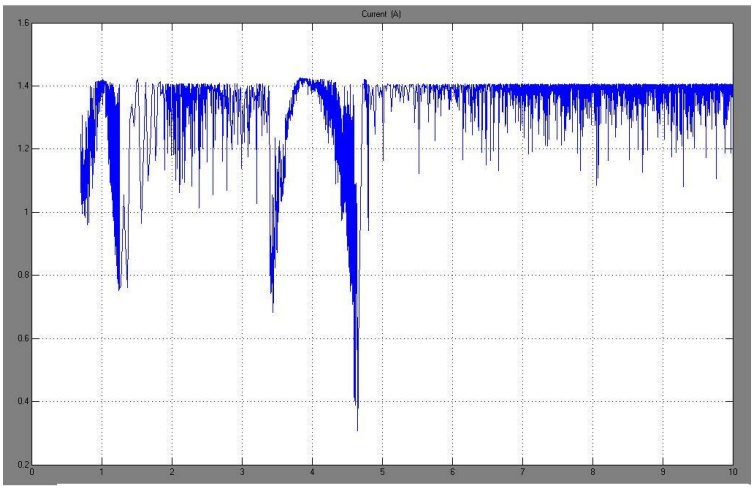

Figure 18: Current Produced : 1.4 A

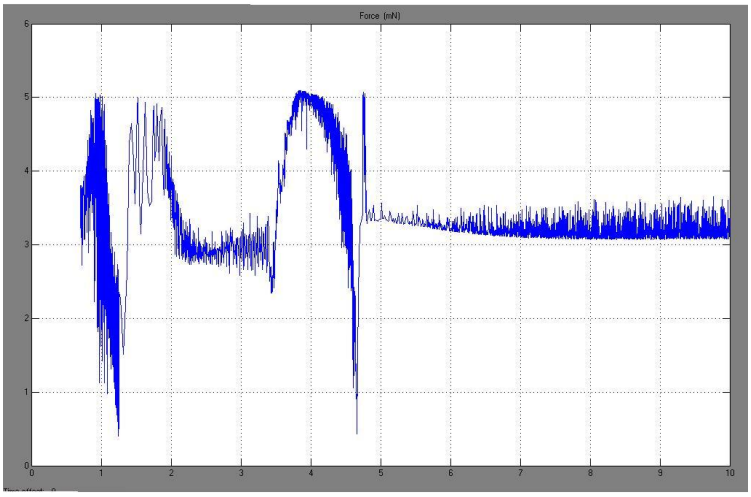

Figure 19: Force produced for current of 1.40 A

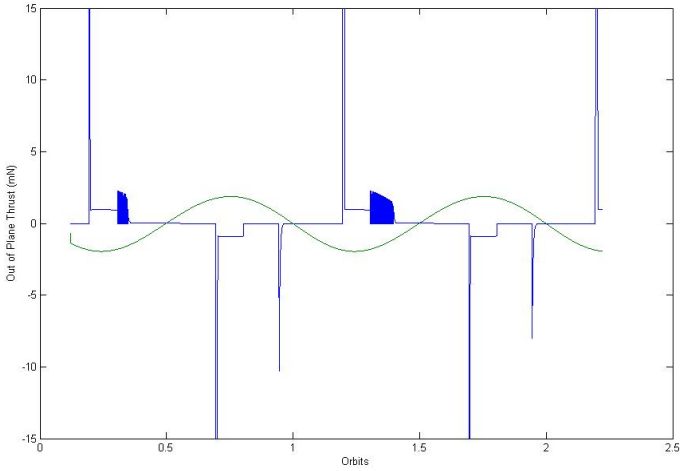

Figure 20: Out of Plane Thrust(mN)

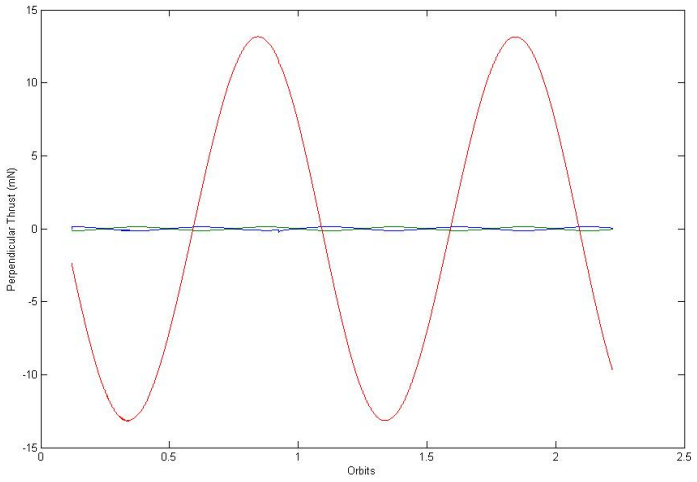

Figure 21: Perpendicular Thrust(mN)

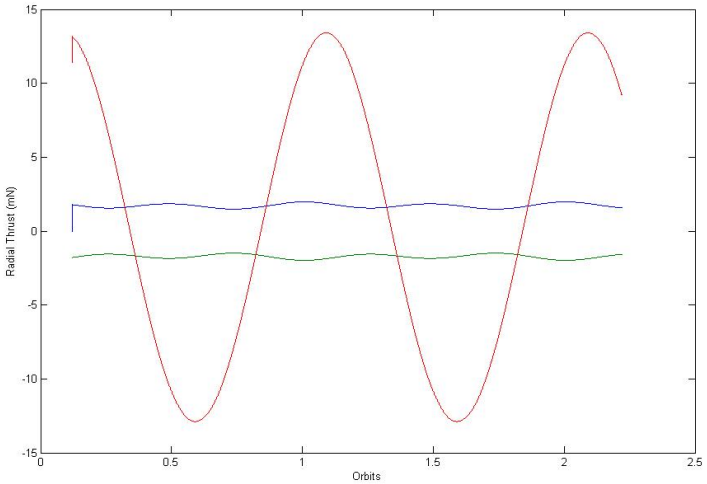

Figure 22: Radial Thrust(mN)

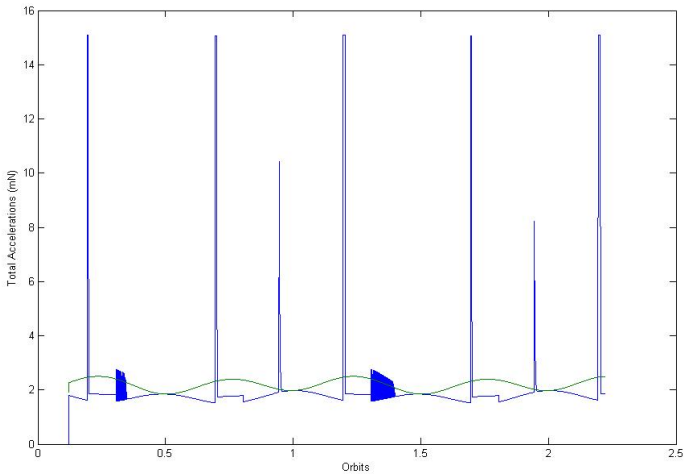

Figure 23: Total Thrust(mN)

