ATTITUDE CONTROL SYSTEM FOR THE HIGH ENERGY TRANSIENT EXPERIMENT SMALL SATELLITE

by

Daniel H. Chang

B.S. Aeronautics and Astronautics
Massachusetts Institute of Technology
(1989)

Submitted to the Department of Aeronautics and Astronautics in Partial
Fulfillment of the Requirements for the Degree of

MASTER OF SCIENCE

in

AERONAUTICS AND ASTRONAUTICS

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 1991

© Daniel H. Chang, 1991

Signature of Author:_____
Department of Aeronautics and Astronautics
10 May 1991

Certified by:_____
Professor Andreas von Flotow
Associate Professor, Department of Aeronautics and Astronautics

Accepted by:_____
Professor Harold Y. Wachman
Chairman, Department Graduate Committee

Aero

Attitude Control System for the High Energy Transient Experiment Small Satellite

by

Daniel H. Chang

ABSTRACT

The High Energy Transient Experiment (HETE) is a NASA funded project managed by the MIT Center for Space Research. The goal is to provide a comprehensive study of high energy, short duration γ-ray, X-ray and UV burst phenomena with wide band spectral measurements of events as well as precise and near real-time determination of their source locations using a small, low Earth orbit satellite.

On-orbit attitude control requirements of the satellite are driven by power and payload needs and include maintenance of a sun-pointing attitude and tight regulation of drift rates. A control system is designed using a momentum bias/magnetic torquing strategy. It is shown that specifications can be met using only cheap and readily available components, with one of the science instruments doubling as a high quality (albeit slow) star tracker for further cost reduction. LQG with control gains continuously scheduled to the local field vector direction is found to be a conceptually simple yet effective control design technique for circumventing the limited controllability associated with magnetic torquing.

This thesis presents a preliminary control design study. The disturbance torque environment and the Earth magnetic field are modelled, and controllers are designed for on-orbit operations in all field arrangements, as well as for housekeeping tasks such as momentum management. Performance and robustness issues are addressed analytically and through extensive simulation. The aim is to provide solid ground for detailed design and development by evolving realistic specifications for attitude control system hardware and by achieving reasonable estimates of expected performance.

Thesis Supervisor: Dr. Andreas von Flotow
            Title: Associate Professor of Aeronautics and Astronautics, MIT

Acknolowdgements

It seems pretentious somehow to produce a document in a field that I knew next to nothing about only two years ago, but I suppose that's the whole point of a thesis. Nevertheless, those who helped along the way should be acknowledged:

Table of Contents

I. Introduction

1.1 The HETE Mission

HETE is an acronym for "High-Energy Transient Experiment," and is a NASA sponsored mission to investigate short duration (on order of 1 sec.) transient astronomical phenomena in the X-ray, γ-ray, and UV bands. The mission proposal to NASA ([Rick87]) states as mission objectives the following:

- Primary Objective- γ-Ray Burst Sources

  - Understand the basic nature of these enigmatic objects through precise source locations (~ 6 arcsec) and spectral studies spanning up to 6 decades of photon energy.

  - Probe the underlying physics of these sources under some of the most exotic conditions of temperature, density, and magnetic field found anywhere in the universe.

- Secondary Objectives

  - X-Ray Burst and Transient Sources

    - Continuous monitoring of X-ray burst sources over a large fraction of the celestial sphere.
    - Serendipitous discovery of new classes of transient X-ray phenomena.

  - UV and Optical Transients

    - Continuous monitoring of known classes of optical transients (e.g. flare stars) over a large fraction of the celestial sphere.
    - Serendipitous discovery of new classes of transient optical and UV phenomena.

Theoretical background, observational history, and scientific justification of the study of high energy transients may be found in [Rick87] and its references.

From an observational hardware point of view, HETE is a small, low Earth orbit satellite with a preliminary design mass of 250 lbm maximum. A momentum-bias configuration with magnetic torque de-saturation and active sun tracking/nutation

damping is the current design[1]. The spacecraft itself will be discussed in Chapter 2. The following instruments comprise the payload:

- Omnidirectional γ-ray Spectrometer: 2π sr. field of view
- Wide-Field X-ray Monitor: ~ 2.5 sr. total field of view
- UV CCD Camera Array: ~ 2.5 sr. total field of view

The mission objective of observing a "large fraction of the celestial sphere" is accomplished with an inertially stabilized, sun pointing mission attitude. The instruments are then mounted to observe anti-sun. Since the UV instrument will not function in the presence of a sun-lit Earth, the choice a low-inclination orbit is made to ensure orbit-daily occultation of the sun. Thus we have an instrument field-of-view cone centered on the ecliptic plane which sweeps 360° about the solar north pole over the course of a year, as Figure 1.1 shows:
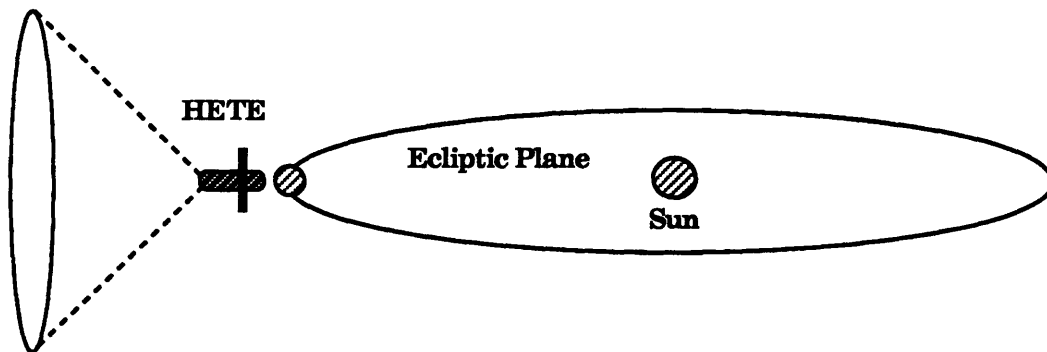
Figure 1.1: HETE field of view as function of time of year. The cone width is not specified but, as seen from the preliminary instrument specification above, is limited by the X-ray and UV instruments. It is in this sense that a "large fraction of the celestial sphere" is observed.

Downlink consists partly of a low-bandwidth, continuously broadcast UHF channel which can be monitored by small, portable receivers connected to personal computers. It is envisioned that many such receive-only secondary ground will be distributed at participating observatories and offices around the world and communicate on the ground via fax. A high bandwidth S-band up/downlink is used for high data rate

---

[1]Much of the preliminary spacecraft bus design and analysis work is to be credited to Aero/Astro Inc., Herndon, VA.

transfers and will communicate with three primary ground stations distributed on the ground track of the orbit. Ground communications with the primary ground stations is via Internet.

Satellite on-orbit operation consists of continuous monitoring of the instruments and storage of the most recent volume of data in on-board memory. Status of the satellite and instruments is continuously broadcast on the low-bandwidth channel. Should it be determined that a significant event has been observed, the data memory is saved for broadcast on the high-bandwidth channel at the next available opportunity. Secondary ground stations are also alerted to the presence and location of the event through the low-bandwidth downlink. Thus, in summary, HETE is both an instrument package in its own right and a sentinel to alert ground observatories of the presence of high energy transient events.

A rough breakdown of the HETE responsibilities follows. These institutions comprise the core group of the HETE project.

Project Management:	MIT Center for Space Research
Spacecraft Contractor:	Aero/Astro Inc., Herndon, VA
γ-ray Instrument Development:	CNES (France) University of California at Berkeley
X-ray Instrument Development:	Los Alamos National Labs ISAS (Japan)
UV Instrument Development:	MIT Center for Space Research MIT Lincoln Labs
Theoretical Development:	University of Chicago University of California at Santa Cruz
Orbital Operations:	MIT Center for Space Research

1.2 Mission Profile and ACS Specifications

1.2.1 Choice of Orbit

The remainder of this thesis assumes the use of a 28° inclination circular orbit at an altitude of 300 nautical miles (556 Km.). The selection is made based on desired

lifespan, instrument sensitivity to charged particles at higher altitudes/inclinations, and orbital illumination fraction.

1.2.2 Launch and Orbit Raising

The choice of launch vehicle is not yet determined as of this writing. The original concept of a Shuttle GAS-can launch has the initial appeal of low cost. However, this is offset by the need for orbit raising capability (Shuttle altitude is 160 nautical miles) and the increased schedule uncertainty and bureaucratic complexity of a Shuttle launch. Furthermore, the 1990 Augustine Report to NASA recommends the use of expendable launchers for missions not requiring a human presence, a policy NASA is currently trying to implement. Therefore, it is likely that an expendable launcher (Pegasas or Delta) will be used. Since either launch vehicle has the capability to place 250 lbm of payload into the desired orbit [Wert91], orbit raising would no longer be necessary, greatly simplifying spacecraft design and operations.

An on-board propulsive capability based on compressed Freon propellant and low-thrust, long duration operation has been proposed by Aero/Astro Inc. Attitude control in the form of thrust vector alignment and nutation damping is necessary for the proposed orbit raising maneuver. This issue is not addressed in this thesis.

1.2.3 Attitude Acquisition

Acquisition of the mission attitude after achieving the proper orbit can be broken into a de-tumble phase and a slew to the sun-vector. Momentum wheel spin-up must also take place. The amount of de-tumbling necessary depends on the launch vehicle choice. In any case, the described maneuvers are standard applications of magnetic control laws know in the literature. Discussion is therefore deferred to Chapter 6.

1.2.4 Orbit Night Control

Orbit night is the primary observation half of the orbit, since, as mentioned, the instruments require ambient darkness. It is also the mission phase with the most stringent ACS specification: the instruments require inertial rates are to be maintained below 2 °/hr about each body axis (1 °/hr = $\pi/(3600 \bullet 180)$ rad/s). As the instruments are wide field-of-view, there is no strict pointing requirement.

The star-indentification algorithms in the UV cameras provide attitude information during orbit night.

1.2.5 Orbit Day Control

The most important spacecraft operation during orbit day is battery re-charging. Therefore, the ACS specification of maintaining the solar panel normal to within a 5° cone of the sun vector is given. Additionally, it is necessary to maintain inertial rates such that, upon transitioning to orbit night, the star-ID algorithms behind the UV instrument can achieve an attitude solution. This maximum rate is not yet determined, but design of the orbit day controller shall proceed with the target specification of 20 °/hr.

The other major ACS housekeeping task during orbit day is wheel momentum maintenance. Since the concept is to use magnetic torquers for this, no expendables are involved, making daily (orbit daily) desaturation a viable option.

Sun-sensors provide attitude information during orbit day.

1.2.6 Slew Maneuver Control

To provide an additional degree of observational freedom, a rapid-slew maneuver capability about the momentum wheel axis is proposed. The concept is to slew the sun vector from the nominal direction to some azimuth on the plane of the momentum wheel, then transition to orbit night in this new attitude with the instruments now pointed in a new direction. The reverse slew would be performed at the transition to orbit day to maximize exposure of the solar panels for battery re-charge. By also controlling the roll-angle of the wheel axis with respect to the ecliptic plane, the entire anti-sun celestial hemisphere can be brought within the instruments' fields of view, should the observation need arise.

1.3 ACS Design Philosophy

From another point of view, the HETE project is also a demonstration of the cost effectiveness of a small, mission specific spacecraft development effort involving a tightly knit "university/industry team," with little NASA intervention. Distillation of this to the specific task of ACS development leads to a design philosophy incorporating the following ideas:

• Little redundancy in key hardware is sought.

• The use of proven technology is stressed, using off-the-shelf components where possible. In the proposed ACS, the UV camera is the only component without much space-flight heritage, this disadvantage being amply offset by the elimination of the need for a separate star tracker or IMU.

• The amount of analysis on the ground is balanced against the possibility to tune the performance of the ACS in orbit. It is further noted that the operators of HETE at MIT CSR are experienced with the SAS-3 program of the late 1970's, and will also be intimately involved in the spacecraft design and development.

1.4 Thesis Overview

This thesis conducts a preliminary design of the HETE observation phase ACS, with the intent of demonstrating that the proposed hardware complement is capable of supporting the mission profile as outlined. A summary of the chapter contents follows:

• Chapter 2: details the proposed ACS hardware complement and develops sensor / actuator models to be used in subsequent design and analysis.

• Chapter 3: relevant orbital environment elements such as disturbance torques, the geomagnetic field, and lighting conditions are analyzed to support control design and provide for realistic control performance simulations.

• Chapter 4: control actuator strategies specific to the mission requirements and hardware complement are evolved, and dynamics of the spacecraft and kinematics of the sensors are examined.

• Chapter 5: detailed control design is carried out for the orbit night, orbit day, and slew maneuver phases; performance and stability robustness are analyzed and simulation results are presented.

• Chapter 6: references the literature for control strategies relevant to initial attitude acquisition; discusses additional work necessary to complete the slew maneuver control design.

II. Description of Hardware

This chapter discusses briefly the proposed overall design of the HETE spacecraft[1], and in greater detail the attitude control system (ACS) actuators and sensors. Models for the ACS hardware are developed for later use in control design.

The following notation conventions are used for vector quantities throughout this thesis:

$$\underline{V}_A = C_{AB}\,\underline{V}_B$$

- Geometric ("Gibbs") vectors : **V** in **boldface**
- $\underline{V}_A$ is the arbitrary vector **V** expressed in reference frame A,
- $\underline{V}_B$ is the arbitrary vector **V** expressed in reference frame B,
- C_{AB} is the 3 X 3 direction cosine matrix relating them, $C_{BA} = (C_{AB})^{-1} = (C_{AB})^T$
- Angular velocity of frame B with respect to A: $^A\omega^B$
- Derivative of vector **V** with respect to frame A: $^A\dfrac{d}{dt}(V)$
- Unit vector x in frame A: A_1
- Shorthand for frame A: $\mathcal{F}_A$ This is compatible with vectrix notation in [Hugh86], although vectrix notation is not used.

2.1 Spacecraft Bus

Figure 2.1 is an isometric view of the currently proposed design. The body-fixed orientation reference frame, which we define to be $\mathcal{F}_B$, is defined in the figure.

The HETE bus is an exoskeletal design with loads supported mostly by the boxes which enclose the various components. Designed initially for a Shuttle GAS-can launch, the spacecraft envelope is limited to a cylinder 35" high and 19" diameter, and the spacecraft mass is limited to 250 lbm. The instrument package sits on the $-B_3$ direction from the spacecraft bus. A planar view of the instrument package layout in the $+B_3$ direction is given in Figure 2.2.

The 4 solar paddles are to be constructed of carbon fiber and are hinged at their bases to the bus. They launch folded (tips pointing along $-B_3$), with the outer surface curved to better fit the inner surface of a GAS cannister. The current plan calls for rigid locking of

---

[1]Much of the preliminary spacecraft bus design work has been performed by AeroAstro Inc., Herndon, VA. Instrument preliminary design responsibility breaks down according to the organization chart of Chapter 1. Mechanical integration of instrument package is performed at MIT CSR.