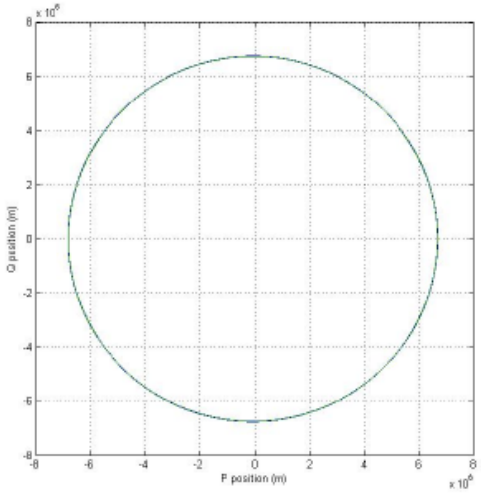

Figure 24: Planar Maneuver

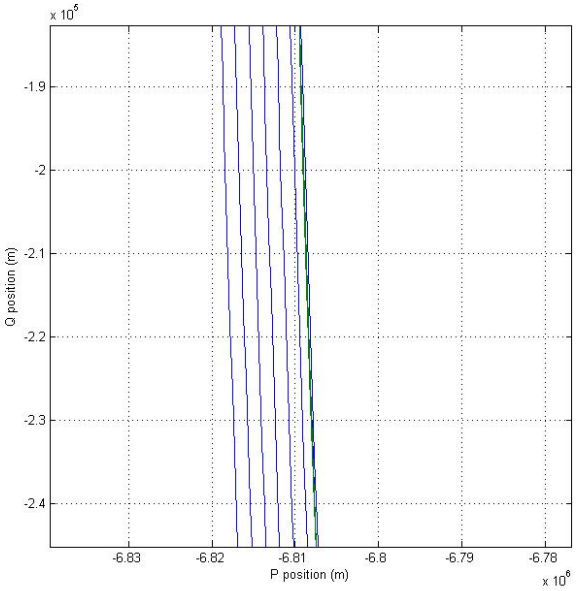

Figure 25: Planar Maneuver - Zoomed In

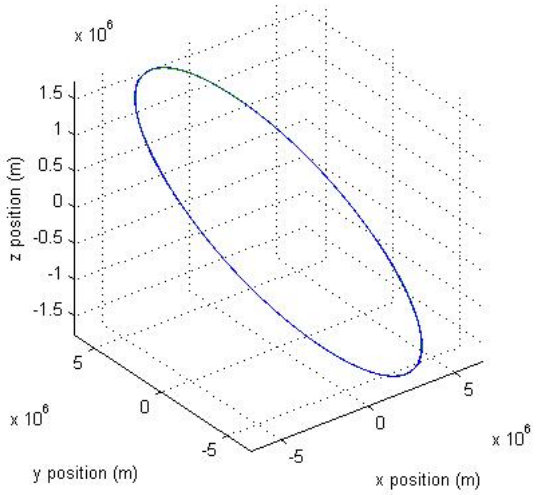

Figure 26: 3-D Image - Planar Maneuver

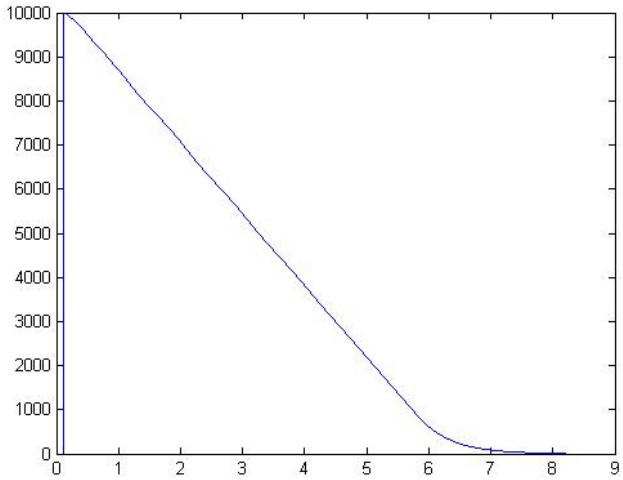

Figure 27: Distance(m) v/s Orbits

Now from the graph in Figure 27, its very clear that the initial conditions of the radius is 10000 m and, it taken about 7 orbits to reach the desired radius, using the Thrust.

The graph in Figure 28, is basically when the inclination and the ascending node are different, i.e the satellite has to perform a non-planar maneuver. But still, the controls algorithm are designed in such a way that, not only does it take care of the planar, but also non-planar maneuvers in the given time and using the same energy.

Once the desired trajectory is achieved, then maintaining a minimum distance from the chosen satellite takes about only 1-2 orbits, as shown in the Figure 29. Apart from that, the ADCS takes care of the satellite to remain stable and always face at the sun.The Graph in Figure 30 gives us an idea of how the system remains stable using the Torque Coils in all the three axis.

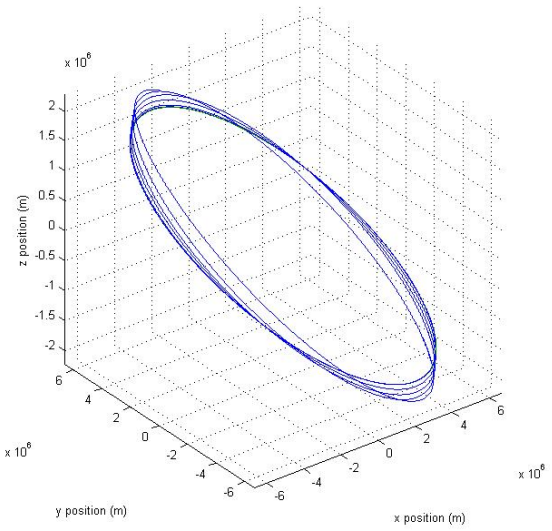

Figure 28: Change in Ascending Node and Inclination

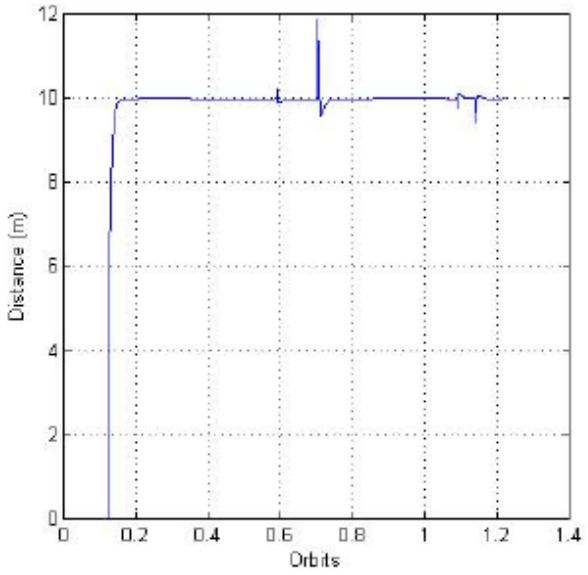

Figure 29: Distance from chosen Satellite

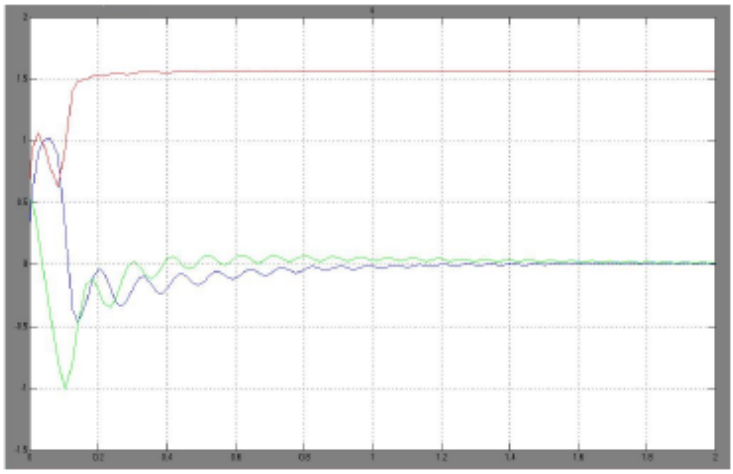

Figure 30: Applied Torque on Coils

CHAPTER NINE: CONCLUSION

Controls Algorithm for satellite to maneuver in different orbits was developed. The Flexibility in the controls algorithm is such that, any kind of maneuver can be done, for the satellite to achieve a desired trajectory. Its just time-dependent, if the maneuver is little more involved. Basically this helps in reaching other satellites, look and them, take pictures, and also look out for those "Killer Satellites", and send all these information back to the Base. These kind of algorithm has lots of other applications, other than the one mentioned above. The algorithms are a good way to see solutions while visualizing dynamical effects. The only problem one might face is that because its having low thrust capabilities, it might take longer than other thrusters. But in spite of these minor disadvantages, overall its a win-win situation, as the cost is less and there can be multiple number of such satellites deployed at the same time and still do a work of a big satellite at a much lower cost. It can implemented on picosatellites like the FUNSAT; in this concept, propulsion is virtually free!

APPENDIX A
DEFINITIONS AND NOTATIONS

A.1 Classical Orbital Dynamics

Kepler's first law of planetary motion state that a satellite's orbit is an ellipse and the body it orbits is at one focus. Any Keplerian orbit can be completely described by six orbital elements, two to describe the size and shape, three to describe the orientation, and one to describe the satellite location. In this report, the classical orbital elements will be used. These elements are: semi-major axis, eccentricity, inclination, right ascension of the ascending node, argument of periapsis and true anomaly at epoch. A variety of orbital elements can be used to describe orbits where the classical system is just one of the more common. The orbital elements are shown in Figure 31 and described below.

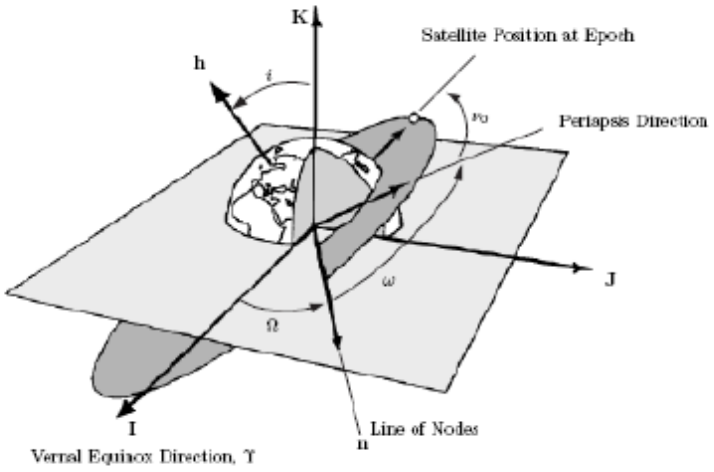

Figure 31: Orbital Elements

Semi-major Axis (a): The semi-major axis describes the size of the orbit. For a circular

orbit, the semi-major axis is the radius of the earth plus the altitude (h) of the satellite. For elliptical orbits, the semi-major axis is half of the major axis diameter.

Eccentricity (e): The eccentricity describes the shape of the orbit ellipse. It is a measure of how circular an orbit is. It can be calculated as the distance from the center of an ellipse to the focus divided by the semi-major axis. For a circular orbit, the eccentricity is 0.

Inclination (i): The inclination describes the orientation of the orbit. It is defined as the angle between the orbit plane and the equatorial plane.

Right Ascension of the Ascending Node (Ω): The Right Ascension of the Ascending Node (RAAN) describes the orientation of the satellite. It is the angle from the vernal equinox to the ascending node. The ascending node is the location where the satellite crosses the equatorial plane traveling from south to north.

Argument of Perigee (ω): The argument of perigee describes the orientation of the orbit. It is the angle between the ascending node and the eccentricity vector. The eccentricity vector points from the Earth's center to the perigee of the ellipse with a magnitude equal to the eccentricity. It is undefined for circular orbits.

True Anomaly (v_o): The true anomaly describes the location of the satellite within the orbit. It is the angle between the eccentricity vector and the satellite measured in the direction of the satellite's motion.

While the above six orbital elements are classically used to describe the position of a satellite, they are not exclusive. Instead of the argument of periapsis, the longitude of periapsis, π, is occasionally used. This is the angle from the I direction to periapsis measured

eastward. Additionally, the argument of latitude at epoch, uo can be used. This is the angle in the plane of the orbit between the ascending node and the radius vector of the satellite at epoch. These values are related by

$$u_o = \omega + v_o \tag{A.1}$$

The true longitude at epoch, lo, may also be used to describe the position of the satellite. This angle is measured eastward from the I axis to the ascending node, and then in the orbital plane to the radius direction at epoch. This relation is defined as

$$l_o = \Omega + \omega + v_o = \pi + v_o = \Omega + u_o \tag{A.2}$$

Certain orbits cause some of the orbital elements to be undefined. When the orbit is circular, there is no periapsis, and ω, π, and v_o are undefined. When the orbit is equatorial, there is no ascending node and therefore ω and u_o are undefined. In these cases, using l_o instead of ω is useful.

A.2 Coordinate Systems

When describing orbital motion, it is necessary to reveal which coordinate system is used.The coordinate systems must be inertial such that the frame is fixed to an outside observer. A fixed coordinate system may also be used to describe satellite motion, but a

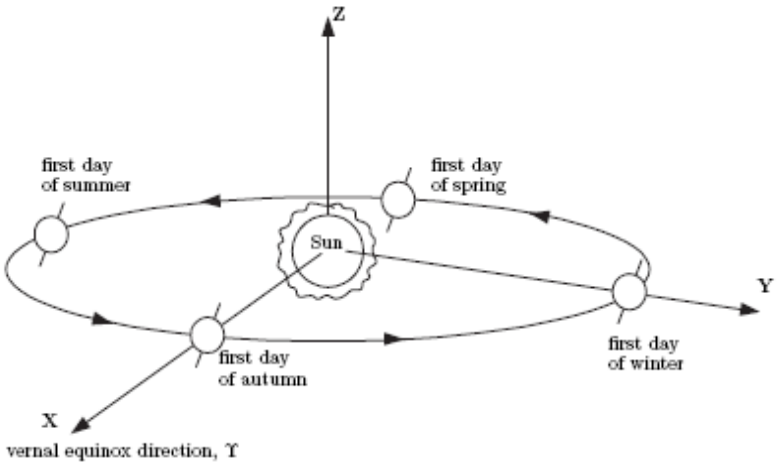

Figure 32: Heliocentric-Ecliptic Coordinate System

rotation between the inertial and Earth-fixed coordinate systems must be incorporated into the definition.

Heliocentric-Ecliptic : Bodies that orbit around the Sun, such as the Earth and other planets as well as interplanetary space vehicles, are typically described in the Heliocentric-ecliptic frame of reference shown in Figure 32. This reference frame is inertial, with the Z direction perpendicular to the plane of the ecliptic, which is the plane of the Earth's revolution around the Sun. The direction of the X axis is in the vernal equinox direction, and the Y direction is orthogonal.

Earth-Centered Inertial : The Earth-Centered Inertial (ECI) coordinate system is centered in the middle of the Earth, as shown in Figure 33. The Z axis points through the geographic North Pole, or the axis of rotation. The X axis is in the direction of the vernal

equinox, and the Y direction is orthogonal. The Earth rotates with respect to the ECI coordinate frame.

Earth-Centered Earth-Fixed : The Earth-centered Earth-fixed (ECEF) reference frame also has its origin at the center of the Earth, but it rotates relative to inertial space, shown in Figure The K axis is through the North Pole, and the I axis points to the Greenwich Meridian. The angle between the vernal equinox direction and the Greenwich Meridian must be defined. This is known as the Greenwich sidereal time, θ_g. Greenwich sidereal time is documented at various epochs and can be extracted from data tables as θ_{g0}. At any time after epoch, θ_g can be determined from θ_{g0} by

$$\theta_g = \theta_{g0} + \omega_{\oplus}(t - t_0) \tag{A.3}$$

where $\omega_{\oplus}$ is the angular velocity of the Earth. On January 1, 2000 at midnight, the value of θ_{g0} was equal to $6^h39^m52.2707^s$, or 99.96779° according to the Multi year Interactive Computer Almanac from U.S. Naval Observatory. This value changes slightly from year to year, and on January 1, 2001 was equal to $6^h42^m51.5354^s$, or 100.71473° .

A.3 Reference Frames

Three main reference frames are used to describe the orientation, or attitude, of a spacecraft in orbit. These are the inertial, orbital, and body frames.

Inertial Frame : An inertial frame is used for attitude applications. The X direction

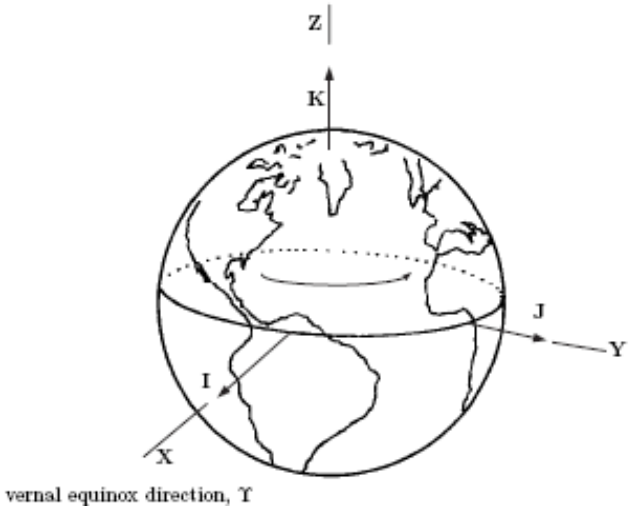

Figure 33: Earth-Centered Inertial Reference Frame

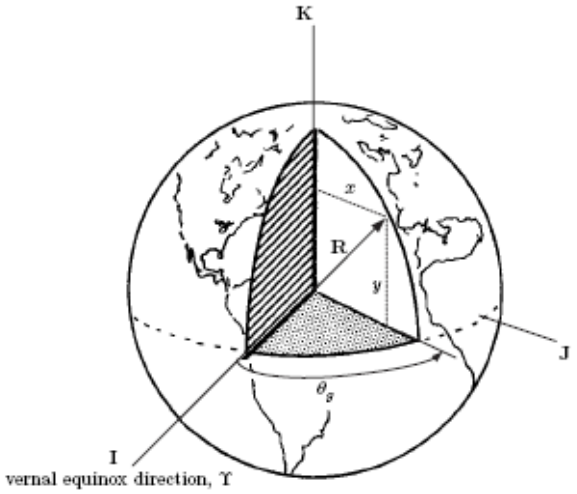

Figure 34: Earth-Centered Earth Fixed Reference Frame

points from the focus of the orbit to the vernal equinox, the Z direction is in the orbital angular velocity direction, and Y is perpendicular to X and Z.

Orbital Frame : The orbital frame is located at the mass center of the spacecraft, and the motion of the frame depends on the orbit. This frame is non inertial because of orbital acceleration and the rotation of the frame. The o3 axis is in the direction from the spacecraft to the Earth o2 is the direction opposite to the orbit normal, and o1 is perpendicular to o2 and o3. In circular orbits, o1 is the direction of the spacecraft velocity. The three directions o1, o2, and o3 are also known as the roll, pitch, and yaw axes, respectively. Figure 35 shows a comparison of the inertial and orbital frames in an equatorial orbit.

Body Frame : Like the orbital frame, the body frame has its origin at the spacecraft's mass center. This frame is fixed in the body, and therefore is non-inertial. The relative orientation between the orbital and body frames is the basis of attitude dynamics and control.