the paddles in the deployed position after deployment. Some thought may be given to the possibility of introducing a spring/dashpot mechanism at the hinge for the purpose of nutation damping, but the control design in this thesis proceeds assuming undamped nutation modes. The ellipsoidal structure at the $+B_3$ end of the bus is the proposed Freon 114 tank for the orbit raising phase of the mission. Since the desired orbit is 300 nautical miles circular, deployment from a Shuttle would require orbit raising. However, it is likely at the time of writing that a decision to launch on an expendable vehicle (Pegasus or Delta) would eliminate such a need, giving all the accompanying benefits of weight savings and reduced mission complexity. Attitude control for the orbit raising phase is not considered in this thesis, as mentioned in Chapter 1.
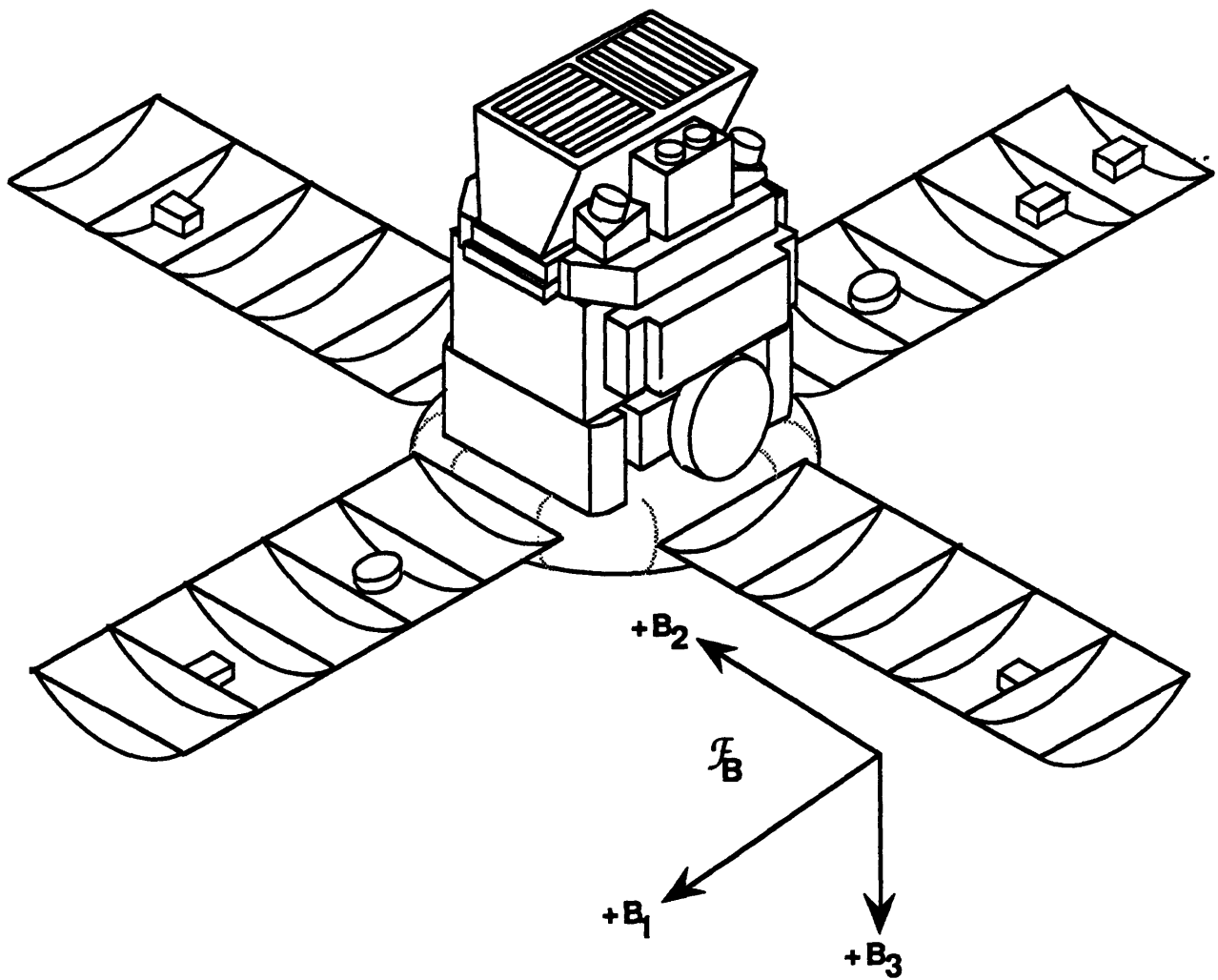
Figure 2.1: Isometric view of HETE, with the body fixed orientation reference frame defined. Drawing courtesy of AeroAstro Inc.
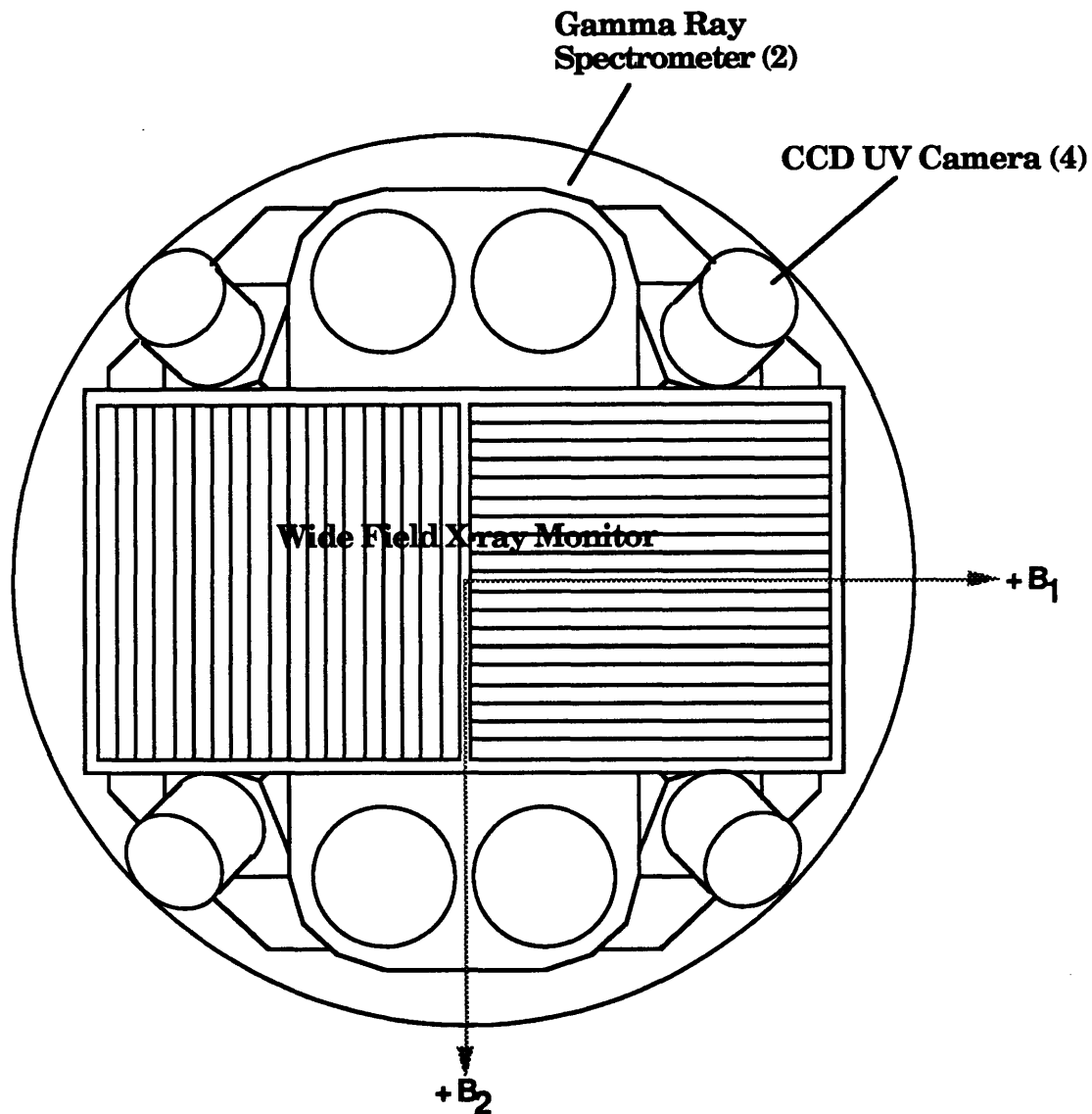
Figure 2.2: planar view of instrument package. All instruments see in the -B₃ direction (out of page). The UV instrument, which doubles as attitude instruments during orbit night, is a cluster of 4 CCD cameras with slightly overlapping fields of view. Drawing courtesy of AeroAstro Inc. and MIT CSR.

The desired spacecraft attitude is inertially stabilized and sun pointing. The terms azimuth and elevation are defined to specify the direction of the vector pointing to the sun (henceforth called the sun vector) in $\mathcal{F}_B$. Since the nominal sun vector is in the B_1-B_3 plane and parallel to B_3, azimuth and elevation are 0 for B_3. Otherwise, they obey the right hand rule about $+B_2$ and $+B_1$ respectively, as shown in Figure 2.3.
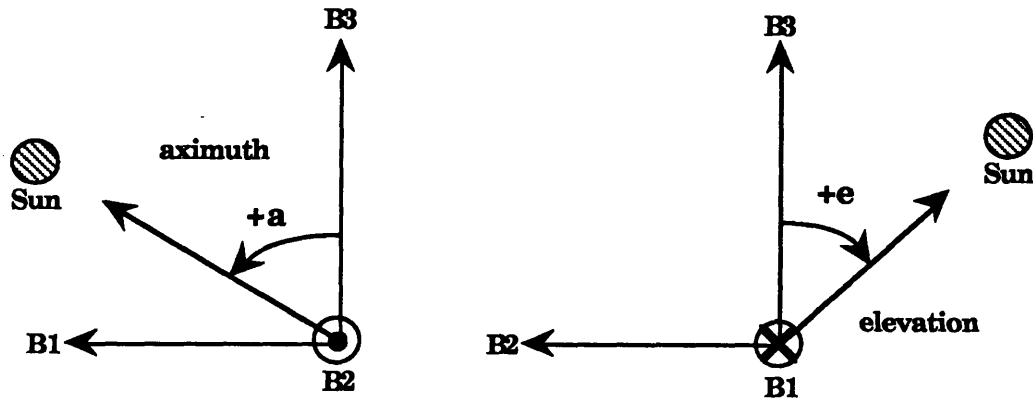
Figure 2.3: sun-vector direction defined by azimuth and elevation. Note positive +a obeys the right hand rule about +B$_2$, while +e obeys the right hand rule about +B$_1$.

The additional degree of freedom, roll angle about the sun vector, is defined relative to the sun-frame in Section 3.1.

The table below summarizes the geometric properties used in disturbance torque models in this thesis. They are distilled from the preliminary design proposed by Aero/Astro Inc.

S/C envelope width:	18"	(envelope cylindrical)
S/C envelope length:	35"	
Paddle hinge points:	8" from +B$_3$ face	
Paddle length:	27.1"	
Paddle inner radius:	9"	
Paddle arc-length:	11"	
On orbit c.m. location:	18.4" from +B$_3$ face	

Mass properties have been estimated for the preliminary design. The inertias, of course, change as a function of solar panel position and amount of fuel in the tank. For the purposes of this thesis, the on-orbit configuration- with paddles fully deployed and no propellant- is the only one considered.The inertia matrix expressed in $\mathcal{F}_B$ in units of Kg•m^2 is:

```
ib =
     4.8600    -0.0100     0.0100
    -0.0100     5.4100    -0.0600
     0.0100    -0.0600     4.0800
```

Examination of the eigenstructure reveals the principal inertias to be:

$$ip =$$

4.8599	0	0
0	5.4129	0
0	0	4.0772

while the principal vectors, with magnitudes normalized to 1 and components expressed in $\mathcal{F}_B$, are:

0.9997	-0.0189	-0.0122
0.0194	0.9988	0.0449
0.0113	-0.0451	0.9989

The body fixed principal frame $\mathcal{F}_P$ is defined such that the three principal vectors form an orthogonal basis, in which case the matrix just given represents the direction cosine matrix C_{BP}.

It is of course emphasized that the mass properties are from a preliminary design and will no doubt change as the the design is finalized. Nevertheless, the following assumptions are made:

- Reasonably close correspondence between $\mathcal{F}_P$ and $\mathcal{F}_B$ will be maintained in the final design. The approximation $\mathcal{F}_P = \mathcal{F}_B$ is made in control design in Chapter 5. Specification of the offset between the two frames (so as to better define "reasonably close correspondence") is boiled down to the more descriptive parameter of an Euler rotation angle about an Euler axis in the B_1-B_3 plane, as will be discussed in Chapter 4.

- Decisions on the amount of bias angular momentum to store based on these inertia estimates will still be roughly valid for the final design.The decision is based partly on the tradeoff between inertial stiffness vs. nutation frequency, which of course depends on the body inertias.

2.2 ACS Hardware & Models

Figure 2.4 shows the proposed attitude control system hardware suite. Not shown are the computation hardware- which will simply be integrated with other housekeeping tasks- and an Earth horizon sensor, which would be necessary only if an orbit raising maneuver is to be performed. Such a maneuver, if using compressed Freon as propellant, would require body spin to provide centrifugal pull of the fuel to the bottom of the tank. Spin about B_2 (not B_1, because then the wheel could not be used to kill the spin upon arrival at the desired orbit), with B_2 aligned to the orbit normal, would then have to be actively

maintained. Roll and pitch sensing could then be performed in a very conventional manner with horizon sensing.
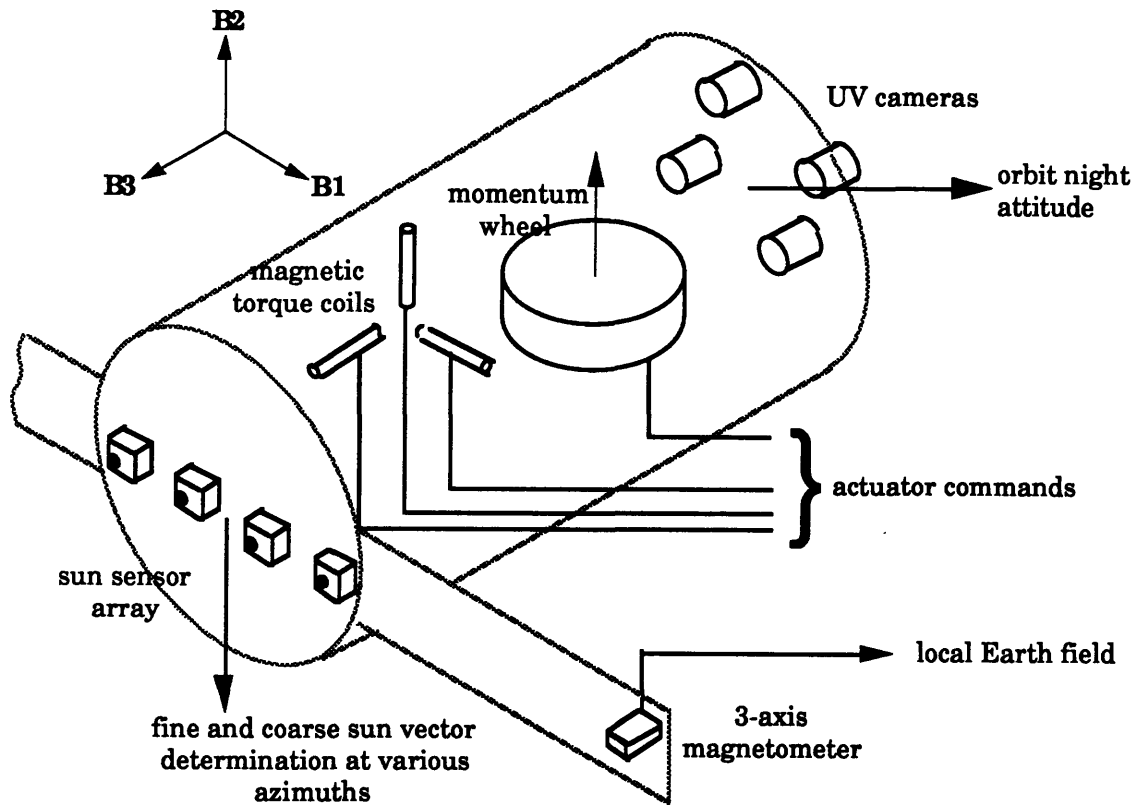
Figure 2.4: schematic representation of the HETE ACS hardware for on-orbit control. See text for clarification of the sun-sensor arrangement. Drawing courtesy of AeroAstro Inc.

2.2.1 Momentum Wheel

The momentum wheel is oriented with its spin axis along B_2. This choice has been made primarily so that spacecraft rotation about B_2, which is unstiffened, is observable by sun sensors facing toward B_3, the nominal desired sun vector direction. Thus, an active loop from sun-sensor azimuth measurement to wheel torque can always be closed to regulate attitude and rate about B_2. The following definitions are made because of this choice and will be used throughout this thesis:

- "spin-plane" B_1-B_3 plane

- "spin-axis" B_2 axis

The current candidate for the momentum wheel is the Scanwheel by Ithaco Space Systems, Ithaca, NY. It is of the right size, weight, and momentum capacity, with reasonably low power consumption characteristic of brushless DC motors. The following table of characteristics, quoted from product literature, is used in the design plant model for control design:

wheel inertia (about spin axis):	6.3×10^{-3} Kg•m^2
speed range:	± 6000 RPM
maximum torque:	20×10^{-3} N•m
weight (no electronics):	2.3 Kg (5 lbm)

Significantly, resolution of the commandable torques is not given. Details of the motor control loop design is not discussed in this thesis, but calculation of the effects of command torque quantization on body rate control in a closed loop setting is given in Chapter 5, after the control design. Given the stringency of the body rate specifications, it is fair to assume that the wheel control loop will consist of a high-resolution encoder driving a phased-locked loop. It is also assumed that the bandwidth of the wheel control loop is at least a decade wider than that of the attitude control loop, which should not be difficult since the attitude control loops have sample rates of 1 Hz during orbit day and 0.25 Hz during orbit night, and the design nominal wheel speed is 3000 RPM. More on in Chapter 4.

The wheel model used in control performance simulation takes into account several imperfections. As will be seen, the most significant is the torque command resolution. For the remainder of this thesis, the quantization step size is assumed to be 5 μN•m, which for the Ithaco wheel model corresponds to an equivalent acceleration command resolution of 8×10^{-4} rad/s. This assumed quantization step is calculated in Section 5.2.5 to be permit meeting drift rate specifications with a reasonable margin, and therefore represents a goal to be achieved in the wheel-control loop design. We also assume that the torque ripple associated with the motor is within this quantization step size.

There is no intention to go long enough between wheel desaturation to permit the wheel speed to drop past 0, so no Coulomb friction / wheel speed deadband model need be considered. The actual wheel inertia is assumed to be offset from the design plant model value by $\pm 5\%$, the sign being random. Finally, the actual wheel axis is assumed to be offset from B_2 somewhere in a conservatively wide cone of $5°$ width.

2.2.2 Magnetic Torquers

Since the presently considered design calls for active control using linear actuators, the assumption is made that a linear relationship exists between the current applied to the coil and the magnetic moment obtained. This dictates the core material used, most likely "air" [Wert78]. Accurate control of the current supply is therefore also necessary, implying the need for a separate closed loop to compensate for coil resistance and supply voltage changes. In the context of a digital control system, this is best accomplished using a pulse-width modulation scheme to allow operation of the drive transistors in their efficient saturated region as well as simple digital to analog interface. The design moment range is ± 4 A$\cdot$m^2 per coil. It will be shown that performance is sufficient with coils of this size. The following imperfections are accounted for in the torquer model used in simulation:

- resolution: ± 4 A$\cdot$m^2 / 8 bits
- $\pm 10\%$ offset per coil factor[1], sign random

As implied by Figure 2.4, the coils are arranged along the three $\mathcal{F}_B$ axis. Possible misalignments from the design arrangement are an orthogonal rotation and non-orthogonal distortions. For simplicity, only an orthogonal misalignment is modelled explicitly. The misalignment model is a rotation by a 5° Euler angle about a randomly selected Euler axis.

2.2.3 Sun Sensor Array

Conflicting needs on sun sensing drives the proposal for an array of sun-sensors. As will be shown in Chapter 5, body rate control specifications during orbit day requires sufficient angular resolution on the measurement, which limits the field of view per sensor. However, the slew maneuver about the spin-axis described in Chapter 1 requires acquisition of the sun vector from a wide range of possible azimuths on the spin plane. This requirement is moderated by the fact that the fields of view of the payload itself are

---

[1] the magnetic moment (dimensions A$\cdot$m^2) is the product of the input current to the coil and the coil factor, defining the latter quantity. This models imperfection in the construction of the coil.

sufficiently wide to make unnecessary the ability to slew to any arbitrary azimuth. Rather, an array of desired azimuth "notches" can be pre-selected. This leads to the compromise configuration of an array of fine sun sensors, placed on the +B_3 face of the spacecraft with their boresights along these azimuth notches. The appropriate sensor can then be turned on when necessary. The actual number of azimuth notches desired has not yet been determined, though this is arbitrary from the control design point of view. Sections 4.2.2 and 5.4 discuss the kinematics and array configuration in greater detail.

It is noted that, for the purpose of initial attitude acquisition and recovery from an unexpected ACS shutdown, the sun-sensor configuration described above need only be augmented with very coarse sun-presence sensors (e.g. a single solar cell) on the other body faces.

Turning attention to the individual sun sensor, the current proposal calls for the use of an analog quad-cell variety, where 4 solar cells are arranged in a 2 x 2 matrix and are illuminated by light passing through a square cutout in the covering above the cells. The cutout is such that its geometric center is directly above the center of the solar cell matrix, making boresight of the sensor the vector between the two centers. Alignment of the sun vector with the boresight therefore produces equal current output on all cells, defining the null condition.