Principal Axis : Principal axes are a specific body-fixed reference frame. This axis system has its origin at the mass center, and is oriented such that the moment of inertia matrix is diagonal and known as the principal moments of inertia.

A.4 Rotation Matrix

The rotation matrix can be interpreted in three different ways; as a transformation of a vector represented in one coordinate frame to another frame, as a rotation of a vector within the same frame and finally as a description of mutual orientation between two frames. The

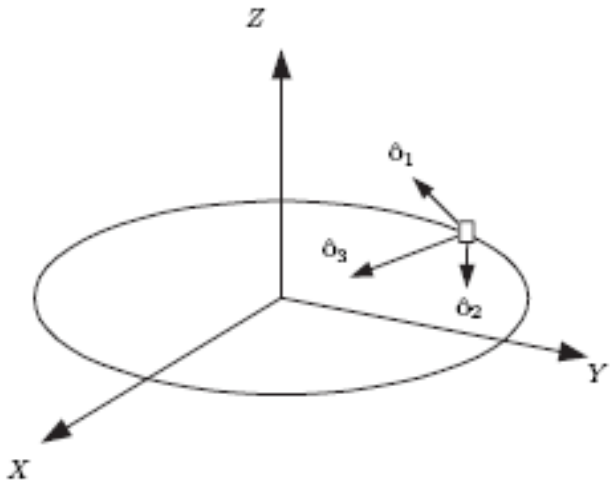

Figure 35: Earth-Centered Inertial and Orbital Reference Frames

rotation matrix R from frame a to b is denoted R_a^b.

It is defined by

$$SO(3) = R|R\epsilon R^{3x3}, R^T R = I \text{ and } detR = 1 \tag{A.4}$$

where R^{3x3}

is the set of all 3X3 matrices with real elements and I is the 3X3 identity matrix. In general, the rotation of a vector from one frame to another, can be written with the following notation:

$$v^{to} = R_{from}^{to} v^{from} \tag{A.5}$$

A useful parameterization of the rotation matrix is the angle-axis parameterization, $R_{\alpha,\beta}$

corresponding to a rotation β about the λ axis:

$$R_{\lambda,\beta} = I + S(\lambda)sin\beta + (1 - cos\beta)S^2(\lambda) \tag{A.6}$$

where S is the skew-symmetric operator. The rotation matrix also satisfies

$$R_a^b = (R_a^b)^{-1} = (R_a^b)^T \tag{A.7}$$

Simple rotations using Euler angles as parameters, are defined as

$$R_x(\psi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & \sin\psi \\ 0 & -\sin\psi & \cos\psi \end{bmatrix} \tag{A.8}$$

$$R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \tag{A.9}$$

$$R_z(\phi) = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A.10}$$

where the subscriptions x,y and z denotes the axis the angles ψ, θ and ϕ revolves about, respectively.

A.5 Transformation between different frames

The different rotations between frames used in this report are described here.

A.5.1 Transformation from Earth-Centered Orbit to ECI & ECEF frames

The rotation between these frames can only be done using the orbit estimator, and is necessary to be able to compare measurements with their respective reference model.

$$R_{OC}^{I} = R_z(-\Omega)R_x(-i)R_z(-\omega) \tag{A.11}$$

$$R_{OC}^{E} = R_z(-\Omega + \theta)R_x(-i)R_z(-\omega) \tag{A.12}$$

where Ω is the Right Ascension of the Ascending Node, i is the inclination of the satellite, ω is the Argument of Perigee and θ is the ascension of the zero meridian.The R_y and R_z are the different rotations defined by equations A.8 and A.8 respectively.

A.5.2 Transformation from ECEF and ECI frame

The rotation of the ECEF frame relative to the ECI frame is a rotation about the coincident z_I and z_E-axes, equal to an angle $\alpha = \omega_e t$, where ω_e is the Earth rotation rate, and t is the time passed since the ECEF and ECI frame were aligned. This rotation can be expressed as

$$R_E^I = R_{z_I,\alpha} = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A.13}$$

A.5.3 Transformation from ECI to Orbit frame

This rotation is dependent on the satellite rotation velocity ω_o. The Orbit frame is rotated an angle β about the y_I-axis, and is expressed as $\beta = \beta_0 + \omega_0 t$, where β_0 is the drop angle, or latitude position, of the satellite and t is the time since last passing of 0^0 latitude. This can be expressed as

$$R_{y_I,\beta} = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \tag{A.14}$$

In addition, the Orbit frame is upside-down relative to the ECI frame. This motivates

the following rotation about the x_I-axis:

$$R_{z_I,\pi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\pi & -\sin\pi \\ 0 & \sin\pi & \cos\pi \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{A.15}$$

which, combined with Equation A.14, gives the total rotation necessary to transform a vector given in ECI frame to an Orbit frame representation:

$$R_I^O = R_{x_I\pi} R_{y_I\beta} = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & -1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} = \begin{bmatrix} \cos\mu & 0 & \sin\mu \\ 0 & 1 & 0 \\ -\sin\mu & 0 & \cos\mu \end{bmatrix} \tag{A.16}$$

since both μ and β represent the latitude position of the satellite

A.5.4 Transformation from Orbit to Body frame

The rotation matrix used extensively in this report, is the transformation between Orbit and Body frame, represented by R_O^B . This rotation is dependent on the attitude of the satellite, and by estimating this rotation matrix, the attitude can be determined. Using Equation A.6 with $\lambda = \varepsilon$ and $\beta = \varepsilon$ we get the rotation

$$R_{\varepsilon,\eta} = 1 + 2\eta S(\varepsilon) + 2S^2(\varepsilon) \tag{A.17}$$

The rotation matrix from Body to Orbit frame is thus

$$R_B^O = 1 + 2\eta S(\varepsilon) + 2S^2(\varepsilon) \tag{A.18}$$

and by using the definition Equation A.7 on Equation A.18, a representation of the rotation from Orbit to Body Frame can be calculated:

$$R_O^B = (R_B^O)^T = \begin{bmatrix} \varepsilon_1^2 - \varepsilon_2^2 + \varepsilon_3^2 + \eta^2 & 2(\varepsilon_1\varepsilon_2 + \eta\varepsilon_3) & 2(\varepsilon_1\varepsilon_3 - \eta\varepsilon_2) \\ 2(\varepsilon_1\varepsilon_2 - \eta\varepsilon_3) & -\varepsilon_1^2 + \varepsilon_2^2 - \varepsilon_3^2 + \eta^2 & 2(\varepsilon_2\varepsilon_3 + \eta\varepsilon_1) \\ 2(\varepsilon_1\varepsilon_3 + \eta\varepsilon_2) & 2(\varepsilon_1\varepsilon_3 - \eta\varepsilon_1) & -\varepsilon_1^2 - \varepsilon_2^2 + \varepsilon_3^2 + \eta^2 \end{bmatrix} \tag{A.19}$$

The rotation matrix can also be written as

$$R_O^B = \begin{bmatrix} c_1^B & c_2^B & c_3^B \end{bmatrix} \tag{A.20}$$

where $c_i^B = \begin{bmatrix} c_{ix}^B & c_{iy}^B & c_{iz}^B \end{bmatrix}^T$, are column vectors. When $c_3^B = \begin{bmatrix} 0 & 0 & \pm 1 \end{bmatrix}^T$, the z_0_axis and $z_B - axis$ are aligned. This vector can thus be seen as a description of the deviation between z_0_axis and $z_B - axis$, and is an indication of the performance of attitude control system.

A.6 Time

Time is an important parameter when calculating the position of an object in orbit. The standard SI unit(International System of Units) for time is a second(sec). The second is then used to determine minutes[min], hours[h], days, months and years. There are a fixed ratio between seconds, minutes, hour, day and week, but no fixed ratio between days, months and years. This is due to the use of leap-days, it is therefore inconvenient to use for computer computations. A solution to this is to use the Julian Day.

A.6.1 International atomic time(TAI)

International atomic time, is a very accurate and stable reference time. It uses the radiation period of a cesium nuclide, 133 Ce, where 9,192,631,770 periods constitute 1 second in the SI system.

A.6.2 Universal Time(UT)

Universal Time, also called Greenwich mean time(GMT), is the mean solar time at the Royal Greenwich Observatory near London in England, which coincide with the 0 longitude.

A.6.3 Coordinated Universal Time(UTC)

Coordinated Universal Time, also called Zulu time (Z), was introduced and broadcasted in January 1972 (Kristiansen 2000). A second in UTC is equal to a second in TAI, but it is kept within 0.90 seconds of the actual rotation of the Earth, by correcting it with 1 second steps, usually at the end of June an December (This gives rise to the leap year).

A.6.4 Civil Time

Different time zones, give rise to the Civil time (T_{civil}), which is the time observed by people on their clocks. Civil time differs from the UT, by an integer number of hours, and can be roughly calculated by :

$$T_{civil} \approx UT \pm (L + 7.5)/15 \tag{A.21}$$

where L is the longitude, given in degrees (with positive sign in eastern direction), and UT and T_{civil} given in integer numbers, and T_{civil} is irrespective of daylight-saving time.

A.6.5 Julian day

The Julian calender is a continuous time count from the number of days since Greenwich noon on January 1, 4713BC. This is the solution adopted for astronomical use, and was proposed by the Italian scholar Joshep Scaliger in 1582 AD. The Julian Day can be calculated

with:

$$JD = day + \frac{153 - m}{5} + 365y + y/4 - 32083 \tag{A.22}$$

where:

$$month = month + 12a - 3 \tag{A.23}$$

$$y = year + 4800 - a \tag{A.24}$$

and

$$a = \frac{14 - month}{12} \tag{A.25}$$

A.6.6 Modified Julian day(MJD)

The Julian Date is, as mentioned above, adopted for astronomical use, but for space application it present a minor problem, because it starts at 12:00 UT,instead of 00:00 UT, as the civil calender does. To remedy this the Modified Julian Date is used. The

MJD is given by:

$$MJD = JD - 2400000.5 \tag{A.26}$$

and starts at midnight.

A.7 Quaternions-Kinematic Equation

The quaternion $\mathbf{q}$, is a 4 x 1 matrix consisting of a vector portion, $\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 & \varepsilon_2 & \varepsilon_3 \end{bmatrix}^T$ and a scalar portion η,and it is represented as follow,

$$\mathbf{q} = \begin{bmatrix} \boldsymbol{\varepsilon} \\ \eta \end{bmatrix} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \eta \end{bmatrix} \tag{A.27}$$

Quaternions have advantages and disadvantages over rotation matrix notation. The singularities that exist when certain Euler angles are small are eliminated with the use of quaternions. However, the physical meaning of quaternions is obscure and not as intuitive as rotation angles.

The quaternion is in the form,

$$\mathbf{q} = \begin{bmatrix} \mathbf{a}\sin(\frac{\phi}{2}) \\ \cos(\frac{\phi}{2}) \end{bmatrix} \tag{A.28}$$

where a is the euler axis vector, and ϕ is the angle of rotation.

At time $t + \Delta t$, the quaternion is equal to

$$\mathbf{q}(t+\Delta t) = (\cos\frac{\Delta\phi}{2} + \sin\frac{\Delta\phi}{2}\begin{bmatrix} 0 & a_3 & -a_2 & a_1 \\ -a_3 & 0 & a_1 & a_2 \\ a_2 & -a_1 & 0 & a_3 \\ -a_1 & -a_2 & -a_3 & 0 \end{bmatrix})\mathbf{q}(t) \tag{A.29}$$

Since Δt is very small, and $\Delta\phi = \omega\Delta t$, where ω is the magnitude of the instantaneous angular velocity of the body, the following small angle assumptions are used:

$$\cos\frac{\Delta\phi}{2} = 1, \;\; \sin\frac{\Delta\phi}{2} = \frac{1}{2}\omega\Delta t \tag{A.30}$$

This leads to

$$\mathbf{q}(t+\Delta t) = [1 + \frac{1}{2}S(\boldsymbol{\omega})\Delta t]\mathbf{q}(t) \tag{A.31}$$

This equation represents the kinematic equation of motion of the spacecraft.

where $S(\omega)$ is the skew symmetric matrix.

$$S(\boldsymbol{\omega}) = \begin{bmatrix} 0 & \omega_3 & -\omega_2 & \omega_1 \\ -\omega_3 & 0 & \omega_1 & \omega_2 \\ \omega_2 & -\omega_1 & 0 & \omega_3 \\ -\omega_1 & -\omega_2 & -\omega_3 & 0 \end{bmatrix} \tag{A.32}$$

The derivative of the quaternion is

$$\dot{\mathbf{q}} = \frac{1}{2}\begin{bmatrix} \eta & -\varepsilon_3 & \varepsilon_2 \\ \varepsilon_3 & \eta & -\varepsilon_1 \\ -\varepsilon_2 & \varepsilon_1 & \eta \\ -\varepsilon_1 & -\varepsilon_2 & -\varepsilon_3 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \dot{\mathbf{q}} = \frac{1}{2}\begin{bmatrix} \eta I_{3x3} + S(\boldsymbol{\varepsilon}) \\ -\boldsymbol{\varepsilon}^T \end{bmatrix} \boldsymbol{\omega}_{ob}^{b} \tag{A.33}$$

where $S(\boldsymbol{\varepsilon})$ is a skew symmetric matrix

$$S(\boldsymbol{\varepsilon}) = \begin{bmatrix} 0 & -\varepsilon_3 & \varepsilon_2 \\ \varepsilon_3 & 0 & -\varepsilon_1 \\ -\varepsilon_2 & \varepsilon_1 & 0 \end{bmatrix} \tag{A.34}$$

APPENDIX B
PLANAR ORBITAL STATE EQUATIONS

The first state equation is the radial rate.

$$\frac{dr}{dt} = V_R \tag{B.1}$$

The time rate of change of the radial and perpendicular velocities are obtained from the gravitational motion equations. It is known that the position vector in the RSW frame is

$\mathbf{r} = r \cdot \mathbf{e}_R$

Taking the time derivative results in the velocity vector as follows

$$\frac{d\mathbf{r}}{dt} = r\frac{d\mathbf{e}_R}{dt} + \frac{dr}{dt} \cdot \mathbf{e}_R \tag{B.2}$$

$$\frac{d\mathbf{e}_R}{dt} = \frac{d\nu}{dt} \cdot \mathbf{e}_S \tag{B.3}$$

$$\frac{d\mathbf{e}_S}{dt} = -\frac{d\nu}{dt} \cdot \mathbf{e}_R \tag{B.4}$$

$$\frac{d\mathbf{r}}{dt} = \frac{dr}{dt} \cdot \mathbf{e}_R + r\frac{d\nu}{dt} \cdot \mathbf{e}_S = V_R \cdot \mathbf{e}_R + V_S \cdot \mathbf{e}_S \tag{B.5}$$

The radial rate and true anomaly are obtained from the velocity vector

$$\frac{dr}{dt} = V_R \tag{B.6}$$

$$V_S = r\frac{d\nu}{dt} \tag{B.7}$$

Taking the time derivative of the velocity gives the acceleration vector

$$\frac{d^2\mathbf{r}}{dt^2} = V_R \cdot \frac{dv}{dt} \cdot \mathbf{e}_S + \frac{dV_R}{dt} \cdot \mathbf{e}_R - V_S \cdot \frac{dv}{dt} \cdot \mathbf{e}_R + \frac{dV_S}{dt} \cdot \mathbf{e}_S \tag{B.8}$$

$$\frac{d^2\mathbf{r}}{dt^2} = \frac{V_R V_S}{r} \cdot \mathbf{e}_S + \frac{dV_R}{dt} \cdot \mathbf{e}_R - \frac{V_S^2}{r} \cdot \mathbf{e}_R + \frac{dV_S}{dt} \cdot \mathbf{e}_S \tag{B.9}$$

$$\frac{d^2\mathbf{r}}{dt^2} = (\frac{dV_R}{dt} - \frac{V_S^2}{r}) \cdot \mathbf{e}_R + (\frac{dV_S}{dt} + \frac{V_R V_S}{r}) \cdot \mathbf{e}_S \tag{B.10}$$

Inserting it into the gravity potential equation above results in

$$\begin{aligned}(\frac{dV_R}{dt} - \frac{V_S^2}{r} + \frac{\mu}{r}) \cdot \mathbf{e}_R + (\frac{dV_S}{dt} + \frac{V_R V_S}{r}) \cdot \mathbf{e}_S + (\frac{dV_W}{dt}) \cdot \mathbf{e}_W \\ = (P_R + U_R) \cdot \mathbf{e}_R + (P_S + U_S) \cdot \mathbf{e}_S + (P_W + U_W) \cdot \mathbf{e}_W\end{aligned} \tag{B.11}$$

Solving for the time rate changes and turning the vector equation into matrix form gives

$$\begin{bmatrix} \frac{dV_R}{dt} \\ \frac{dV_S}{dt} \\ \frac{dV_W}{dt} \end{bmatrix} = \begin{bmatrix} \frac{V_S^2}{r} - \frac{\mu}{r} + P_R + U_R \\ -\frac{V_R V_S}{r} + P_S + U_S \\ P_W + U_W \end{bmatrix} \tag{B.12}$$

The state equation for the true anamoly is taken from knowing the perpendicular velocity is the angular rate times the radial distance.