Since the physical output is the current difference between adjacent cells, it is seen that geometrically the measured quantities are the cosines of the projected angles of the sun vector on two planes orthogonal to the solar cell plane and to each other. These quantities are made precise by the following definitions:

$\mathcal{F}_Q$:   quad-cell frame, with Q_3 = boresight axis

$\underline{S}$:   sun-vector expressed in $\mathcal{F}_Q$, such that:
$$\underline{S} = [s_1 \ s_2 \ s_3]^T, \quad ||\underline{S}|| = 1$$

θ_1, θ_2: angles between $\underline{S}$ projections and Q_3, as shown in Figure 2.5 below

ζ_1, ζ_2: sun-sensor current difference outputs: $\zeta_1 = \cos(\theta_1)$, $\zeta_2 = \cos(\theta_2)$
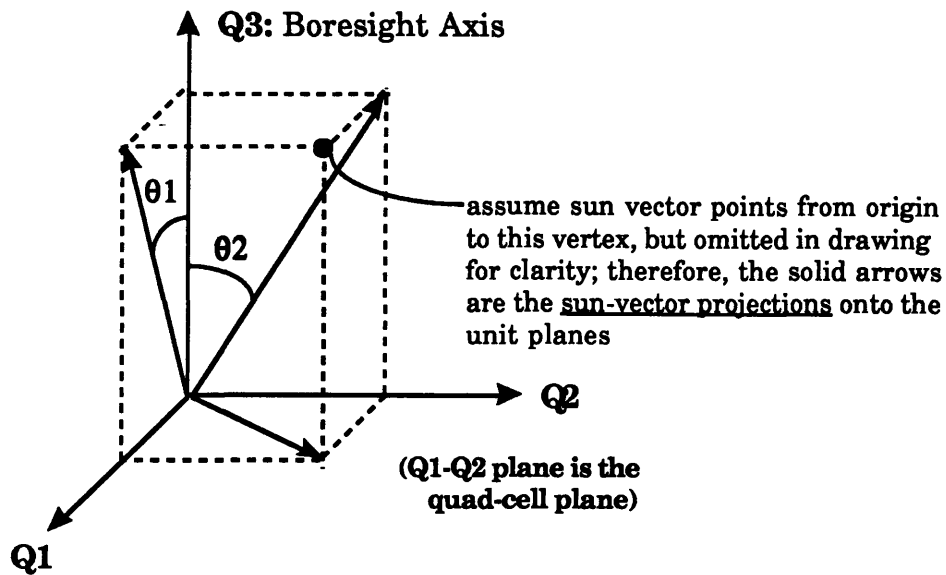
Figure 2.5: definition of the sun sensor frame $\mathcal{F}_Q$ and the angles θ_1 and θ_2, which are made by the projections of the sun-vector onto the Q_1-Q_3 and Q_2-Q_3 planes with the boresight axis Q_3 respectively.

The sensor field of view, then, defines the range of θ_1 and θ_2. It will be shown in Section 4.2.2 that for calculation of spacecraft kinematic variables, a more convenient sun-sensor "output" set is the vector [s_1 s_2] itself. Since the processing of the raw sensor output is digital, this is not difficult to implement. The components of $\underline{S}$ can be solved for given θ_1, θ_2, and the constraint that $||\underline{S}|| = 1$.

Modelling the resolution of the measurements [s_1 s_2] is necessary, as controller performance depends greatly on this. The actual digitization step occurs on sampling a voltage signal corresponding to the current difference between adjacent cells, and the following modelling assumption, which should obviously be implemented anyway, is made:

- For a given field of view, the scaling factors are adjusted such that the full range of θ produces the corresponding full voltage range on the ADC.

We use the following design baseline for the fine sun-sensors:

- 10 bit ADC   • ±5° field of view

Therefore, the resolution on θ is (10/1024) degrees. Now, consideration of Figure 2.5 reveals that:

$$s_1 = s_3 \tan(\theta_1) \qquad s_2 = s_3 \tan(\theta_2) \qquad (2.1)$$

But it is clear that if θ_1 and θ_2 are small, s_3 is approximately 1, and $\tan(\theta)$ is approximately θ. Therefore, we will use the approximation that the resolution on s_1 and s_2 are (10/1024).

Misalignment and biases due to various factors of the individual sun sensors are not modelled. This is justified analytically in Sections 4.2.2 and 5.5, but the simple answer is that the only effect of the misalignments and biases is to limit the tracking performance of the controller. This is obvious and need not be accounted for explicitly in simulations.

Noise on the sun-sensor signals is also not modelled. This is because the pre-filter between the sensor and the control loop is at the very low bandwidth of $\omega_{pf} = 0.75$ rad/s. Section 5.3.2 discussed the choice of pre-filter bandwidth. It is reasonable to expect there to be no electronic noise power of consequence this low in the spectrum.

2.2.4 Magnetometers

A 3-axis magnetometer set is necessary to implement torque commands from the controller, as the needed magnetic moment from the coils depends on the local field. The design baseline model is a Schonstedt model with ± 60 μT dynamic range, with digital sampling from a 10 bit ADC. Linearity and bias of the instrument itself is quoted to be below this (120/1024 = 0.117 μT/bit), so 0.117 μT can serve as a the resolution of the instrument.

The instrument on board the spacecraft is subject to biases from several sources. The sensitive axis of the instruments may be misaligned with the case, but this is small compared to the misalignment of the case with $\mathcal{F}_B$, since the instrument is mounted at the end of a solar paddle. Interference from the spacecraft itself takes place in three forms: a residual body dipole, current loops, and the magnetic moment generated by the torque coils. Chapter 4 defines the angle θ as the angle between B_3 and the projection of the local field vector on the spin plane. It is seen in Chapter 5 that in addition to translating a torque request to a set of coil moment commands, the magnetometer signal is also used to compute θ, on which the control laws are scheduled. Robustness to offsets in θ measurement is investigated in Chapter 5, and calculating the θ error bound from an assumed magnetometer bias is discussed in Section 4.2.3. For the moment, we simply state the modelled bias and misalignment errors in simulations: ± 3 μT in each axis, sign random; 5° orthogonal rotation of $\mathcal{F}_Q$ with respect to $\mathcal{F}_B$ about a random Euler axis.

Since HETE's science mission requires that the spacecraft ephemeris be known always (an on-board GPS receiver is envisioned), and since during orbit night the CCD cameras provide high-quality attitude information, the situation is ideal for calibration of the magnetometers in orbit by dynamically comparing the measurements with a high-order field model. This can be done either on-board or on the ground. Results as good as 0.25 μT RMS bias have been obtained in [Lern79], so an assumed 3 μT bias should be sufficiently conservative.

The magnetometer measurement is requested once per sample period by the attitude controller, which operates at either 0.25 Hz or 1 Hz. However, since the intention is to measure the Earth's field, and since it will be seen in Chapter 5 that the magnetometer is not in the closed loop, the bandwidth of the magnetometer pre-filter is mostly dictated by the spectral characteristics of the field itself[1]. It is demonstrated in Sections 3.2.3 through 3.2.5 that pre-filter bandwidths in even the decade below 1 Hz would not block significant field information. Therefore, as in the case with the sun-sensor, it is safe to assume that electronic noise from the instrument need not be modelled explicitly.

2.2.5 UV Cameras

Star identification by the UV camera is required to obtain the accurate celestial fix of burst events required as part of the science mission. Therefore, the instrument makes for a good attitude instrument during orbit night, when its view is not occulted by the Earth. The angular resolution of a particular point source in the sky is on the order of an arcsecond [Pers91], but since the calculation of spacecraft attitude is made from matching the observed positions of many stars to a catalog, the attitude resolution is significantly better than this [Pers91]. For simulation purposes we will assume perfect[2] attitude information from the instrument as a first estimate in this thesis, while developing the tools to evaluate the performance degradation from limited resolution in Chapter 5.

---

[1] this may seem confusing. The control loop is designed with control inputs having dimensions of torque, and the fact that the torque is actually generated by commanded coil currents calculated using magnetometer measurements is not known to the loop. Therefore, setting the magnetometer filter bandwidth too low may cause instability by introducing large actuator errors, and not by the large phase lag associated with a filter which is actually in the loop. This clarifies the assertion that the bandwidth of the filter is mostly dictated "by the spectral characteristics of the field itself."

[2] to within numerical quantization of the DSP chips supporting data reduction from the instrument. As an example, the Motorola 56000 series uses 24 bit arithmetic, which is certainly enough resolution to make this factor insignificant.

The current design of the instrument calls for a frame rate of 0.25 Hz., with a delay of several hundred milliseconds before an attitude fix is calculated. Increasing the frame rate from this is difficult because of a non-linear rise in power consumption to increased frame rate [pers91]. The orbit night controller, therefore, is designed in Chapter 5 with a 0.25 Hz. sample rate, and simulations use the conservative estimate of a 1 second delay between the sample time and the output of a new actuator command set.

2.2.6 Some Clarifications

Many of the sensor/actuator imperfection models given above have random directional components to them. The intent is to sidestep difficult issues- for example, specifying the worst misalignment direction of the torquers- by performing a large number of simulations with the direction chosen randomly each time. Though inelegant, this approach has permitted the conclusion that, within the magnitude limits given, all other imperfections modelled are insignificant compared to the effects of wheel torque quantization and sun-sensor angular resolution. This is discussed in Chapter 5 as well.

III. Orbital Environment Models

In estimating actuator size and controller performance, an adequate model of the disturbance torque environment is necessary. The qualifier "adequate" is stressed, since a great deal of effort can be wasted on the task of modelling disturbances to better than is necessary. That the attitude control scheme employs no expendables eliminates the need for extremely detailed disturbance analysis. Magnetic torquing plays a large role in the attitude control scheme, and since both the magnitude and direction of the magnetic control torques allowed are constrained by the local Earth field vector, a reasonable model of the Earth's field and, more importantly, visualization of the geometry of the field in the spacecraft's reference frame are desirable. All of these considerations necessitate the introduction of well defined reference frames to facilitate discussion of quantities of interest. These issues are addressed in this chapter.

3.1 Geometric and Astrodynamic Considerations

This section defines and relates all the reference frames necessary to consider specifications and performance, analyze disturbances, simulate the Earth field, and perform other software functions necessary to support control design validation simulation. Brief consideration is also given to issues such as lighting conditions and orbit perturbations. Appendix A-3a contains MATLAB code implementing the transformations. The notation convention established in Chapter 2 is used.

3.1.1 Sun Frame and Earth Equatorial Frame

The sun frame, designated $\mathcal{F}_S$, is an inertial frame[1] with S_1 pointing towards the vernal equinox, S_3 pointing towards the solar north pole (as defined by the right hand rule applied to the direction of earth travel in the ecliptic), and S_2 such that a dextral system S_1 X $S_2 = S_3$ is formed. The classic definition of the vernal equinox is the direction <u>from the Earth to the Sun</u> on the first day of spring ([Wert78]). Note therefore on the first day of <u>fall</u> S_2 coincides with the direction of earth travel in its orbit.

---

[1]Unless otherwise noted, the term "inertial" will be used to denote that non-Newtonian motions are very slow with respect to the time scale of the spacecraft's attitude dynamics.

The equatorial frame, designated $\mathcal{F}_E$, is also inertial, with $\mathbf{E_1} = \mathbf{S_1}$, $\mathbf{E_3}$ pointing towards the Earth geographic North Pole, and $\mathbf{E_2}$ such that a dextral system is formed. Clearly, $\mathcal{F}_E$ and $\mathcal{F}_S$ are simply related by the obliquity of the ecliptic (23.4 degrees at Epoch 2000). Figure 3.1 and equation (3.2) relate the two frames, the ecliptic obliquity denoted here by θ:

$$C_{SE} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & \sin(\theta) \\ 0 & -\sin(\theta) & \cos(\theta) \end{bmatrix} \tag{3.2}$$
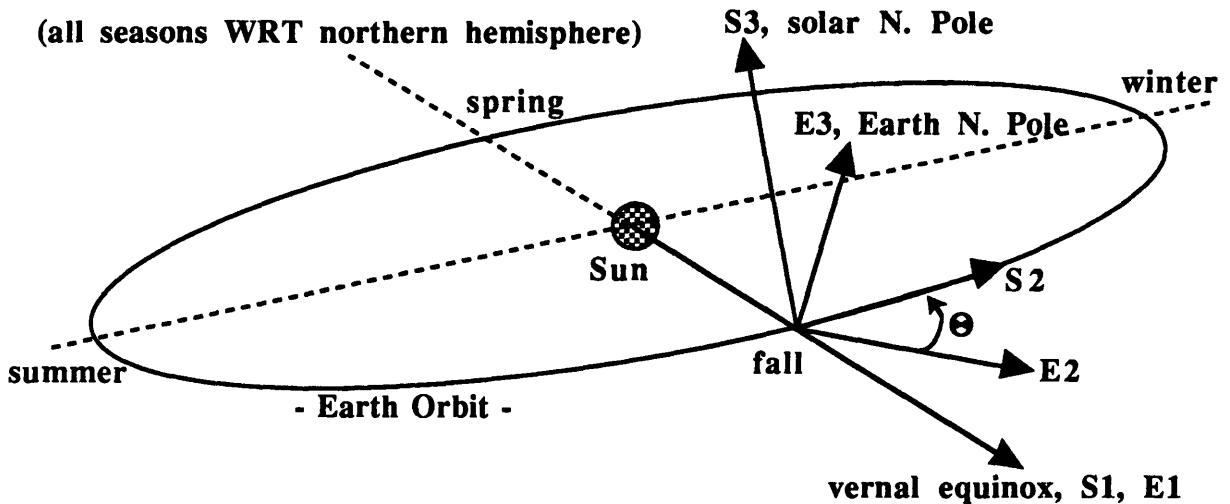
Figure 3.1: Geometry of the sun and earth frame definitions. Note that although $\mathbf{E_2}$ is in the Earth equatorial plane, it does not rotate with the earth, but rather remains perpendicular to the direction of the vernal equinox for all times of the year.

3.1.2 Greenwich Frame

The Earth Greenwich Geocentric frame, designated $\mathcal{F}_G$, is defined so as to facilitate the determination of local longitude/latitude. This tracking of essentially local time information is necessary because the Earth's magnetic field rotates with the Earth, and thus all field models take as parameters local longitude and latitude.

G_1 is defined as the vector from the center of the Earth to the intersection of the prime (Greenwich) meridian with the Earth equatorial plane. G_3 coincides with E_3, pointing along the Earth North Pole. G_2 is again defined to complete a dextral system. Note therefore $\mathcal{F}_G$ is related to $\mathcal{F}_E$ by a rotation about the North Pole, as shown in figure 3.2 and equation (3.3). We note in passing that the specification of $\mathcal{F}_G$ with respect to $\mathcal{F}_E$, along with a sun vector (time of year information), determines the local time.
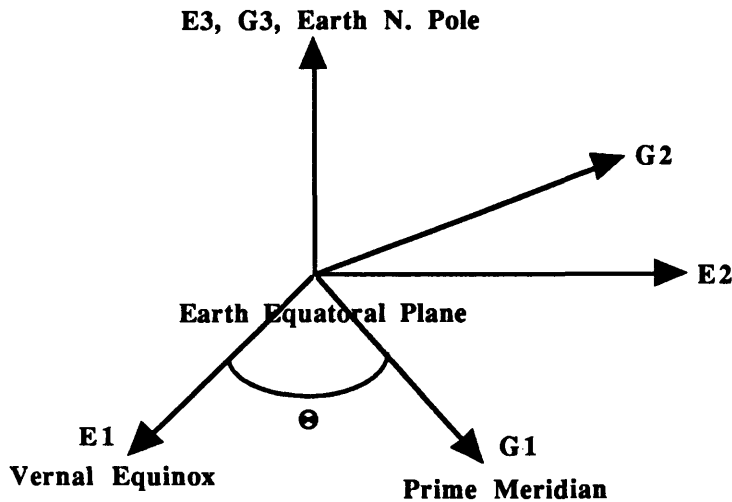
Figure 3.2: Relation between Equatorial frame and Greenwich frame, defining the latter.

$$C_{GE} = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad (3.3)$$

3.1.3 Orbital Plane Frame

An orbital plane about the Earth is specified with two parameters, the inclination and the right ascension. The inclination is the angle between the orbit plane and the equatorial plane, defined positive from the east half of the equator when seen head-on (flat). Denoting the inclination by "i", we see therefore i<90° for a prograde orbit, i>90° for retrograde. The right ascension is the angle, measure eastward from the vernal equinox along the equatorial plane, of the ascending intersection of the orbit with the equatorial

plane. The orbit intersects the equatorial plane at two points, the ascending intersection being distinguished from its counterpart by being the intersection where the satellite goes from south to north. The line joining the ascending and descending nodes is called the line of nodes.

The orbital plane is designated $\mathcal{F}_O$. O_1 is the vector pointing from the center of the Earth to the ascending node, O_3 is the vector normal to the orbit plane, and O_2 completes the dextral system. The direction of O_3 is such that the right hand rule applied to the spacecraft orbital velocity is satisfied. $\mathcal{F}_O$ is related to $\mathcal{F}_E$ by two simple rotations about the North Pole and about O_1, as shown below:
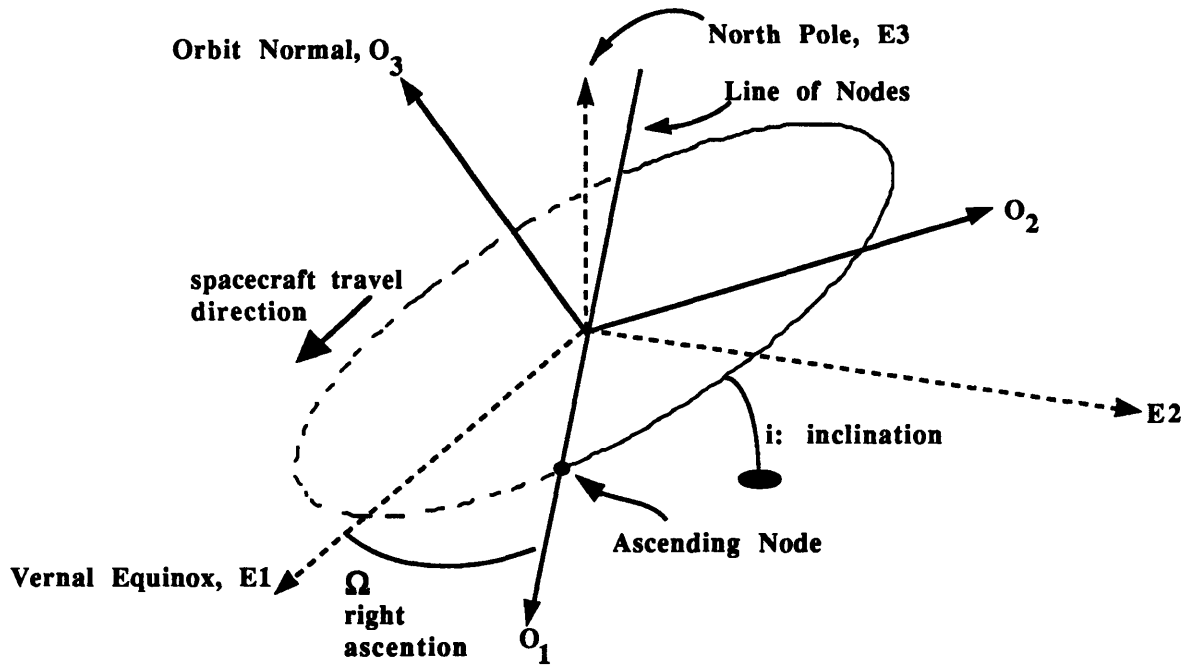
Figure 3.3: relates Equatorial frame to Orbital plane frame, defining the latter

$$C_{OE} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(i) & \sin(i) \\ 0 & -\sin(i) & \cos(i) \end{bmatrix} \begin{bmatrix} \cos(\Omega) & \sin(\Omega) & 0 \\ -\sin(\Omega) & \cos(\Omega) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.4}$$

3.1.4 Orbit Local Vertical- Local Horizontal (LVLH) frame

This frame locates the spacecraft in its orbit, as well as providing the orientations of the local vertical (towards Earth below) and local horizontal, which for circular orbits corresponds to the orbital velocity vector direction. Earth-observation spacecraft dynamics and control analysis is usually performed in this frame. Location of the spacecraft is determined by the true anomaly η, defined as the angle from the right ascension (O_1) to the local vertical, positive in the direction of spacecraft travel. The LVLH frame, which we also designate $\mathcal{F}_L$, is defined with L_1 = local horizontal, L_2 = orbit normal, L_3 = local vertical (nadir). Figure 3.4 and (3.5) below relate $\mathcal{F}_L$ to $\mathcal{F}_O$. Note that, to make $\mathcal{F}_L$ dextral, L_2 points opposite to O_3.
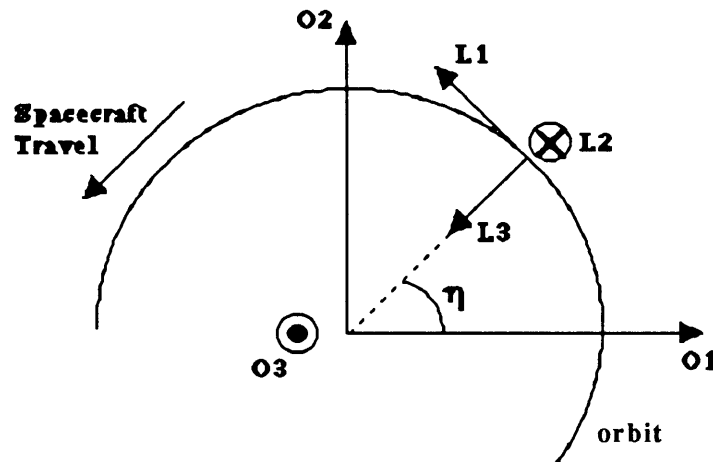
Figure 3.4: relates the Orbital plane frame to the LVLH frame

$$C_{LO} = \begin{bmatrix} -\sin(\eta) & \cos(\eta) & 0 \\ 0 & 0 & -1 \\ -\cos(\eta) & -\sin(\eta) & 0 \end{bmatrix} \qquad (3.5)$$

3.1.5 Local North-East-Down frame

The geophysics community, which provides models of the Earth magnetic field as well as regularly updated field measurements, expresses the local Earth field in a variation of the LVLH frame called the local North-East-Down frame, which we will designate $\mathcal{F}_N$. $\mathcal{F}_N$ is easiest to visualize on the surface of the earth. N_1, local North, is the direction to travel such that one will arrive at the North Pole without changing longitude.

Similarly, traveling along local East N_2 results in no latitude change. N_3 is the local nadir. Historically, the measurement of the local field direction consists of specifying the angle of a needle from local geographic North (the Declination) and the angle towards the nadir (the Inclination or Dip). Considering figure 3.4, we see that for a prograde spacecraft, the relationship between $\mathcal{F}_L$ and $\mathcal{F}_N$ is:

$$C_{LN} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.6}$$

3.1.6 Mission Mode Nominal Body Frame Orientation

HETE is sun pointing during its mission mode. Thus a convenient transformation for the nominal attitude of the spacecraft's body fixed frame $\mathcal{F}_B$ (as defined in chapter 2) consists of specifying a sun-vector, which contains time of year information, and a roll angle about the sun-vector. We designate this transformation C_{BS}, connecting $\mathcal{F}_B$ and the sun frame $\mathcal{F}_S$. Note that the principal use of such a transformation is in evaluating the nominal orbital conditions encountered by the spacecraft; it is therefore assumed that a control loop actively maintains the attitude such that the transformation C_{BS} specifies the spacecraft attitude adequately for this purpose. Note also that C_{BS} is sufficient to relate $\mathcal{F}_B$ to all other frames defined. For example, to determine the orbital velocity vector in the body frame for, say, a drag torque estimate, the transformation $C_{BL} = C_{BS}C_{SE}C_{EO}C_{OL}$ would appropriately simplify the calculation.

Figure 3.5 defines the angles ϕ and ψ used in this discussion. Note that the angle of sun vector as defined in relation to the seasons. $\phi = 0$ corresponds to the earth being at spring equinox, 90° --> summer solstice, 180° --> autumn equinox, and 270° --> winter solstice (seasons in the northern hemisphere). ψ is defined such that at $\psi = 0$, the body y axis B_2 points toward the solar north pole. Positively increasing ψ is defined so as to obey the right hand rule about the sun vector, as seen in figure 3.5.

(F$_T$ = temporary frame, T3 = S3 = solar north pole)
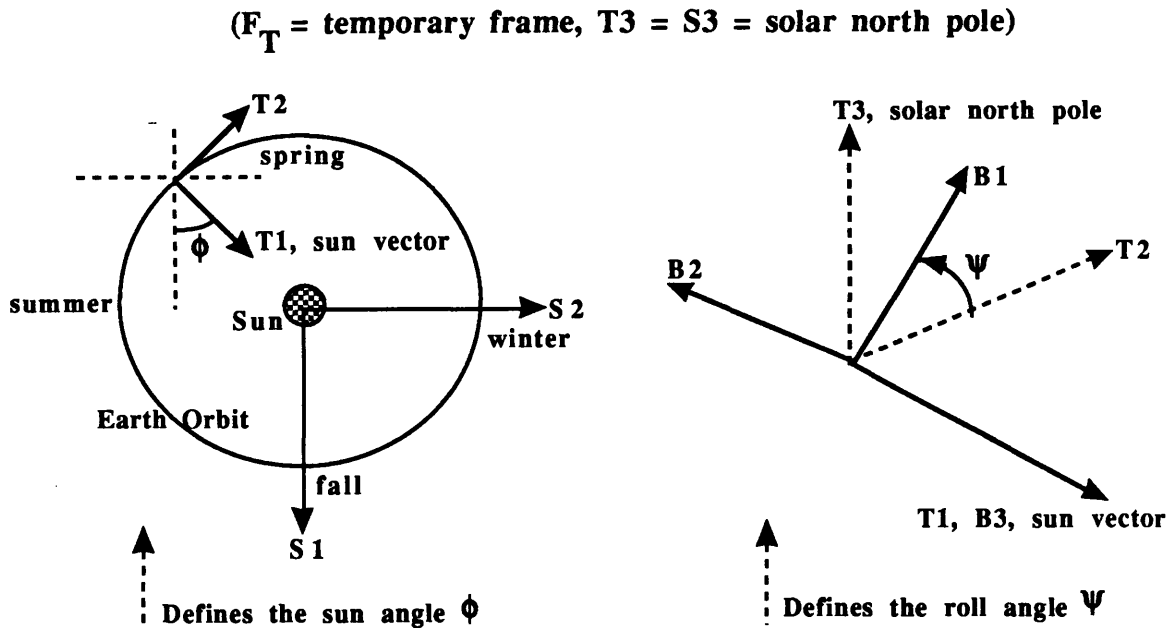
Figure 3.5: shows the relationship between the sun frame and the nominal body frame. Note the definition of an intermediate frame $\mathcal{F}_T$ to clarify the transformation.

$$C_{TS} = \begin{bmatrix} \cos(\phi) & \sin(\phi) & 0 \\ -\sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad C_{BT} = \begin{bmatrix} 0 & \cos(\psi) & \sin(\psi) \\ 0 & -\sin(\psi) & \cos(\psi) \\ 1 & 0 & 0 \end{bmatrix} \qquad C_{BS} = C_{BT}C_{TS} \qquad (3.7)$$

3.1.7 Perturbations from Keplerian Orbit

All orbital elements are subject to perturbations from the Keplerian 2-body ideal, but all occur at a time scale much slower than need be considered for analysis of attitude dynamics. Furthermore, since the planned mission orbit is circular, perturbations to the argument of the perigee are irrelevant. However, node regression greatly affects the distribution of the Earth's magnetic field into a sun-oriented spacecraft frame, and is therefore considered here. Node regression of satellite orbits is largely due to non-spherical terrestrial mass distribution, with interactions with the Moon and the Sun contributing very little to the same effect. Earth oblateness is modelled by expanding a geopotential function in a series of Legendre polynomials, with the first term corresponding to a sphere, the second to an ellipsoid, and on up. The coefficient of the

second term outsizes that of the third by more than 2 orders of magnitude, so calculations employing only a second order expansion are sufficient. [Wert91] gives such an approximate formula for $d\Omega/dt$ which is plotted below. Note that at 28° inclination, $d\Omega/dt$ = **-6.7 deg/day**.
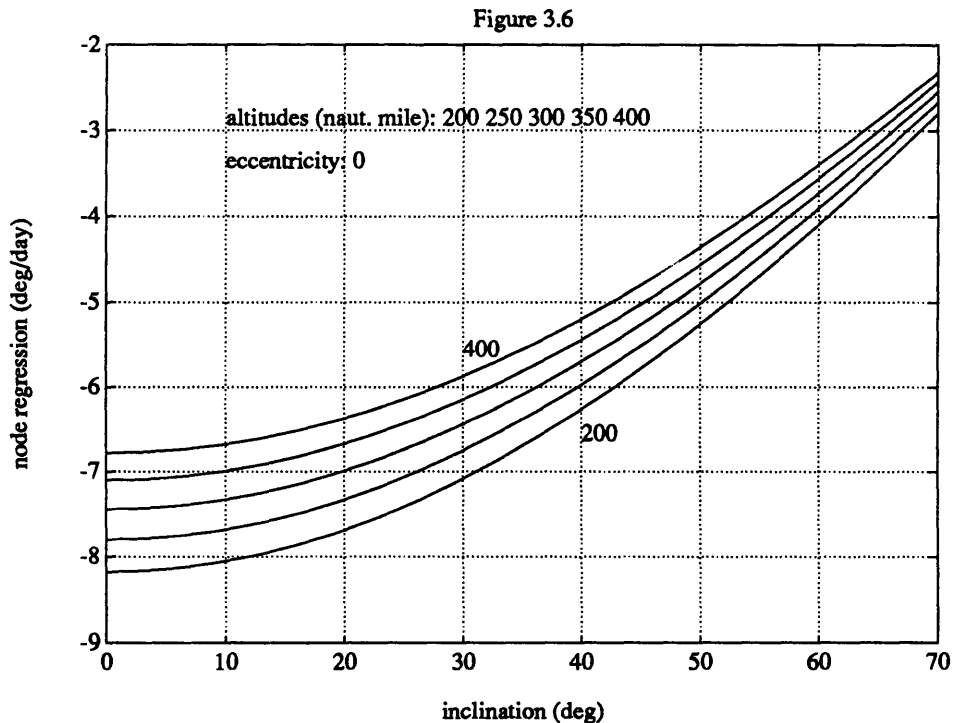
Figure 3.6

Figure 3.6: Node regression rate as function of inclination for circular orbits at altitudes indicated.

3.1.8 Ground Track Calculation

In exercising the standard Earth field model it is necessary to calculate the longitude and co-latitude[1] at some given point in the orbit as parameters. This amounts to calculating the ground track for the spacecraft. With the reference frames formally defined above, this task is made more straightforward. The following is an algorithm expressed in terms of the defined reference frames, assuming the current orbital elements and time of day are given in the C_{EO}, C_{OL}, and C_{GE} transformations.

---

[1]The co-latitude is measured from the North Pole, eg. 0 to 90° correspond to the northern hemisphere, 90° = equator.

- Co-latitude = angle between G_3 (North Pole) and $-L_3$ (local zenith on orbit):

$$\phi_c = \cos^{-1}\{[0 \quad 0 \quad 1]C_{GL}\begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix}\}$$

$$C_{GL} = C_{GE}C_{EO}C_{OL}$$

- Longitude = angle between G_1 (Prime Meridian) and <u>projection</u> of local zenith to equatorial plane:

- $\underline{Z}_E \equiv C_{EO}C_{OL}\begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} = \begin{bmatrix} z_{E_1} \\ z_{E_2} \\ z_{E_3} \end{bmatrix}$ local zenith in $\mathcal{F}_E$

- $\underline{Z}_{Ep} \equiv \dfrac{[z_{E_1} \quad z_{E_2} \quad 0]^T}{\left\| [z_{E_1} \quad z_{E_2} \quad 0] \right\|}$ unit projection to equatorial plane; Z_{Ep} is 3x1

- $\phi_L = s(\cos^{-1}\{[1 \quad 0 \quad 0]\,C_{GE}\,Z_{Ep}\})$ calculated longitude, s is ± 1

where: $s = \text{sgn}(v_3)$, $\underline{v} \equiv \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \times (C_{GE}\,Z_{E_p}) = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$

The third component of the vector $\underline{v}$ as defined, which is in $\mathcal{F}_G$, is positive if Z_{Ep} lies east of the prime meridian, which is $[1 \quad 0 \quad 0]^T$ in $\mathcal{F}_G$.

Figure 3.7 shows an example ground plot. The code to implement the algorithm above and generate figure 3.7 is included in Appendix A-3b.
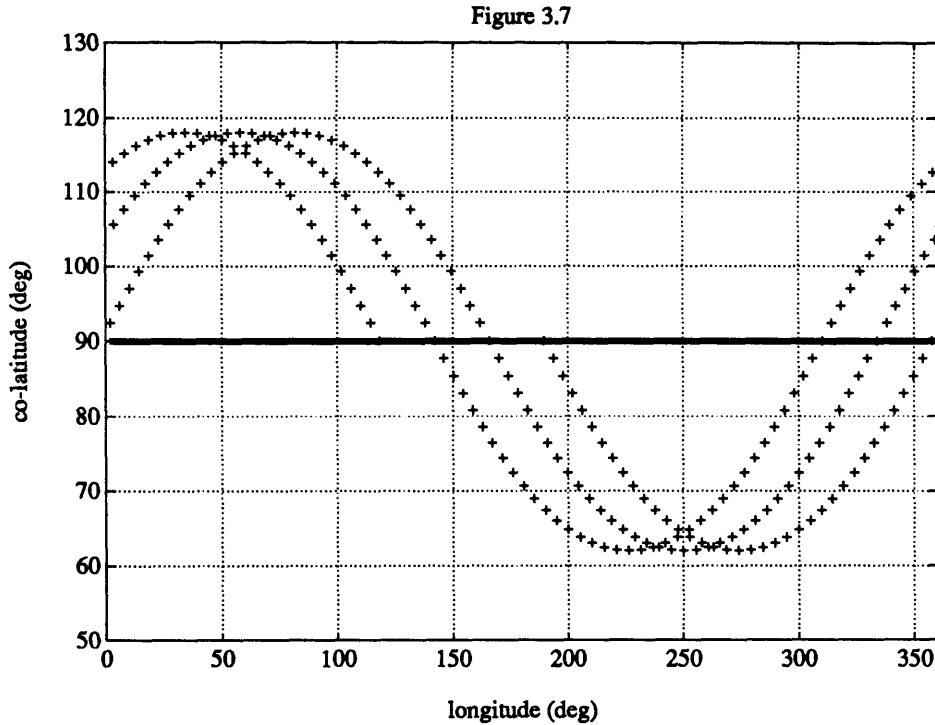
Figure 3.7

Figure 3.7: An example ground track calculation for 3 orbits at 28° inclination. The starting local time and orbit right ascension are arbitrary.

3.1.9 Lighting Conditions

An analysis of solar eclipse by the Earth is readily performed within the framework of the reference frames defined. [Wert91] presents a geometric formula for the eclipse fraction in a circular orbit as a function of Earth angular radius and angle of the sun above the orbital plane, implemented in the code in Appendix A-3c to generate figure 3.8.

The field of view of the CCD camera, as well as the percentage of the field of view which must be dark (ie. not illuminated by Earth view) in order for it to function as an attitude instrument, has not yet been determined. Therefore, no analysis of optimum day/night mode switching is given in this thesis, and simulations assume an arbitrary switch time with appropriately set initial conditions.
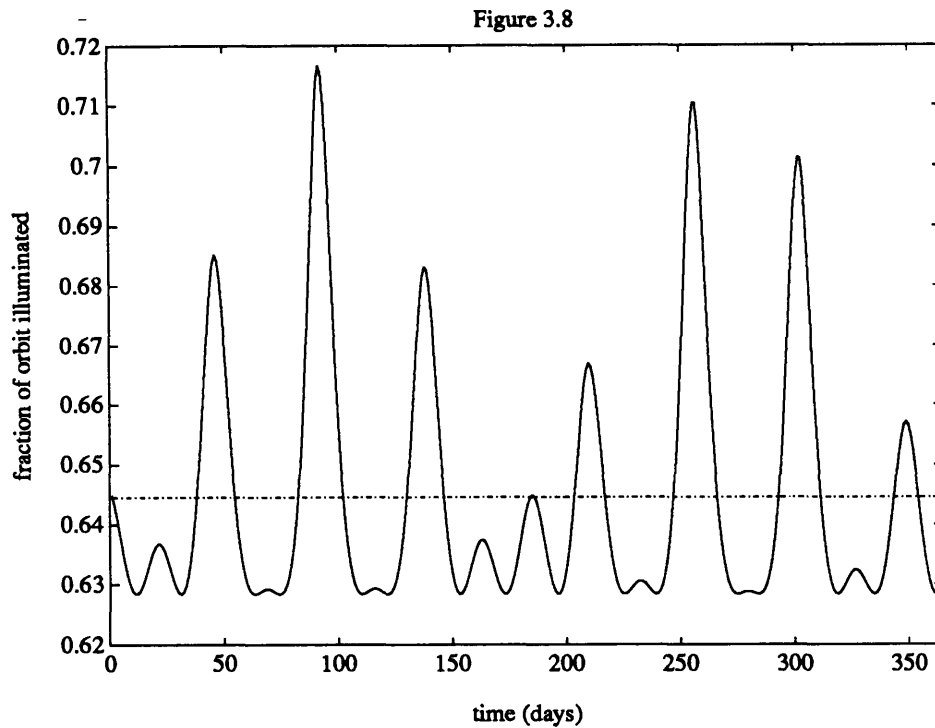
Figure 3.8

Figure 3.8: Fraction of orbit spent with sun visible for a 300 nautical mile circular orbit at 28° inclination, node regression = -6.7° per day. Mean over the course of a year is 64.4% (dashed line). Initial orbit right ascension is arbitrary.