$$V_S = r\frac{d\nu}{dt} \tag{B.13}$$

$$\frac{d\nu}{dt} = \frac{V_S}{r} \tag{B.14}$$

Hence the equations affecting the attributes of the orbit are

$$\frac{dr}{dt} = V_R \tag{B.15}$$

$$\frac{dV_R}{dt} = \frac{V_s^2}{r} - \frac{\mu}{r^2} + P_R + U_R \tag{B.16}$$

$$\frac{dV_S}{dt} = -\frac{V_R \cdot V_S}{r} + P_S + U_S \tag{B.17}$$

$$\frac{dv}{dt} = \frac{V_S}{r} \tag{B.18}$$

APPENDIX C
ANGULAR ORBITAL STATE EQUATIONS

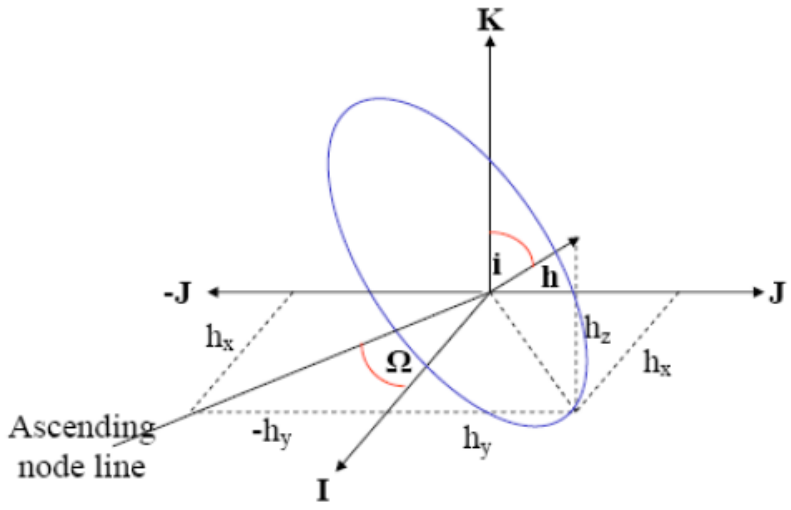

Figure 36: Angular Momentum Vector

The angular momentum vector can be related to the ascending node and inclination angle by geometry

Geometrically, the inclination and ascending node are given by

$$\cos(i) = \frac{h_z}{h} \tag{C.1}$$

$$tan(\Omega) = -\frac{h_y}{h_x} \tag{C.2}$$

To find their rates of change take their time derivatives

$$\sin(i) \cdot \frac{di}{dt} = \frac{h.\frac{dh_z}{dt} - h_z.\frac{dh}{dt}}{h^2} \tag{C.3}$$

$$\frac{di}{dt} = \frac{h.\frac{dh_z}{dt} - h_z.\frac{dh}{dt}}{\sin(i) \cdot h^2} \tag{C.4}$$

$$\frac{1}{cos^2(\Omega)} \cdot \frac{d\Omega}{dt} = \frac{h_y.\frac{dh_x}{dt} - h_x.\frac{dh_y}{dt}}{h_x^2} \tag{C.5}$$

$$\frac{d\Omega}{dt} = cos^2(\Omega) \cdot \frac{h_y.\frac{dh_x}{dt} - h_x.\frac{dh_y}{dt}}{h_x^2} \tag{C.6}$$

Since

$$cos^2(\Omega) = \frac{h_x^2}{h_x^2 + h_y^2} \tag{C.7}$$

The ascending node time rate of change can be written as

$$\frac{d\Omega}{dt} = \frac{h_y.\frac{dh_x}{dt} - h_x.\frac{dh_y}{dt}}{h_x^2 + h_y^2} \tag{C.8}$$

From inspection, it can be concluded that geometrically the angular momentum vector are,

$$h_x = h \cdot \sin(i) \cdot \sin(\Omega) \tag{C.9}$$

$$h_y = -h \cdot \sin(i) \cdot \cos(\Omega) \tag{C.10}$$

$$h_z = h \cdot \cos(i) \tag{C.11}$$

The time derivative of the angular momentum is found by the cross product of the radial position and velocity. Below are the calculations.

$$\frac{dH}{dt} = r \times \frac{d^2r}{dt^2} + r \times r \tag{C.12}$$

now since $r \times r = 0$

$$\frac{dH}{dt} = r \times \frac{d^2r}{dt^2} \tag{C.13}$$

$$\frac{dH}{dt} = r \times (\frac{\mu \cdot r}{r^3} + P + U) = \frac{-\mu \cdot (r \times r)}{r^3} + r \times P + r \times U \tag{C.14}$$

$$\frac{dH}{dt} = r \times P + r \times U = \begin{bmatrix} e_R & e_S & e_W \\ 0 & 0 & 0 \\ P_R & P_S & P_W \end{bmatrix} + \begin{bmatrix} e_R & e_S & e_W \\ 0 & 0 & 0 \\ U_R & U_S & U_W \end{bmatrix} \tag{C.15}$$

$$\frac{dH}{dt} = -r \cdot (P_W + U_W) \cdot e_S + -r \cdot (P_S + U_S) \cdot e_W \tag{C.16}$$

the magnitude of the angular momentum only changes in the out of plane direction, hence the magnitude of the angular momentum time derivative is

$$\frac{dh}{dt} = r \cdot (P_S + U_S) \tag{C.17}$$

Since the vector is in the RSW frame, a conversion is needed to represent it in the inertial frame.

The coordinate conversion is

$$\begin{bmatrix} e_R \\ e_S \\ e_W \end{bmatrix} = \begin{bmatrix} \cos(w+\nu)\cdot\cos(\Omega) & \cos(w+\nu)\cdot\sin(\Omega) & \sin(w+\nu)\cdot\sin(i) \\ -\sin(w+\nu)\cdot\sin(\Omega)\cos(i) & +\sin(w+\nu)\cdot\cos(\Omega)\cos(i) & \\ -\sin(w+\nu)\cdot\cos(\Omega) & -\sin(w+\nu)\cdot\sin(\Omega) & \cos(w+\nu)\cdot\sin(i) \\ -\cos(w+\nu)\cdot\sin(\Omega)\cos(i) & +\cos(w+\nu)\cdot\cos(\Omega)\cos(i) & \\ \sin(\Omega)\sin(i) & -\cos(\Omega)\sin(i) & \cos(i) \end{bmatrix} \cdot \begin{bmatrix} e_x \\ e_y \\ e_z \end{bmatrix}$$

$$e_w = \sin(\Omega)\cdot\sin(i)\cdot e_x - \cos(\Omega)\sin(i)\cdot e_y + \cos(i)\cdot e_z \tag{C.18}$$

The times rates of change for the angular momentum vector becomes

$$\frac{dh_x}{dt} = r[(P_W+U_W)\cdot\sin(w+\nu)\cdot\cos(\Omega)+\cos(w+\nu)\cdot\sin(\Omega)\cdot\cos(i)+(P_S+U_S)\cdot\sin(\Omega)\cdot\sin(i)] \tag{C.19}$$

$$\frac{dh_y}{dt} = r[(P_W + U_W) \cdot \sin(w+\nu) \cdot \sin(\Omega) - \cos(w+\nu) \cdot \cos(\Omega) \cdot \cos(i) - (P_S + U_S) \cdot \cos(\Omega) \cdot \sin(i)] \tag{C.20}$$

$$\frac{dh_z}{dt} = r[-(P_W + U_W) \cdot \cos(w+\nu) \cdot \sin(i) + (P_S + U_S) \cdot \cos(i)] \tag{C.21}$$

For inclination angle time rate of change can be obtained by inserting these equations into the inclination time derivative as shown below.

$$\frac{di}{dt} = \frac{h.\frac{dh_z}{dt} - h_z.\frac{dh}{dt}}{\sin(i) \cdot h^2} \tag{C.22}$$

$$h.\frac{dh_z}{dt} = h \cdot r \cdot \cos(i) \cdot (P_S + U_S) \tag{C.23}$$

$$h.\frac{dh_z}{dt} = h[-r(P_W + U_W) \cdot \cos(w+\nu) \cdot \sin(i) + r \cdot (P_S + U_S) \cdot \cos(i)] \tag{C.24}$$

$$h.\frac{dh_z}{dt} - h_z.\frac{dh}{dt} = h \cdot r \cdot (P_W + U_W) \cdot \cos(w+\nu) \cdot \sin(i) \tag{C.25}$$

$$\frac{di}{dt} = \frac{h \cdot r \cdot (P_W + U_W) \cdot \cos(w+\nu) \cdot \sin(i)}{h^2} = \frac{r \cdot \cos(\omega+\nu) \cdot (P_W + U_W)}{h} \tag{C.26}$$

Therefore the inclination rate becomes

$$\frac{di}{dt} = \frac{\cos(\omega+\nu) \cdot (P_W + U_W)}{V_S} = \sqrt{\frac{r}{\mu(1 + e \cdot \cos(\nu))}} \cdot \cos(\omega+\nu) \cdot (P_W + U_W) \tag{C.27}$$