3.2 Terrestrial Magnetic Field in LEO

Equation 3.8 is the physical basis of magnetic attitude control:

$$\tau = m \times \beta \qquad\qquad (3.8)$$

where: τ = control torque vector
          m = dipole moment vector generated by coils
          β = local field vector[1], units of field intensity (flux density)

---

[1]note β is used to avoid confusion with B, which refers to the body fixed reference frame

That the magnitude of the control torque available is limited by the available field is clear. More important, however, is the directionality of τ implied by the cross product. τ is restricted to be in the plane normal to β, significantly complicating control law design. Furthermore, to calculate **m** to generate a commanded τ requires measurement of the current β from magnetometers. Therefore, the geometry and time variations of the terrestrial field must be investigated.

Several systems of units are commonly used in the literature to measure the quantities in (3.8) and related quantities. The units in the first columns of Table 3.1 will be used in this thesis; their relation to other common units are give in adjacent columns:

Quantity:	Unit Used Here:	Other Common Units:
Magnetic Flux:	Weber (Wb)	1 maxwell = 10^{-8} Wb
Field Intensity (Flux Density):	1 Tesla = 1 Wb/m^2 (m = meter)	1 Gauss (G) = 10^{-4} Tesla 1 gamma (g) = 10^{-9} Tesla
Dipole Moment:	A·m^2 (A = ampere)	1 Wb·m = $(10^7/4p)$ A·m^2 1 pole·cm = 10^{-3} A·m^2
Torque:	N·m	

Table 3.1: Magnetic Units

3.2.1 Geometry and Terminology

As mentioned, frame $\mathcal{F}_N$ defined in Section 3.1.5 is the standard reference frame used by the geophysics community, which, although intended for use on the surface of the Earth originally, generalizes to use in orbit easily. Figure 3.9 defines the horizontal component **H**, the vertical component **V**, the declination, and the inclination (or dip). Note that the "magnetic equator" is defined as the locus of 0° dip angle on the Earth surface. No special name is given to the locus of 0° declination.
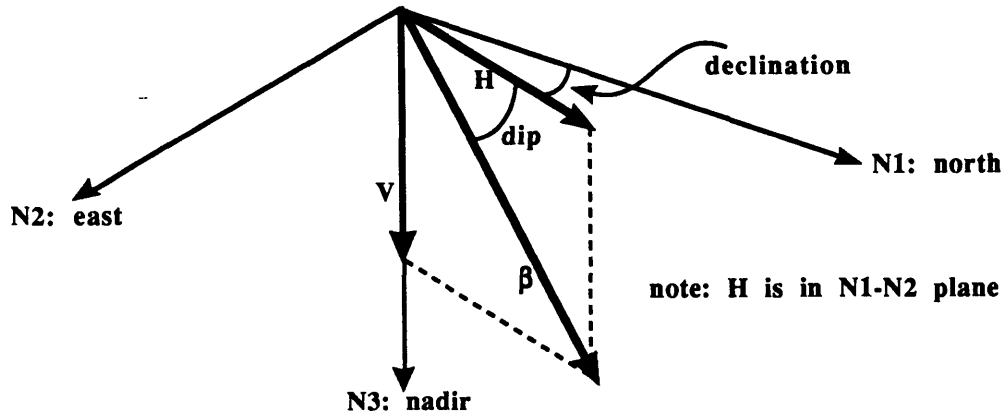
Figure 3.9: definition of Earth field terminology in frame $\mathcal{F}_N$. The field vector is β.

3.2.2 Main Field Modelling

The lack of a theoretical understanding of the terrestrial field's geophysical origins does not prevent the development of an empirically fitted model of the field. Making the assumption that the Earth is spherical with a non-conductive surface (true for the levels of surface currents which would have to exist to significantly invalidate this assumption), Maxwell's equations state the field β must satisfy:

$$\nabla \times \beta = 0 \tag{3.9}$$

$$\nabla \bullet \beta = 0 \tag{3.10}$$

Therefore, a potential function exists for the field which satisfies Laplace's Equation:

$$\beta = -\nabla V \tag{3.11}$$

$$\nabla^2 V = 0 \tag{3.12}$$

For an arbitrary field, the general form of the solution to (3.12) on the surface of a sphere is given by an expansion in terms orthogonal Legendre functions:

$$V(r,\theta,\phi) = R_e \sum_{n=1}^{k} \sum_{m=0}^{n} (\frac{R_e}{r})^{n+1}(g_{mn}\cos(m\phi) + h_{mn}\sin(m\phi))P_{mn}(\theta) \tag{3.13}$$

Where: R_e = sphere (Earth) radius (equatorial)

  r = distance from sphere center

  ϕ = East longitude, θ = co-latitude

g_{mn}, h_{mn} = spherical harmonic coefficients (dimension of flux density)

$P_{mn}(\theta)$ = associated Legendre function

Precise measurements of the Earth field at grid points around the world are used to perform a least squares fit to determine the coefficients g_{mn} and h_{mn}, which comprise the IGRF- International Geomagnetic Reference Field. The IAGA (International Association of Geomagnetism and Aeronomy) updates the IGRF model every 5 years, publishing measurements, model coefficients and estimates of their secular variation for the near future, and grid point measurements. Data from IGRF 1985 published in [IAGA86] is used in this thesis. Computationally efficient recursion formulae exist for P_{mn} and the derivatives $\beta = -\nabla V$ which are used to compute the IGRF as a function of (r,θ,ϕ). A MATLAB implementation based on [Mali81] is given in Appendix B.

IGRF 1985 expands (3.13) to tenth order (k = 10). Generally, the full expansion is used in simulation, while lower order approximations exist for the purpose of visualization and deriving rough but analytically tractable results. The crudest (and most familiar) approximation is the spin-axis dipole model, comprising (3.13) with n = 1, m = 0:

$$V(r,\theta,\phi) = \frac{R_e^3}{r^2} g_{01} \cos(\theta) \qquad (3.14)$$

The approximation consists of a dipole with moment $(R_e^3 g_{01})$ aligned with the geographic poles (note no ϕ dependence in (3.14)). The actual geomagnetic poles are offset from the geographic poles by 11.5°, the geomagnetic north pole being in Greenland at 78.5° N and 69° W. This tilt is accounted for by allowing m = 1 terms in the n = 1 approximation of (3.13), forming the tilted dipole model:

$$V(r,\theta,\phi) = \frac{R_e^3}{r^2} \{g_{01}\cos(\theta) + g_{11}\cos(\phi)\sin(\theta) + h_{11}\sin(\phi)\sin(\theta)\} \qquad (3.15)$$

The quadrupole allows all n = 1 and n = 2 terms in (3.13)- 8 coefficients total. [NASA69-1] contains the potential expression, which will be skipped here. [NASA69-1] also gives maximum magnitude deviations from an 8th order model of 60% in LEO for the tilted dipole, 40% for the quadrupole. Maximum angular deviations in LEO are 25° for the tilted

dipole, 15° for the quadrupole. The error maxima are functions of latitude, and occur within the latitude range of a 28° inclination orbit.

3.2.3 Field Transients and Magnetometer Filtering Considerations

The spacecraft magnetometer output provides a measurement of the local field vector to translate a torque command from the attitude control loop to a dipole command (i.e. equation 3.8). To limit noise, it is desirable to low pass filter the magnetometer signal using as low a bandwidth as possible. However, if the filtered output is significantly different than the actual field, the achieved torque will differ from the commanded torque, and the control loop will suffer. Therefore, it is necessary to characterize the spectrum of the field vector's variation in orbit.

The actual Earth magnetic field is composed of the "main field," which is modeled by the IGRF and subject to long term (secular) variations, and shorter term transient variations imposed on the main field, which are ignored by the IGRF. Secular variations are small and slow (<150 nT/yr according to [NASA69-1]) variations whose origins are most likely geophysical. On the other hand, transient variations are disturbances caused by interactions with the active plasma environment of the solar system. The IGRF models the main field, the spherical harmonic coefficients and their estimated secular variations being computed from highly filtered, multiple ground measurements- and is accurate to within 100 nT in the 8th order expansion for LEO [Wert78]. As will be detailed in sections 3.2.4 and 3.2.5 below, the main field as seen in $\mathcal{F}_B$ has no power in components past about 6 time orbital rate (orbital period is ~95 minutes). Transient variations are not as predictable, but have been observed sufficiently to characterize spectrally. The information in table 3.2 is collected from [NASA69-1] and [Tasc88].

As a magnitude comparison, the proposed magnetometers have a dynamic range of +/- 60 μT, with 10 bit (1024) discretization. Thus 1 bit corresponds to 117 nT. Furthermore, even with on-orbit calibration of the magnetometer's biases and misalignment, it is not expected that the field measurement with respect to the defined body reference frame could be any better than within 500 nT[1][pers91] [Lern79]. The proposed control loop clock rate is 1 Hz. From Table 3.2, it is then seen that a filter in the decade 0.05 to 0.5 Hz will not obstruct any transients of magnitude significant compared to the figures above. Therefore, we

---

[1]This does not imply magnetometer accuracy to 500 nT per axis is necessary. The controllers designed in Chapter 5 can tolerate considerably worse.

conclude that natural field transients is not a significant factor in the magnetometer filter design.

Name:	Observed Magnitude Bound:	Characteristic Times:	Comments:
diurnal variations	400 nT	daily	smoothly oscillating over day long period
auroral electrojet disturbance	2500 nT	~ 10 min.	mostly near polar regions, duration: 30 min. - 2 hrs
magnetic storm- sudden commencement phase	50 nT	1 to 6 minutes	sudden increase, simultaneous world wide
magnetic storm- main and recovery phases	1000 nT lower latitude 2500 nT polar	several hours	slow changing recovery to pre-storm levels, with several possible over/under shoots; duration: several days
sudden impulse variation	20 nT	~ 5 minutes	duration ~ one hour
micropulsations	< 5 nT	0.1 thru 600 sec.	small oscillations

Table 3.2- Field transient characteristics in LEO. Note that the characteristic time column denotes approximate rise/fall time or oscillation period, not duration. It is thus a rough handle on bandwidth.

3.2.4 The Main Field Model in Orbit

Figures 3.10 thru 3.12 show, respectively, the total intensity, dip, and declination of the Earth field, from [IAGA86]. Although taken from data on Earth surface, the plots are reasonably representative of the field in LEO. It is seen from (3.11) and (3.13) that the dipole component of the field change inversely with $(R_e/r)^3$, with the higher order components falling faster. At 555km, $(R_e/r)^3 = 87\%$.

Note that the total field intensity encountered for a 28° inclination orbit range approximately from 20 μT to 50 μT, with average around 30 μT. The dip angle varies tremendously, at least ± 50°, and fairly uniformly about the magnetic equator, which is within ± 10° latitude of the equator except going by the SAA (South American Anomaly). The declination, however, is small- at worst 20°, but average somewhere between 0 and 10°. This suggests that, in orbit and with respect to $\mathcal{F}_N$, the North component is fairly constant and biased, the East component is small and unbiased, while the Down component will

undergo large fluctuations with peaks greater than the North component. Field simulations in Appendix C-1 show this clearly.
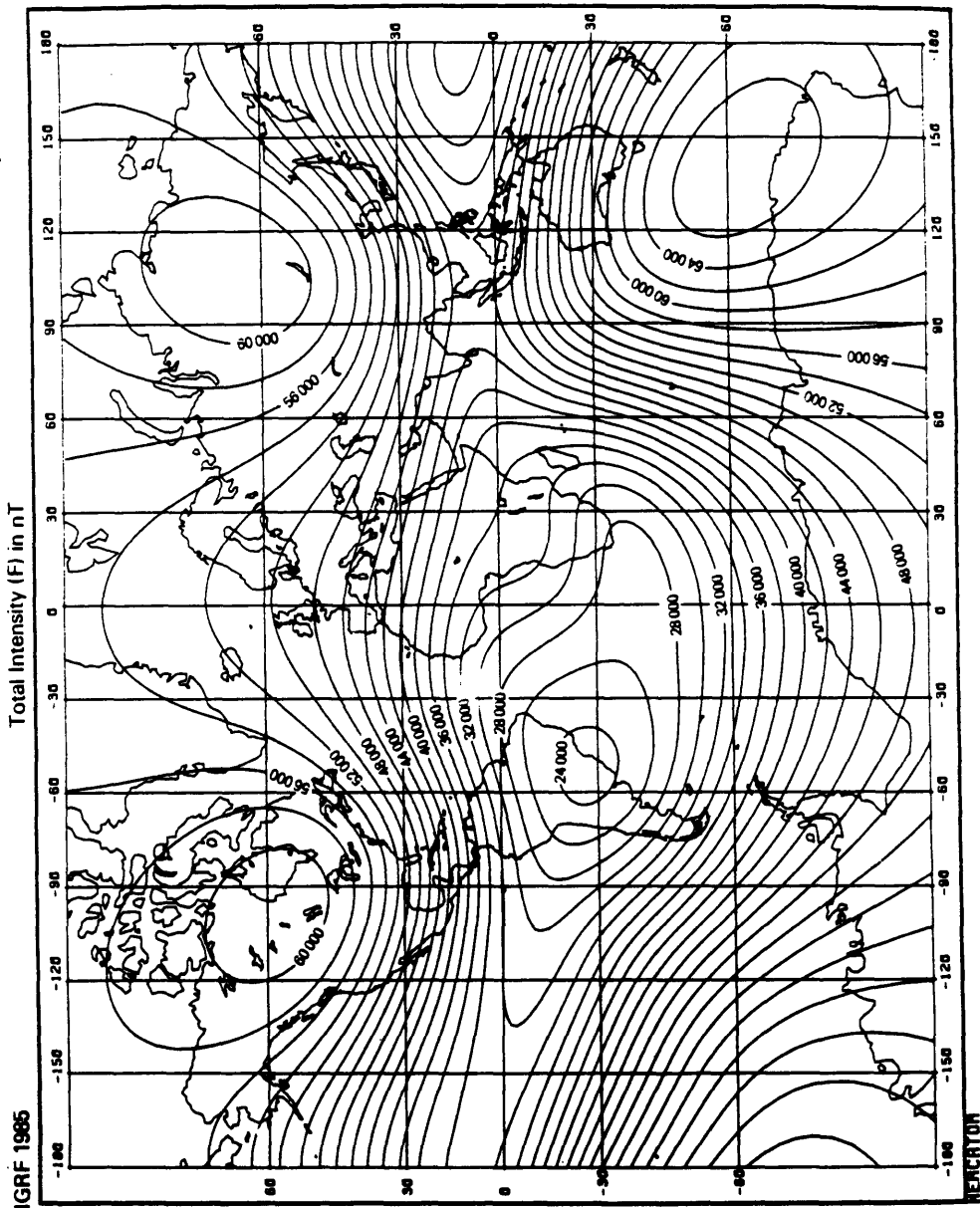
Figure 3.10: Surface field Total Intensity in nT, from [IAGA86]
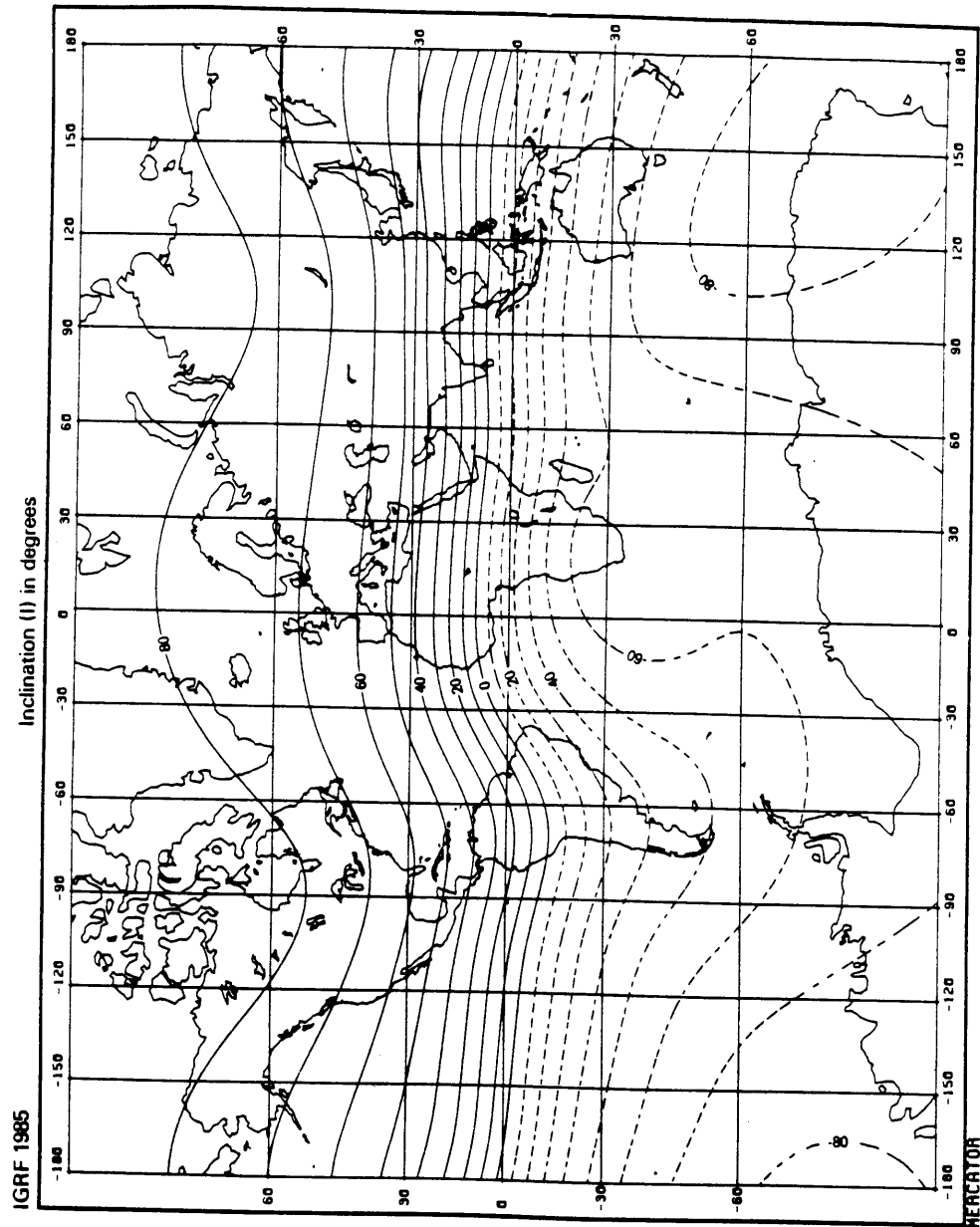
Figure 3.11: Surface field Inclination (Dip) in degrees, from [IAGA86]
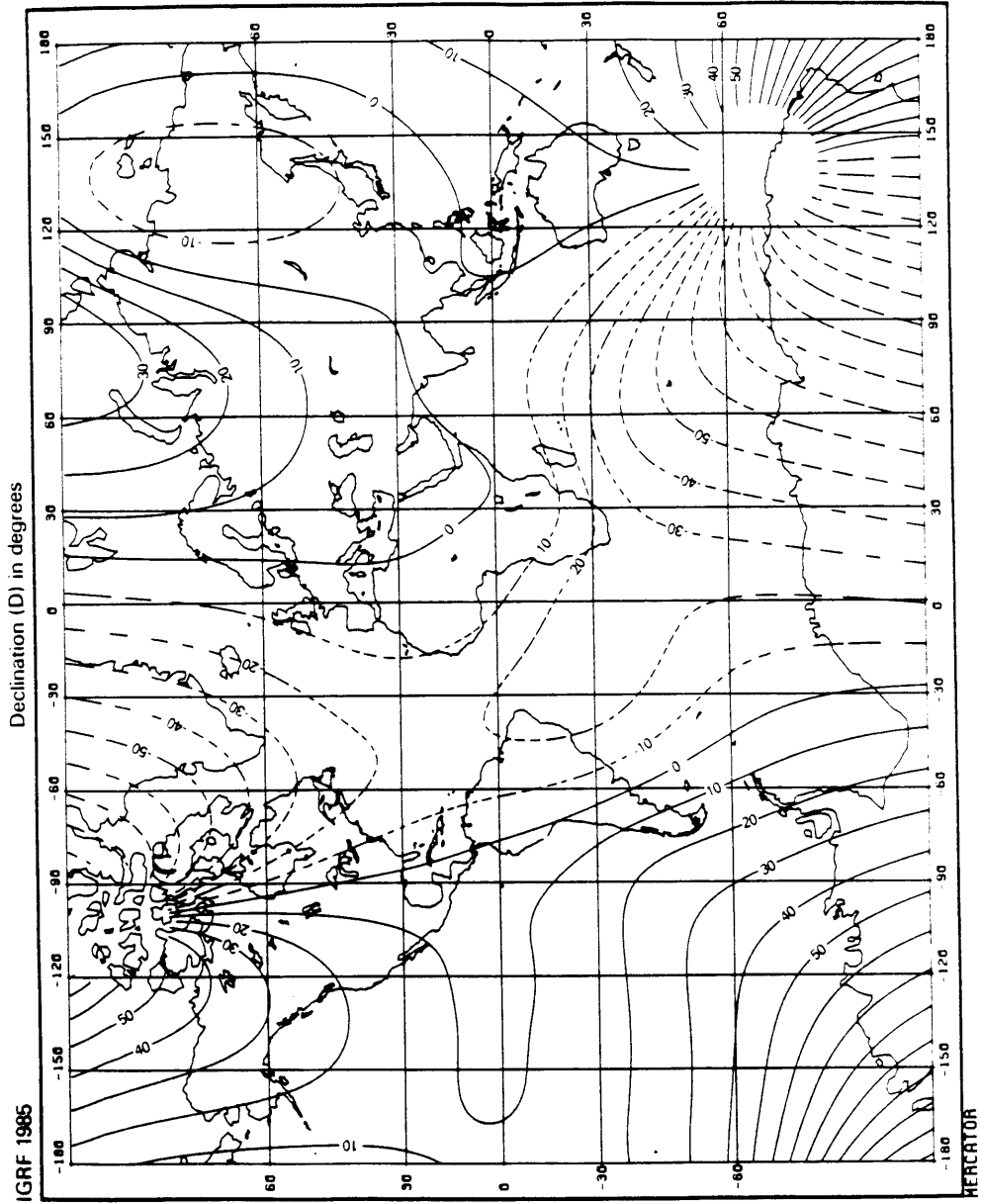
Figure 3.12: Surface field Declination, from [IAGA86]