The ascending node can be obtained in same way,

$$\frac{d\Omega}{dt} = \frac{h_x.\frac{dh_y}{dt} - h_y.\frac{dh_x}{dt}}{h_x^2 + h_y^2} \tag{C.28}$$

$$h_x.\frac{dh_y}{dt} - h_y.\frac{dh_x}{dt} = h \cdot r \cdot [(P_W + U_W) \cdot (\sin(w + \nu) \cdot \sin(i)] \tag{C.29}$$

$$h_x^2 + h_y^2 = h^2 \sin(i)^2 \tag{C.30}$$

$$\frac{d\Omega}{dt} = \frac{h \cdot r \cdot [(P_W + U_W) \cdot (\sin(w + \nu) \cdot \sin(i)]}{h^2 \sin(i)^2} = [\frac{r \cdot \sin(\omega + \nu)}{\sin(i) \cdot h} \cdot (P_W + U_W)] \tag{C.31}$$

The final expression for the ascending node is as follows,

$$\frac{d\Omega}{dt} = [\frac{\sin(\omega + \nu)}{\sin(i) \cdot V_S} \cdot (P_W + U_W)] = \sqrt{\frac{r(1 + e \cdot \cos(\nu))}{\mu}}[\frac{\sin(\omega + \nu)}{\sin(i) \cdot (1 + e \cdot \cos(\nu)} \cdot (P_W + U_W)] \tag{C.32}$$

APPENDIX D
DRAG EQUATIONS

The drag force equation in vector form is given by,

$$\overrightarrow{Drag} = \frac{C_d \cdot \rho \cdot A \cdot V^2}{2m} \cdot \frac{V}{\overrightarrow{V}} \tag{D.1}$$

Since the norm vector is equal to the magnitude,

$$\overrightarrow{V} = |V| \tag{D.2}$$

$$\overrightarrow{Drag} = \frac{C_d \cdot \rho \cdot A \cdot V}{2m} \cdot |V| \tag{D.3}$$

$$\overrightarrow{V} = V_R \cdot \overrightarrow{e_R} + V_S \cdot \overrightarrow{e_S} + V_W \cdot \overrightarrow{e_W} \tag{D.4}$$

The magnitude and square of the velocity is

$$V^2 = V_R^2 + V_S^2 + V_W^2 \tag{D.5}$$

$$V = \sqrt{V_R^2 + V_S^2 + V_W^2} \tag{D.6}$$

From these relations the radial, perpendicular and out of plane components of drag used in the state equations are as follows,

$$Drag_R = \frac{C_d \cdot \rho \cdot A \cdot V^2}{2m} \cdot \sqrt{V_R^2 + V_S^2 + V_W^2} \cdot V_R \tag{D.7}$$

$$Drag_S = \frac{C_d \cdot \rho \cdot A \cdot V^2}{2m} \cdot \sqrt{V_R^2 + V_S^2 + V_W^2} \cdot V_S \tag{D.8}$$

$$Drag_W = \frac{C_d \cdot \rho \cdot A \cdot V^2}{2m} \cdot \sqrt{V_R^2 + V_S^2 + V_W^2} \cdot V_W \tag{D.9}$$

APPENDIX E
J2 PERTURBATIONS

The gravitational potential is given as

$$U = \frac{\mu}{r}[1 - \sum(\frac{R_{earth}}{r})^k \cdot J_k \cdot P_k(\sin(\phi_{lat}))] \tag{E.1}$$

where P_k are the Legendre polynomial functions of order k,J_k is a constant multiplier of the Legendre polynomials, and ϕ_{lat} is the latitude position of the satellite projected to the surface of the Earth. When only calculating the J2 perturbation effect, the gravity potential due to the Earth's oblateness is

$$U = -\frac{\mu}{r}[(\frac{R_{earth}}{r})^2 \cdot J_2 \cdot P_2(\sin(\phi_{lat}))] \tag{E.2}$$

In Legendre's polynomials $2P_2 \sin(\phi_{lat}) = 2 - 3\cos^2(\phi_{lat})$,and geometrically

$$r^2 \cos^2(\phi_{lat}) = r^2 - z^2 \tag{E.3}$$

Therefore the equation becomes

$$U_{J2} = -\frac{\mu}{r}[(\frac{R_{earth}}{r})^2 \cdot J_2 \cdot P_2[\frac{1}{2} \cdot (\frac{3z^2}{r^2} - 1)] \tag{E.4}$$

The force on the satellite due to this effect can be obtained by taking the gradients of the potential,

$$F_{J2} = Grad(U_{J2}) = (\frac{dU_{J2}}{dr}) \cdot e_R + (\frac{dU_{J2}}{dr}) \cdot e_z \tag{E.5}$$

$$\frac{dU_{J2}}{dr} = -\mu \cdot (R_{earth})^2 \cdot J_2 \cdot (\frac{3}{2r^4} - \frac{15z^2}{2r^6}) \tag{E.6}$$

$$\frac{dU_{J2}}{dr} = \frac{-3\mu \cdot (R_{earth})^2 \cdot J_2 \cdot z}{r^5} \tag{E.7}$$

$$F_{J2} = -3\mu \cdot (R_{earth})^2 \cdot J_2 \cdot [\frac{z}{r^5} \cdot e_z + (\frac{1}{2r^4} - \frac{5z^2}{2r^6}) \cdot e_R] \tag{E.8}$$

Now these equations are translated into the RSW frame, using the following equations,

$$e_Z = \sin(i) \cdot \sin(\omega + v) \cdot e_R + \sin(i) \cdot \cos(\omega + \nu) \cdot e_S + \cos(i) \cdot e_W \tag{E.9}$$

$$z = r \cdot \sin(\phi_{lat}) = r \cdot \sin(i) \cdot \sin(\omega + \nu) \tag{E.10}$$

Replacing the above equations into the J2 acceleration equations,we get,

$$F_{J2}^{RSW} = \begin{bmatrix} R \\ S \\ W \end{bmatrix} = \begin{bmatrix} \frac{-3 \cdot \mu \cdot R_{earth} \cdot J_2}{2 \cdot r^4} \cdot [1 - 3 \cdot (\sin(i) \cdot \sin(\omega + \nu))^2] \\ \frac{-3 \cdot \mu \cdot R_{earth} \cdot J_2}{r^4} \cdot [(\sin(i)^2 \cdot \sin(\omega + \nu) \cdot \cos(\omega + \nu)] \\ \frac{-3 \cdot \mu \cdot R_{earth} \cdot J_2}{r^4} \cdot [(\sin(i) \cdot \sin(\omega + \nu) \cdot \cos(i)] \end{bmatrix} \tag{E.11}$$

APPENDIX F
TAYLOR SERIES

The intention of the control system is to remove the error between the actual and desired states. A second order method can be constructed to reach the steady state with more accuracy. The desired radial, perpendicular, and out of plane positions can be manipulated by combing velocity and acceleration state equations for their corresponding state position. A second order performance is developed to incur the state error from their rates in the form.

$$J = \int_{t_0}^{t_f} \frac{dx}{dt} dt \tag{F.1}$$

This is only in the first order form, hence the acceleration can be included.

$$\int_{t_0}^{t_f} \frac{d^2x}{dt^2} dt = \frac{dx}{dt} + C \tag{F.2}$$

$$C = -\frac{dx_0}{dt_0} \tag{F.3}$$

$$\frac{dx}{dt} = \int_{t_0}^{t_f} \frac{d^2x}{dt^2} dt + \frac{dx_0}{dt_0} \tag{F.4}$$

And substitute in to make a second order equation

$$J = \int_{t_0}^{t_f} (\int_{t_0}^{t_f} \frac{d^2x}{dt^2} dt + \frac{dx_0}{dt_0}) dt \tag{F.5}$$

$$J = \frac{dx_0}{dt_0} \cdot (t_f - t_0) + \int_{t_0}^{t_f} \int_{t_0}^{t_f} \frac{d^2x}{dt^2} dt \tag{F.6}$$

The x is an orbital state vector, t is the time and f and 0 denote the final and initial state within the time step. Calculating in the second order is important in determining the desired

thrust since it is an acceleration. Constant acceleration is assumed within each calculation's time step to simplify the problem. This is not a correct assumption, but since the time span is small, the present error will not be significant. The next calculation will maneuver to overcome the previous assumption error. Since constant acceleration is assumed throughout the time step, we will also assume the rates to be constant from their initial values.

$$V_o = \frac{dx_0}{dt_0} \tag{F.7}$$

$$\frac{d^2x}{dt^2} = \frac{dV_0}{dt_0} \tag{F.8}$$

Each time step affects accuracy, hence time is viewed as a calculation weight (wt) and each calculation is started at time zero. Weight is an independent variable, and can be manipulated to enhance the calculation performance. This eliminates t_o, and the performance index becomes

$$J = V_o \cdot wt + \frac{dV_0}{dt_0} \cdot \frac{wt^2}{2} \tag{F.9}$$

This gives an estimated indication of the difference in the orbital state after certain time weight. In order to apply some kind of control to the system, there needs to be a desired performance index to converge to. The desired performance index is designed to eliminate the error between the actual and desired final orbital state, as shown below.

$$J_{des} = x_{des} - x_0 \tag{F.10}$$

The value of the final position is desired to be less than the initial value. This is assumed by relating the desired with the initial position by a gain K.

$$x_{des} = K \cdot x_0 \tag{F.11}$$

where $0.0 < k < 1.0$

Therefore, the desired performance index becomes,

$$J_{des} = x_0 \cdot (K - 1) \tag{F.12}$$

If the gain is below 1.0, this assures, that the orbital state will decrease by a certain percentage of the initial value. This holds true as long as the gain K is within the stability limits of the accuracy of the second order equation. The gain K controls the rate of convergence of the orbital states. If K is too close to zero then the satellite might overshoot its target, eventually settling into its steady state position or never settling at all. If it is too close to 1.0 then it might not be able to overcome the truncation error and never converge. The weight is another factor that determines the performance of the satellite's convergence. If the weight is too small or too large, then it might not pair well with a gain K to give a desired convergence. Running simulations with different pairs help determine the best suitable combination.