We can roughly describe the Earth field seen in $\mathcal{F}_B$ as well. The nominal spacecraft attitude is sun-pointing, and $\mathcal{F}_S$ is inertial over the expected mission life span. Since the orbit node regresses, the spacecraft body frame $\mathcal{F}_B$ in a sun-pointing attitude can vary with respect to $\mathcal{F}_N$ at that particular longitude/latitude by as much as 51 degrees[1] along some axis, depending on the current combination of time of year, right ascension, and true anomaly. As discussed above and shown in Appendix C-1, the Earth field seen in a 28° orbit has a roughly zero-mean Down (nadir) component of slightly greater magnitude than the North component, which for this low inclination orbit is well above zero mean. (The East component is small compared to the other two.) Therefore, conclusion is that the field viewed in $\mathcal{F}_B$ and as a function of time contains a large component at the node regression rate which will distribute the North and Down components across all three <u>body axis</u>.

The spectrum of the main field seen in orbit with respect to $\mathcal{F}_N$ is also shown in Appendix C-1, where it is seen that almost no significant components exist past 6 times orbit rate. Of course, the magnetometers measure with respect to $\mathcal{F}_B$, but as argued above, this adds only a component at the node precession rate, which is approximately 1/800 orbit rate. Therefore, <u>there is no need in simulation to explicitly model either the field's transient variations or the magnetometer filter</u>. The assumption is simply that one takes care of the other.

To better visualize the long term trends of the field seen in $\mathcal{F}_B$ from orbit, the admittedly brute force approach of simulation was taken in favor of extracting information from the field approximations of (3.14) and (3.15). Results are presented in Appendix C-2. The large component at the node regression rate is clearly seen, with one significant conclusion being that the field about all three axis of $\mathcal{F}_B$ will switch signs over time regardless of roll angle about the sun vector. This essentially states that there is no "preferred" attitude about which a simpler control strategy may be pursued, and drives the discussion in Chapter 4.

---

[1] assuming orbit inclination = 28°, add to equatorial obliquity of 23° from ecliptic- see Figure 4.16

3.3 Disturbance Torque Models

Since the attitude control scheme selected for HETE requires no expendables, and since relatively little freedom in vehicle configuration is left after launcher size, instrument field of view and on-orbit power requirements are satisfied, the disturbance calculations do not drive the vehicle configuration. Thus, only rough bounds on the disturbance magnitudes are required. However, since drift rate performance of the controller is a key specification, it is important to consider the spectra of the disturbances encountered as well. The contents of this section are presented with these goals in mind.

3.3.1 Representing the Disturbance Spectrum

The Fourier transform of a _vector_ time function is defined as the Fourier transform of each component _scalar_ function. It would therefore be straightforward to take the results of a disturbance model, as represented by a set of time varying vectors with respect to $\mathcal{F}_B$, and show FFTs of each component. However, the purpose of examining the disturbance spectra is to aid control design. In light of this, it is shown here that a more useful quantity to consider is the magnitude of all three Fourier components as a function of frequency.

In continuous time, consider the generic system with input $\underline{d}(t)$, output $\underline{y}(t)$, and state $\underline{x}(t)$:

$$\underline{\dot{x}} = A\underline{x} + B\underline{d} \qquad \underline{y} = C\underline{x}$$

$$G(s) = C(sI - A)^{-1}B$$
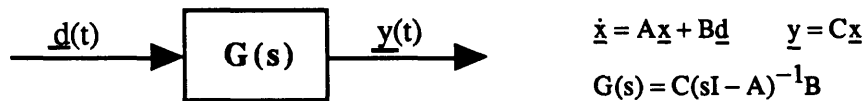
We have in mind the specific case that $\underline{d}(t)$ represents a disturbance torque input to a closed loop system $G(s)$, with $\underline{y}(t)$ representing attitude or rate outputs. The Fourier pair for the input vector can be expressed:

$$\underline{\hat{D}}(\omega) \equiv \begin{bmatrix} \hat{D}_1(\omega) \\ \hat{D}_2(\omega) \\ \hat{D}_3(\omega) \end{bmatrix} = \int_{-\infty}^{\infty} \underline{d}(t)e^{-j\omega t}dt \qquad (3.16)$$

$$\underline{d}(t) \equiv \begin{bmatrix} d_1(t) \\ d_2(t) \\ d_3(t) \end{bmatrix} = \int\limits_{-\infty}^{+\infty} \left[\frac{1}{2\pi} \hat{\underline{D}}(\omega) \right] e^{j\omega t} d\omega \tag{3.17}$$

To simplify the presentation (the general case is notationally cumbersome), consider the case of:

$$\frac{1}{2\pi} \hat{\underline{D}}(\omega) = \underline{d}_m \delta(\omega - \omega_o) \tag{3.18}$$

In other words, in the time domain $\underline{d}(t)$ is a single frequency complex exponential with magnitudes $\underline{d}_m$:

$$\underline{d}(t) = \underline{d}_m e^{j\omega_o t} \tag{3.19}$$

Assuming that the system G(s) is strictly stable and steady state has been reached, the response to $\underline{d}(t)$ is then a vector function of appropriate dimension and at the same frequency as the input:

$$\underline{y}(t) = G(j\omega_o)\underline{d}_m e^{j\omega_o t} \tag{3.20}$$

We could then relate the magnitude vectors of the input and the output:

$$\underline{y}_m \equiv G(j\omega_o)\underline{d}_m \tag{3.21}$$

But since (3.21) is simply a linear transformation from input to output space by the matrix $G(j\omega_o)$, the following singular value relations apply:

$$\left| \underline{y}_m \right| \leq \sigma_{max}\{G(j\omega_o)\} \left| \underline{d}_m \right| \tag{3.22}$$

$$\left| \underline{y}_m \right| \geq \sigma_{min}\{G(j\omega_o)\} \left| \underline{d}_m \right| \tag{3.23}$$

where:

$|v|$ denotes the Euclidean of the complex vector $\underline{v}$

$\sigma_{max}(M)$, $\sigma_{min}(M)$ denote the largest and smallest singular values of a matrix M, respectively

Although the phase information is ignored, the relations (3.22) and (3.23) still say more about the behavior of the system in response to $\underline{d}(t)$ than the actual components of $\underline{d}_m$ alone. Thus, unless the system is such that the separation between σ_{min} and σ_{max} is very large, or that the input/output singular vectors point in intuitively meaningless directions (neither is the case for us- see Chapter 5), (3.22) and (3.23) suggest that an insightful way to represent the spectrum of a given input vector $\underline{d}(t)$ is to plot the one magnitude function of its Fourier decomposition $\left|\frac{1}{2\pi}\hat{\underline{D}}(\omega)\right|$ as a function of ω, instead all three components of the decomposition individually.

Additionally, note that (3.16) is in complex exponential form, making it necessary to keep track of positive and negative frequencies ω. If we are willing to discard the phase information, and provided the function $\underline{d}(t)$ is real, the conjugate symmetry property of Fourier transforms states that:

$$\left|\hat{\underline{D}}(\omega)\right| = \left|\hat{\underline{D}}(-\omega)\right| \tag{3.24}$$

Therefore, it would be less confusing to consider the function:

$$\left|\frac{1}{\pi}\hat{\underline{D}}(\omega)\right| \quad \text{for } \omega \geq 0 \text{ only} \tag{3.25}$$

In reality, analytical expressions for the disturbance $\underline{d}(t)$ are substituted by outputs of disturbance models simulated in discrete time steps. The discrete time equivalent of (3.25) is straightforward to derive; the only trick is to be careful about the scaling factor changes.

The DFT/FFT pair for the input vector $\underline{d}[n]$ is:

$$\underline{d}[n] = \sum_{k=0}^{N-1}\left\{\left[\frac{1}{N}\underline{D}(k)\right]e^{j\left(\frac{2\pi k}{N}\right)n}\right\} \tag{3.26}$$

$$\underline{D}(k) = \sum_{n=0}^{N-1} \left[\underline{d}[n] e^{-j\left(\frac{2\pi n}{N}\right)k} \right] \qquad\qquad (3.27)$$

where: n = discretized time step,
k = normalized frequency bin
N = length of window in DFT

$\underline{D}[k]$ is defined for k up to N only, with N/2 being the normalized Nyquist rate, and the top half of the window (N/2 < k ≤ N) is the counterpart to the negative frequency $\omega < 0$ half of the continuous time spectrum. The singular value relations (3.22) and (3.23) apply also to the response of a discrete time system G(z) to $\underline{d}[n]$. Again, it is less confusing in considering the magnitude of $\underline{D}[k]$ to exploit conjugate symmetry:

$$\left|\underline{D}[k]\right| = \left|\underline{D}[N-k]\right| \qquad \text{for real } \underline{d}[n] \qquad\qquad (3.28)$$

We therefore plot the function:

$$\left|\frac{2}{N}\underline{D}[k]\right| \qquad \text{for } 0 \le k \le \frac{N}{2} \text{ only} \qquad\qquad (3.29)$$

Equation (3.29) is used to present the spectrum of disturbance models in the sections to follow. Plots normalized according to (3.29) approximate the continuous spectrum with the correct magnitude over the half the window. To bolster intuition, plots will be labeled with continuous time spectral units []/Hz, where [] is the dimensions of the time domain quantity.

Note that (3.28) shows $|\underline{D}[0]| = |\underline{D}[N]|$, but $\underline{D}[N]$ is undefined by (3.27). Unlike every other element, the magnitude of the entire DC component is stored in $|\underline{D}[0]|$, making the 2/N normalization in (3.29) invalid for the single element $|\underline{D}[0]|$ alone. To be clear, the short code used to implement this is shown in Appendix A-3d.

Note also that the choice of normalization for the frequency bin k is not specified- a particularly evocative normalization is to units of orbital frequency, which is used in the following sections.

3.3.2 Solar Pressure Torque

Solar pressure disturbance torque is the easiest to estimate on HETE because the mission attitude is nominally sun-pointing or at least at a constant angle with respect to the sun. The disturbance is well approximated as DC, which is the easiest to reject. Furthermore, since the face presented to the sun while in mission attitude is largely symmetrical, the solar pressure torque is mostly negligible compared to the magnitude of other environmental disturbances present. Components about the body 1 and 2 axis are caused by paddle area imbalance, while torque about the body 3 axis is due to solar propelloring. We estimate some numerical values below.

Grossly, the interaction between between radiation of momentum flux P (P has units force/area) and a surface can be lumped into three types- absorption, diffuse reflection, and specular reflection. Considering an elemental area dA, define the quantities N = surface normal unit vector of dA, S = sun-pointing unit vector, and θ = included angle between N and S. Then [Wert78] gives:

Interaction:	Force on dA:	Direction:
absorption	$dF_{abs} = -PC_a\cos(\theta)SdA$	along S
diffuse reflection	$dF_{dif} = PC_d(-\frac{2}{3}\cos(\theta)N - \cos(\theta)S)dA$	along S and N
specular reflection	$dF_{spe} = -2PC_s\cos^2(\theta)NdA$	along N

where C_a, C_d, and C_s are defined as the fraction of the incident radiation which interacts in the denoted matter. Note by definition, $(C_a+C_d+C_s) = 1$. The total force is then the sum of all three expressions:

$$dF = -P\cos(\theta)\left\{(C_a+C_d)S + 2\left(C_s\cos(\theta)+\frac{1}{3}C_d\right)N\right\}dA \tag{3.30}$$

For a sun-pointing attitude, $\theta = 0$, $N = S$, and (3.30) simplifies greatly,

$$dF = -P\left(C_a + C_d + 2C_s + \frac{1}{3}C_d\right)dAN$$

$$= -PN(1 + C_s + \frac{1}{3}C_d)dA$$

$$\approx -2PdAN \qquad \text{choose upper bound of } C_s + \frac{1}{3}C_d \leq 1 \qquad (3.31)$$

We proceed to estimate numbers for area imbalance and propelloring torques. [Wert78] gives $P = 4.5 \times 10^{-6}$ N/m^2 as the solar momentum flux in Earth orbit. To a good approximation, the only surfaces which need be considered are the solar paddles. Assuming the center of pressure for each paddle is at half length, and ignoring the curvature of the paddles, the net torque due to area imbalance from the force model (3.31) can be written:

$$\tau = \pm 2P(A_p f_x)LB_1 \pm 2P(A_p f_y)LB_2 \qquad (3.32)$$

where: L = paddle length (69 cm)
        Ap = paddle area (0.19 m^2)
        P = momentum flux (4.5 $\times 10^{-6}$ N/m^2)
        f_x, f_y = net area imbalance in fraction of Ap along noted axis
        B_1 B_2 = unit vectors in body frame

whereas the torque from all 4 paddles propellored at some angle θ and experiencing force (3.31) can be crudely modelled:

$$\tau = 4PA_p \sin(\theta)LB_3 \qquad (3.33)$$

With upper bounds of say 5% on f_x and f_y and 3° on θ, (3.32) has components at 5.8×10^{-8} N•m, while (3.33) has 1.2×10^{-7} N•m. These are used as the estimate of solar pressure torques and will be shown to be small compared to other disturbances.

3.3.3 Gravity Gradient Torque

The torque due to non-uniform gravitation field distribution on a finite body is well documented and fairly straightforward to estimate. The details of its derivation will be skipped here. The following assumptions are generally accepted as being adequate:

- one celestial primary, uniform spherical mass distribution
- single body spacecraft
- characteristic length of spacecraft << distance to primary center

When integrating the effect over the spacecraft body, the last result allows dropping terms of higher order than (r/R), where r is the distance from the spacecraft center of mass to a mass element under consideration, and R is the distance from that mass element to the primary (Earth) center. Expressing the torque in spacecraft fixed body frame: (see derivation in [Hugh86])

$$\underline{\tau} = 3\left(\frac{\mu}{R_c^3}\right)\underline{C}_3 \times I\underline{C}_3 \qquad (3.34)$$

Where: $\mu = Gm_e = 3.986 \times 10^{14}$ (N$\cdot$m^2)/kg

R_c = distance from mass center to Earth center

    = 6.93×10^6 m for 300 nautical mile orbit

I = (3x3) inertia matrix expressed in body fixed frame

$\underline{C}_3$ = (3x1) third column of C_{BL}

We are immediately able to put an upper bound on (3.34). Since no product of the elements of $\underline{C}_3$ can be greater than unity, mass properties given in Chapter 3 for I gives $|\tau|$ ≤ 5.9×10^{-6} N$\cdot$m. Simulations in sun-pointing attitude at 28° inclination show this is pessimistic by about a factor of 2. The form of (3.34) also makes the spectral content of the torque clear. If $\mathcal{F}_B$ maintains its sun-pointing attitude, it is inertial on the time scale of an orbit period. The periodic component of each element of $\underline{C}_3$ thus varies with frequency equal to orbit rate. The inertia matrix I is not time varying in $\mathcal{F}_B$. Since for any arbitrary vectors $\underline{a}$ and $\underline{b}$, ($\underline{a} \times \underline{b}$) = (-$\underline{a} \times$ -$\underline{b}$), the conclusion is that any periodic component of (3.34) must oscillate at twice orbit rate. This is clearly shown below.

The distribution between the DC and twice orbital rate components of $|\tau|$, as well as the conservatism of the upper bound estimate above, are not easy to see. Simulation of the orbital variations of $\underline{C}_3$ is relatively simple- the code is shown in Appendix A-3e. Figure 3.13 shows a fairly typical result using the mass properties given in Chapter 3 and for a 28° inclination 300 nautical mile orbit. Variations from this spectrum are not observed to be large for different relationships between $\mathcal{F}_B$ and $\mathcal{F}_L$.
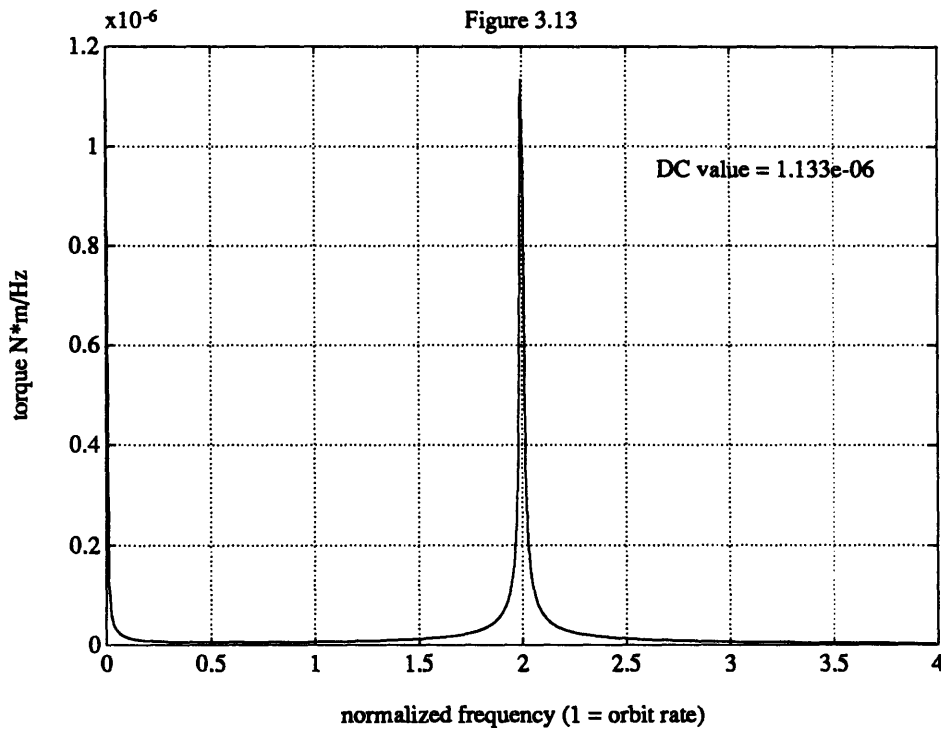
Figure 3.13: FFT of gravity gradient torque model for given mass properties, 28° inclination orbit, sun-pointing attitude. Magnitude normalization is as per equation (3.29)

3.3.4 Residual Dipole Disturbance Torque

It is difficult to say much about this source of disturbance in advance of spacecraft construction, as the magnitude (and certainly the direction) of the residual dipole on the spacecraft is unknown. Experience is of some help. [NASA69-2] suggests an initial estimate of between 1×10^{-3} and 3.5×10^{-3} A•m^2/Kg of non-spinning spacecraft for cases where sound magnetic design practices are followed and compensation is performed after

assembly. Choosing then a dipole strength of, say, 0.2 A•m^2 gives for orthogonal (worst case) torque magnitude in a 25μT field of 5x10^{-6} N•m of torque, which is still potentially the largest disturbance encountered. This suggests that care should be taken to control the spacecraft dipole prior to flight.

It is interesting to see the spectrum of the disturbance torque incured by a constant dipole in orbit. Figure 3.14 a,b show a pair of representative plots for a 0.2 A•m^2 dipole in arbitrary directions, 28° inclination 300 nautical mile orbit, sun-pointing attitude, 8th order IGRF model. The code is again in Appendix A-3e. It is observed from several simulations that through variations on the particular orbit, attitude, and residual dipole direction, the torque is distributed mostly between DC and twice orbital rate, showing as expected the dominance of the dipole component of the main field.
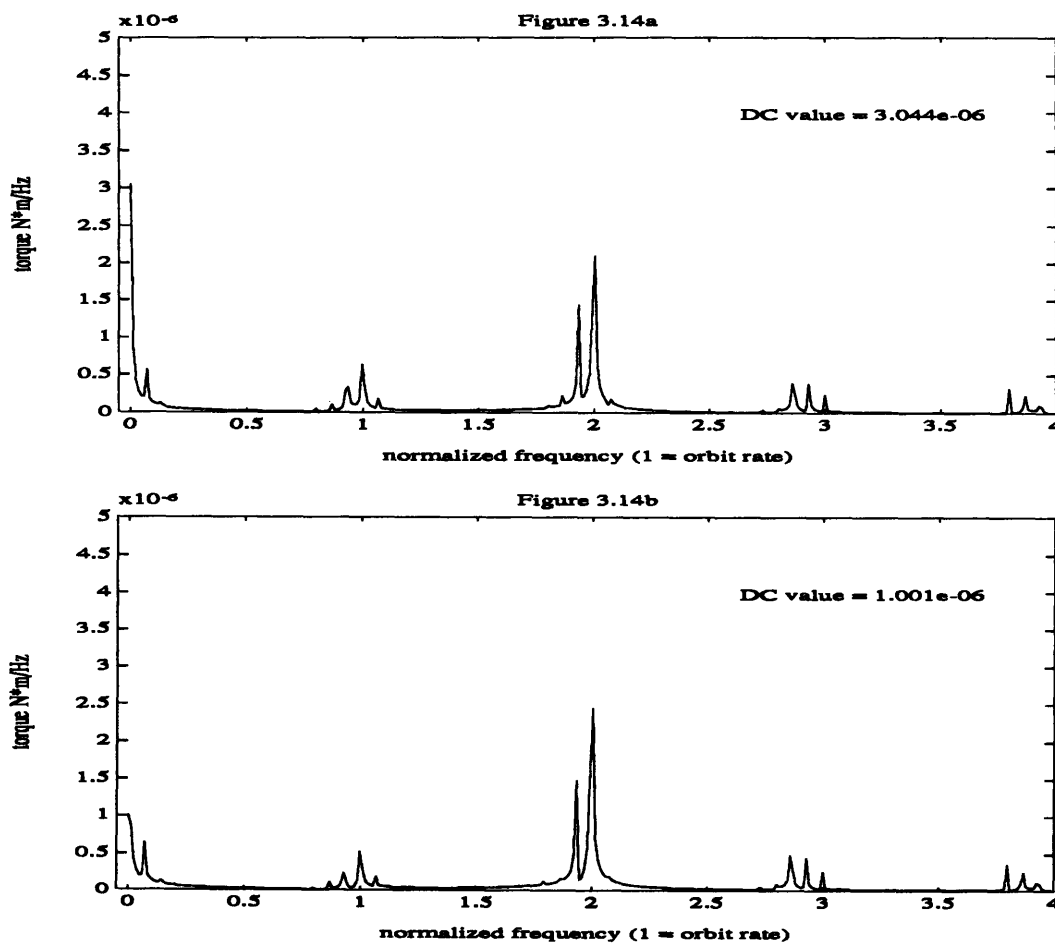
Figure 3.14 a,b: two spectra of disturbance torques from a constant dipole of 0.2 A•m^2 in different orbits. Note the only substantial difference is in the DC component.

3.3.5 Aerodynamic Disturbance Torque

The first thing to note about modelling aerodynamic drag is that it scales with ρv^2, where ρ is the density and v is speed with respect to the atmosphere. Both quantities are subject to significant uncertainties. To a first approximation the atmosphere rotates with the Earth, although [NASA69-3] and [Hugh86] both point out significant variations from this occur. The usual approach is to bury the v uncertainty in a conservative value of the drag coefficient, and use inertial orbital velocity for v. At 555 km, $v = 7.6$ km/s.

The density is subject to even greater variations. Figure 3.15 is taken from data in [Wert91] and shows the 2 orders of magnitude variation possible at 550 km, the intended orbit altitude. The density variation is mostly related to the temperature variation in the thermosphere (120 to 600 km) caused by varying solar ultra-violet output, the most significant components of the variation being the daily cycle and the 11 year solar cycle. The nominal 3 year lifespan of HETE is intended to span a period of relative solar inactivity, thus for subsequent discussion the estimate of $\rho = 2 \times 10^{-13}$ kg/m^3 is assumed. Note this is almost certainly conservative during orbit night, when the attitude rate control specification is most stringent.
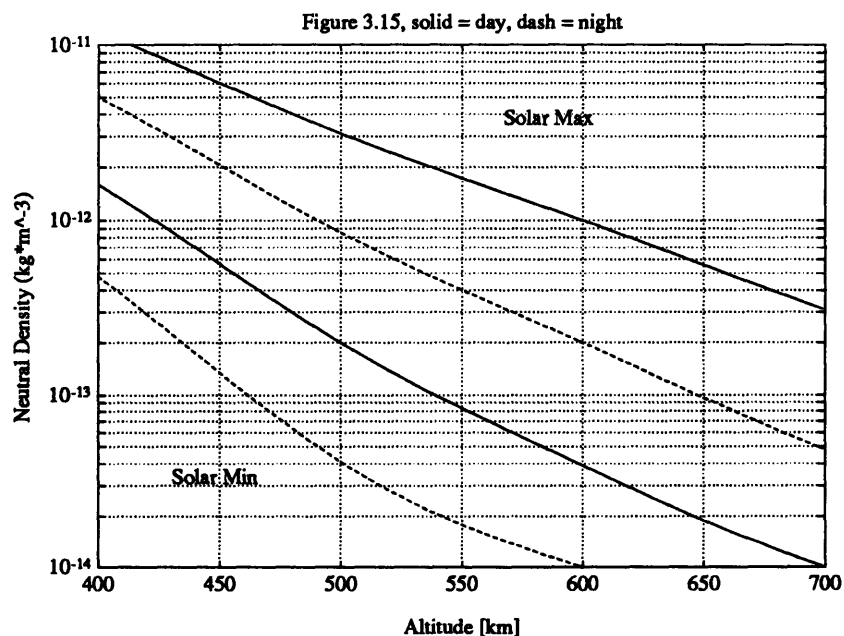
Figure 3.15: Density variation with respect to altitude. Data from [Wert91]. Top band represents Solar Max values; bottom band Solar Min. The dashed lines are orbit night values, solid for orbit day.

Again it is noted that the attitude control system uses no expendables, and that its configuration is largely fixed from other considerations. Therefore, an extremely detailed model of the aerodynamic torque which would capture more than gross changes in configuration is largely unnecessary. Furthermore, it is seen based on the discussion above that any modeling approach which chases small difference in the magnitude of the disturbance is pointless. Therefore, we seek only to capture grossly the disturbance spectrum expected and to generate "realistic" disturbances for control performance simulations. The following simplifications to the model are thus justified:

- Use of the simple molecule-surface interaction:

$$dF = \frac{dA}{2}\rho v^2 C_d\left(\hat{V} \cdot \hat{N}\right)\hat{V} \qquad (3.35)$$

$\hat{V}$ = unit velocity vector of flow, $\hat{N}$ = unit surface normal, C_d = drag coefficient

where the assumption is made that an interacting molecule's momentum is completely absorbed by the surface, hence the force being in the same direction as the velocity. C_d is chosen to be 2.4 in the subsequent discussion.

- Use of basic shapes to represent spacecraft surfaces

- Accounting only for the solar panels' contribution (in effect saying the distance from the center of pressure of the main cylinder to the center of mass is negligible in comparison)

A phenomenon which could potentially complicate matters is the shadowing of the panels by the body. Since it is not immediately clear[1] that it can ignored, we investigate its effect by considering how often the shadowing occurs, and what its effect is on the disturbance profile (spectrum). The first issue is readily addressed by considering figures 3.16 and 3.17 below. The figures motivate that for shadowing to be significant:

- The orbit plane must be close to paralleling the ecliptic plane

- Either B_2 or B_1 must be close to being normal to the orbit plane

---

[1]at least not to the author
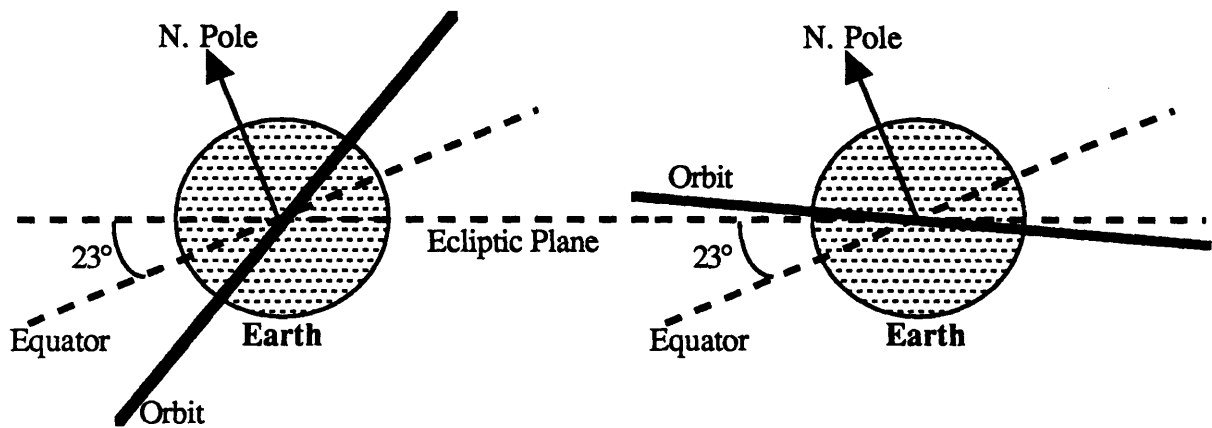
Figure 3.16: The extremes of the orbital plane- ecliptic plane separation, determined by the time of year and the ascension of the orbit. Note that in a sun-pointing attitude the body fixed z axis B_3 is in the plane of the ecliptic, while the orientation of the other 2 axis depend on the roll angle about the sun vector. Note also that the orbital velocity vector (and hence the flow velocity vector) rotates in the orbital plane as seen by an observer fixed to the body.
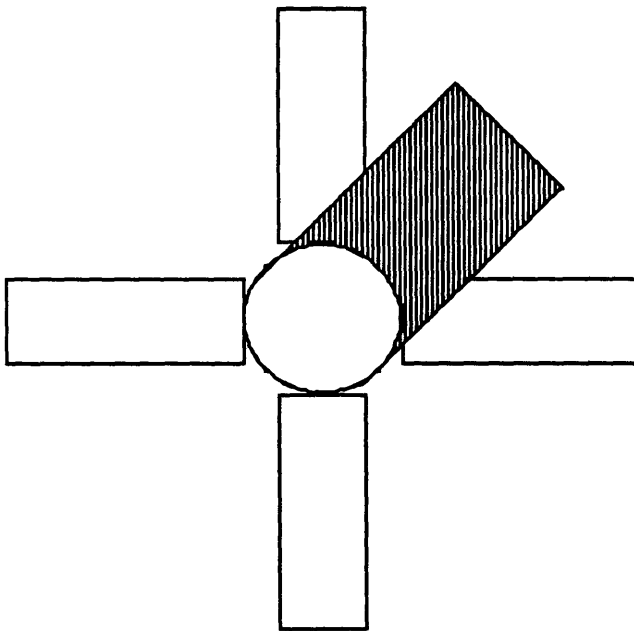
Figure 3.17: roughly scale schematic drawing of HETE viewed from the z axis, showing a possible shadow footprint from a flow velocity vector somewhere in the 2π steradians above the x-y plane.

Worst case occurs when both are satisfied, in which case the the flow velocity vector rotates in either the B_2-B_3 or the B_1-B_3, with the effect of shadowing being confined to that plane. This simplification allows the use of a simple 2-D model to evaluate the worst case effect of shadowing. The calculations are set up below with geometries described by the

figures. Note that in Figure 3.18b, the two bars model the solar panels and are intended to represent areas a distance from the center of mass.