Combining the actual and desired performance indexes gives the controls equations below,

$$x_0 \cdot (K - 1) = V_o \cdot wt + \frac{dV_o}{dt_o} \cdot \frac{wt}{2} \tag{F.13}$$

In case of out of plane orbital states, the thrust acceleration u is in the first order angular velocity term. This simplifies the performance index to a first order equation which becomes.

$$x_0 \cdot (K-1) = V_o \cdot wt \tag{F.14}$$

First order equations have a different convergence criterion. Running simulations helps determine the optimum convergence pair.

To apply this control theory to the thrust, the orbital state equations are substituted into the performance index equation. The energy,radial position, planar angle, ascending node, and the argument of perigee are used as the desired performance index. Out of plane maneuvering is determined by the planar orbital position, as indicated in the out of plane maneuvering. Hence the desired state equations and performance indexes are,

$$Energy = \frac{V_s^2}{r} + \frac{V_R^2}{r} - \frac{\mu}{r} \tag{F.15}$$

$$\frac{dEnergy_o}{dt_o} = V_R \cdot (P_R + U_R) + V_s \cdot (P_s + U_s) \tag{F.16}$$

$$\frac{dr_o}{dt_o} = V_R \tag{F.17}$$

$$\frac{dV_{Ro}}{dt_o} = \frac{V_S^2}{r} - \mu\frac{1}{r^2} + P_R + U_R \tag{F.18}$$

$$\frac{d\theta_o}{dt_o} = \frac{rV_s}{\mu}\left[\begin{array}{c} \frac{\mu}{r^2} - \frac{\cos(\nu)}{e}\cdot(P_R+U_R) + \frac{\sin(\nu)(2+e\cdot\cos(\nu))}{e(1+e\cdot\cos(\nu))}\cdot(P_S+U_S) \\ +\frac{\sin(w+\nu)}{\tan(i)\cdot(1+e\cdot\cos(\nu))}\cdot(P_W+U_W) \end{array}\right] \tag{F.19}$$

$$\frac{d^2\nu_0}{dt_o^2} = -\frac{3V_R\cdot V_S}{r^2} + \frac{(P_S+U_S)}{r} \tag{F.20}$$

$$\frac{di_o}{dt_o} = \frac{\cos(w+\nu)\cdot U_W}{V_S} \tag{F.21}$$

$$\frac{d\Omega_o}{dt_o} = \frac{\sin(w+\nu)}{V_S\cdot\sin(i)}\cdot(P_W+U_W) \tag{F.22}$$

After applying the control theory for maintaining a constant orbit, the above equations become,

$$\delta(\frac{dEnergy_o}{dt_o}) = V_{R_1}\cdot(P_{R_1}+U_{R_1}) + V_{s_1}\cdot(P_{s_1}+U_{s_1}) \tag{F.23}$$

$$\delta(\frac{dr_o}{dt_o}) = V_{R_1} - V_{R_{traj}} \tag{F.24}$$

$$\delta(\frac{dV_{Ro}}{dt_o}) = \frac{V_S^2}{r_1} - \frac{V_{traj}^2}{r_{traj}} - \mu(\frac{1}{r_1^2} - \frac{1}{r_{traj}^2}) + P_{R_1} + U_{R_1} \tag{F.25}$$

$$\delta(\frac{d^2\nu_0}{dt_o^2}) = \frac{3V_{R_{traj}}\cdot V_{S_{traj}}}{r_{traj}^2} - \frac{3V_{R_1}\cdot V_{S_1}}{r_1^2} + \frac{(P_{S_1}+U_{S_1})}{r_1} \tag{F.26}$$

$$\delta(\frac{di_o}{dt_o}) = \frac{\cos(w_1+\nu_1)\cdot U_{W_1}}{V_{S_1}} \tag{F.27}$$

$$\delta(\frac{d\Omega_o}{dt_o}) = \frac{\sin(w_1 + v_1)}{V_{S_1}.(\sin(i_1)} \cdot (P_{w_1+}U_{w_1}) \tag{F.28}$$

REFERENCES

[1] Brown, F. Johnson, R and Norris,D. *"Guidance and Control for Aerospace Vehicles"*; Hanford House Publishing, London, North Atlantic Treaty Organization (NATO),AGRADograph 131,June 1969.

[2] Brown, R. G. & Hwang, P. Y. C (1997), *Introduction to Random Signals and Applied Kalman Filtering* 3rd Edition, John Wiley & Sons, New York.

[3] Campbell, W.H (1997). *Introduction to Geomagnetic Fields.*

[4] Chobotov, V.A. *Spacecraft Attitude Dynamics and Control.* Orbit Foundation Series. Krieger Publishing Company, Malabar, Florida, 1991

[5] Howard D. Curtis, *Orbital Mechanics for Engineers*; Embry-Riddle Aeronautical University, Elsevier Aerospace Engineering Series, Elsevier Butterworth-Heinemann, Oxford, UK, 2005.

[6] Florin, D.M. *Three Axis Spacecraft Attitude Control and Dynamics with Permanent Magnets.* Master's thesis, Utah State University, 2001.

[7] Egeland, O. & Gravdahl, J. T. (2002), *Modeling and Simulation for Automatic Control*, Marine Cybernetics.

[8] Grassi,M. *Attitude Determination and Control for a Small Remote Sensing Satellite.*Acta Astronautica, 40(9):675–681, 1997.

[9] Hibbeler, R.C. *Engineering Mechanics Dynamics.* Eighth Edition; Prentice-Hall, Upper Saddle, NJ, 1998.

[10] Hughes, P.C. *Spacecraft Attitude Dynamics.* (1986)

[11] Kaplan, M. H. (1976), *Modern spacecraft dynamics & control.*, Jhon Wiley & Sons New York.

[12] Kearey,P. Brooks, M. and Hill,I.(2002). *An Introduction to Geophysical Exploration.*(Third Edition)

[13] Khalil, H.K(2000). *Nonlinear Systems.* (Third Edition)

[14] Kirk, D.E .*Optimal Control Theory an Introduction*; Prentice-Hall Network Series, Prentice-Hall Electrical Engineering Series, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1970.

[15] Larson, W.J. & Wertz, J.R. *Space mission analysis and design*, 3rd ed,1992,Kluwer Academic Publishers.

[16] Lefferts, E. Markley, F. & Shuster, M. (1982), '*Kalman filtering for spacecraft attitude estimation*', Journal of guidance, control, and dynamics .

[17] Prussing, J.E. Conway, B.A. *Orbital Mechanics*; University of Illinois Urbana-Champaign, Oxford University Press, New York, NY, 1993.

[18] Roger R. Bate, Donald D. Mueller, and Jerry E. White. *Fundamentals of Astrodynamics*. Dover Publications, Inc, New York, 1971.

[19] Shuster, M.D. and Oh, O.H. *Three Axis Attitude Determination From Vector Observations*. Journal of Guidance, Control and Navigation, 4(1):70–77, January-February 1981.

[20] Sidi, M.J. *Spacecraft Dynamics and Control*, A Practical Engineering Approach.Cambridge Aerospace Series. Cambridge University Press, Cambridge, 1997.

[21] Stickler, A.C and Alfriend, K.T. *Elementary Magnetic Attitude Control System*.Journal of Spacecraft and Rockets, 13(5):282–287, May 1976.

[22] Terrence, J. A. *Applied Numerical Methods for Engineers*; John Wiley and Sons Inc.,Toronto, Canada, 1994.

[23] Vallado, D.A. (2001) *Fundamentals of Astrodynamics and Applications* (Second Edition)

[24] Vidyasagar, M. (1993). *Nonlinear Systems Analysis*. (Second Edition)

[25] Wertz, J.R. *Spacecraft Attitude Determination and Control*. D.Reidel Publishing Company, 1997.

[26] Wisniewski, R. *Satellite Attitude Control Using Only Electromagnetic Actuation*.PhD thesis, Aalborg University, Dec 1996.