2-D torque model with shadowing:
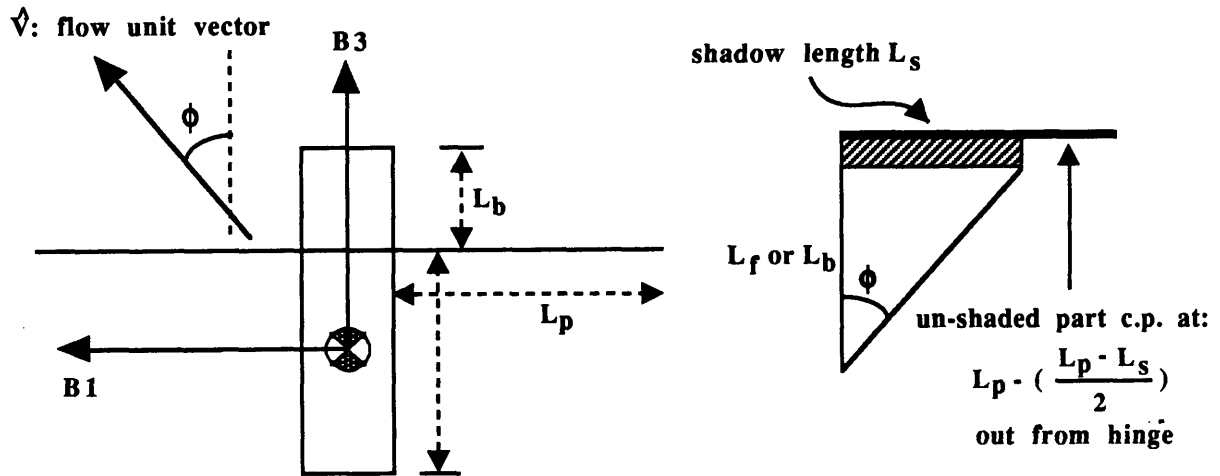
2-D torque model without shadowing
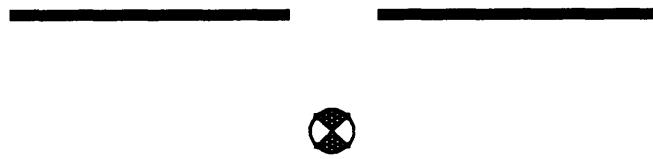
Figures 3.18: gross 2-D models to evaluate effect of shadowing. ϕ rotates 360° per orbit, force model is per eq. (3.35), vehicle dimensions given in Chapter 2. Torque is evaluated by the sum of the cross product between forces and the radii vectors from the c.m. to the respective panels' c.p.'s. The panels are assumed rectangular, making the c.p.'s at the geometric center of the un-shaded portion.

The models are exercised over several orbits and the spectra of the calculated torques are compared in Figure 3.19 (the code is in Appendix A-3f). Clearly, the effect of this worst case shadowing is only to add a component at twice orbital rate, increasing the overall magnitude of the torque but introducing no higher frequency components to the disturbance profile. This observation, coupled with the observation earlier that worst-case

shadowing does not occur too frequently, leads to the conclusion that the considerable increase in complexity associated with modeling the effect of shadowing in 3-D is not worth the effort. We conclude that an extremely simple aerodynamic torque model characterized by:

* geometry boiled down to 4 paddles at prescribed distances from the c.m.
* no shadowing modeled
* force model given by (3.35)

is sufficient for the purposes of generating "realistic" disturbances for simulation, and that in designing the disturbance rejection bandwidth of the controller it is sufficient to leave some extra margin at twice orbital rate. The code for the model is given in Appendix A-3g.
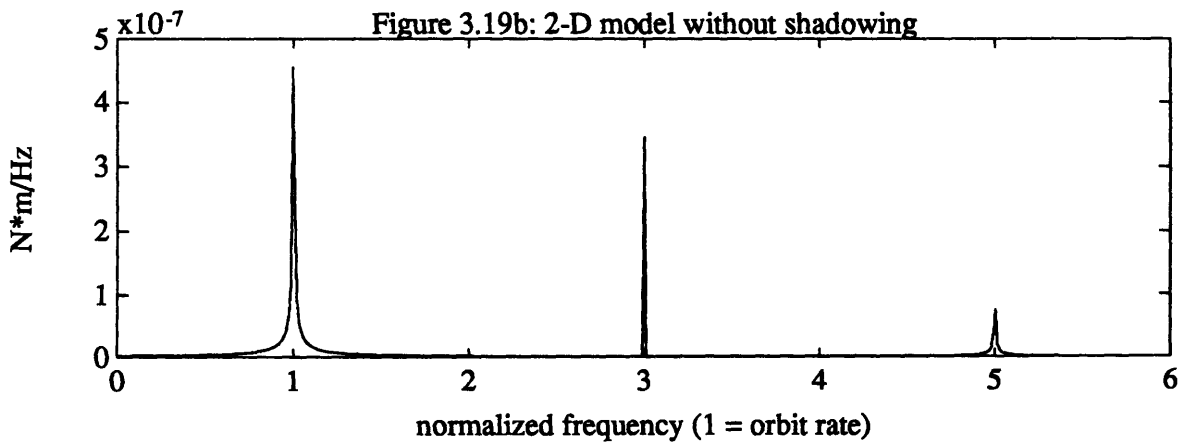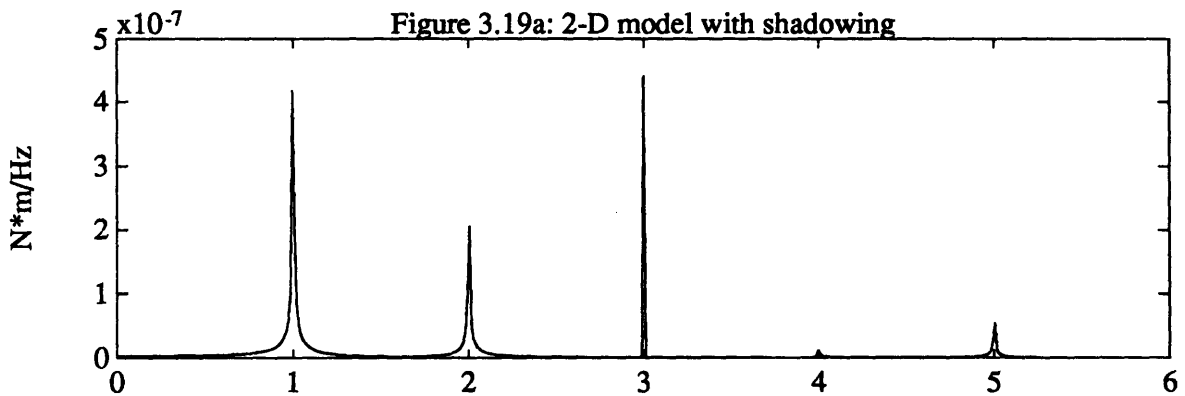
Figure 3.19a,b: FFT of 2-D model outputs. Note difference at twice orbit rate.

3.3.6 Summary

The sections above describe components of an external disturbance torque model which is used to specify the frequency response of the control loops described in the next chapter and to generate support files for control performance simulations. Figure 3.20 show a representative spectrum from the model, which shall serve to motivate disturbance rejection specifications in Chapter 5.
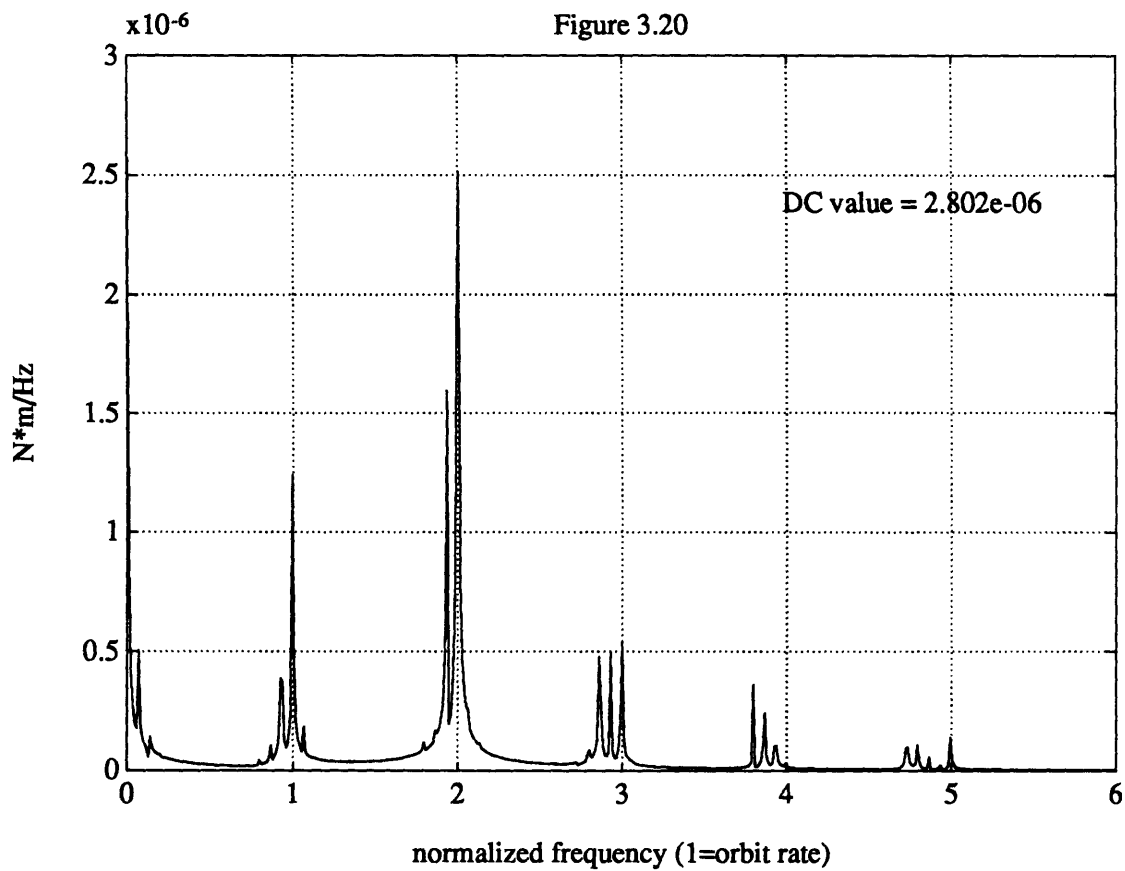
Figure 3.20: Representative spectrum of the net disturbance torque model described in Appendix A-3e.

IV. Control Design Preliminaries

It is said that in control system design, 10% of the effort goes to designing the control law itself, while 90% goes to understanding the plant. This chapter sets up the detailed control design in Chapter 5 by examining issues independent of the control design itself. Dynamics, kinematics, and actuator strategies are discussed. Limits of achievable performance in light of observability/controllability considerations are examined.

The notation for vector quantities established in Chapter 2 is used throughout this chapter.

4.1 Field Constrained Actuator Strategies

The most significant aspect of magnetic attitude control is the limited controllability. As configured, the spacecraft can generate control torques according to (4.1):

$$\tau_c = m \times \beta + \tau_w \tag{4.1}$$

Where τ_w is the torque generated about the wheel axis by commanding an acceleration, m is the commanded dipole vector from the coils, and β is the local Earth field. The constraint imposed by the cross product is that the torque generated by the coils must lie in the plane orthogonal to the β vector (it must also be perpendicular to m, but this isn't really a constraint since m can be chosen). Expressing (4.1) in $\mathcal{F}_B$, where $\mathcal{F}_B$ refers to the body-fixed frame described in Chapter 2, with B_3 = nominal sun pointing axis, B_2 = wheel axis, $B_1 \times B_2 = B_3$:

$$\begin{bmatrix} \tau_{c1} \\ \tau_{c2} \\ \tau_{c3} \end{bmatrix} = \begin{bmatrix} -m_3\beta_2 \\ m_3\beta_1 - m_1\beta_3 \\ m_1\beta_2 \end{bmatrix} + m_2 \begin{bmatrix} \beta_3 \\ 0 \\ -\beta_1 \end{bmatrix} + \begin{bmatrix} 0 \\ \tau_w \\ 0 \end{bmatrix} \tag{4.2}$$

This immediately suggests two possible control strategies, which we designate modes 1 & 2 for convenience:

- Mode 1: "full authority mode." Use τ_w in concert with the magnetic torque to produce a sum torque which can respond to control torque commands in arbitrary directions. Clearly, this cannot work when $\beta_2 = 0$.

- Mode 2: "partial authority mode." Use the spin axis coil m_2 to produce spin[1] plane torques, use τ_w to produce spin axis torques, and use the spin plane coils (m_3 and m_1) for bias momentum management. Note that this is the "traditional" magnetic torquer control strategy. [Wert78]

Evaluation of the options is greatly aided by the angles θ and Ψ defined below:
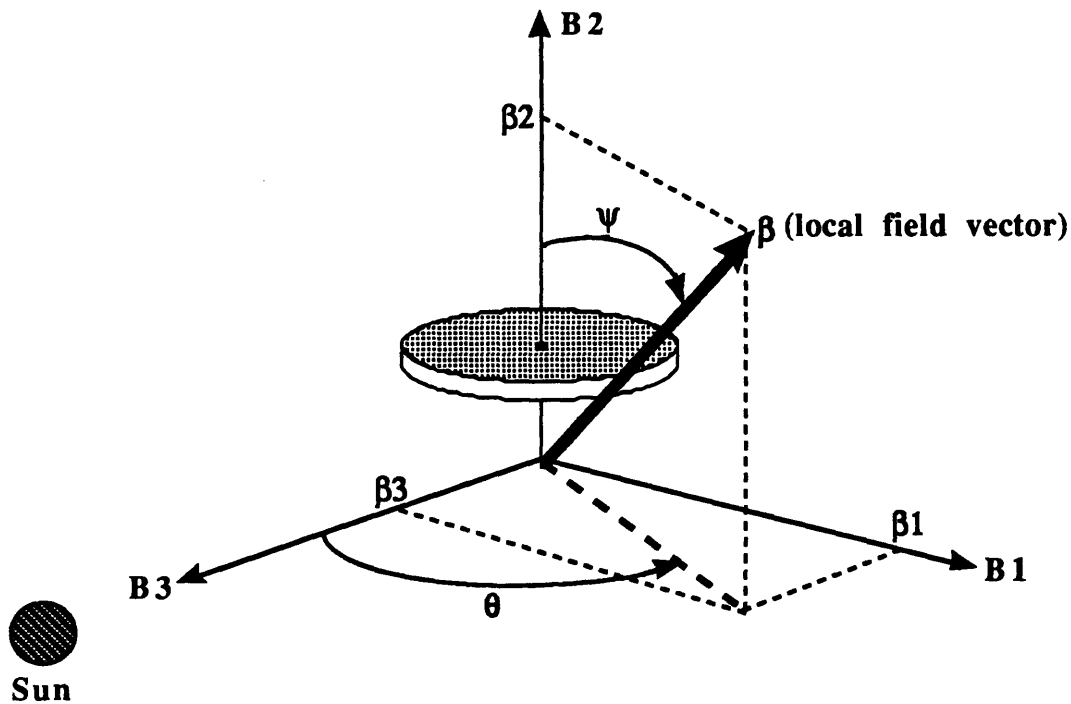
Figure 4.1: definition of θ and ψ in the body fixed frame. The axis of the wheel is fixed to B_2; the direction of the sun vector is nominally controlled to be B_3. β is the local field vector, which when expressed in $\mathcal{F}_B$ has components defined as $[\beta_1\ \beta_2\ \beta_3]^T$.

---

[1]"Spin" is hereafter assumed to mean "wheel spin" unless otherwise noted. Thus the "spin axis" is B_2 and the "spin plane" is the B_1-B_3 plane.

4.1.1 Full Authority Actuator Mode

The problem is now to invert (4.2) to solve for $\underline{m}$ and τ_w given a local field measurement $\underline{\beta}$ and some commanded control vector $\underline{\tau}_c$. Noting that $\underline{m} \times \underline{\beta} = -(\underline{\beta} \times \underline{m})$, we can write:

$$\begin{bmatrix} 0 & \beta_3 & -\beta_2 & 0 \\ -\beta_3 & 0 & \beta_1 & 1 \\ \beta_2 & -\beta_1 & 0 & 0 \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ \tau_w \end{bmatrix} = \begin{bmatrix} \tau_{c1} \\ \tau_{c2} \\ \tau_{c3} \end{bmatrix} \tag{4.3}$$

Clearly, when $\beta_2 \neq 0$, the matrix has rank 3, implying that a family of solutions for $\underline{m}$ and τ_w exist. The apparent extra degree of freedom can be identified by considering the cross product, where $-(\underline{\beta} \times \underline{m})$ is scaled by the sine of the included angle. There is never a reason to impose a coil current penalty by choosing anything but $\underline{m} \perp \underline{\beta}$, and the solution to (4.3) can be made unique by insisting this:

$$\begin{bmatrix} 0 & \beta_3 & -\beta_2 & 0 \\ -\beta_3 & 0 & \beta_1 & 1 \\ \beta_2 & -\beta_1 & 0 & 0 \\ \beta_1 & \beta_2 & \beta_3 & 0 \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ \tau_w \end{bmatrix} = \begin{bmatrix} \tau_{c1} \\ \tau_{c2} \\ \tau_{c3} \\ 0 \end{bmatrix} \tag{4.4}$$

The geometric picture is useful. By insisting on $\underline{m} \perp \underline{\beta}$, $\underline{m}$ is in the same plane as $\underline{m} \times \underline{\beta}$, and perpendicular to it in that plane. However, within that plane, $\underline{m}$ can rotate however we choose it. When $\beta_2 \neq 0$, the vector sum of $\underline{m} \times \underline{\beta}$ and τ_w spans 3-space, allowing the generation of control torques in an arbitrary direction.

Clearly, this actuator mode is inefficient when Ψ comes close to 90°, as large τ_w and $\underline{m}$ are then necessary to vectorally sum to a small $\underline{\tau}_c$. This is discussed in greater detail in section 4.1.3.

4.1.2 Partial Authority Actuator Mode

Mode 2 is used when it is inefficient to use mode 1 during orbit night. It is also used during orbit day, when additional factors come into play:

- the attitude rate specifications are less stringent

- the decreased quality of sensor information makes it pointless to implement a control law which takes proper advantage of arbitrary control torque directionality

- wheel desaturation is easier when the wheel control torque τ_w takes care of external disturbance torques about the spin axis only.

The penalty is of course the additional constraint on the direction of achievable control torques. From (4.2), it is seen that we can generate a torque about the spin axis (B_2), plus a torque in spin plane (B_1-B_3 plane) along the line $\theta \pm 90°$ with m_2. Thus we are no longer able to cancel disturbance torques orthogonal to the spin plane projection of $\underline{B}$. The spin plane coils m_3 and m_1 generate the desaturation torque, along with an undesirable disturbance component in the spin plane proportional in magnitude to β_2.

It is also noted that mode 2 is inefficient when ψ comes close to 0, when the spin plane projection of $\underline{B}$ becomes too small for m_2 to chew on.

4.1.3 Switching Criteria for Orbit Night

As mentioned, since the specifications for attitude rate control are significantly relaxed during orbit day, and only moderate ($\pm 5°$) pointing specifications are imposed, simply turning off the m_2 control loop for the brief (about 100 to 500 seconds- see Chapter 3) period of time when Ψ is close to 0 or 180° to conserve power and prevent continuous actuator saturation is a reasonable strategy. This is not the case during orbit night, when 100 seconds without any cancellation of disturbance torques in the spin plane would produce nutation magnitudes unacceptable for the science instruments. Thus we must consider using both modes 1 and 2 for orbit night, switching between them according to Ψ.

To derive a sound basis for the switching criteria, it is first necessary to make more rigorous the comparisons "Ψ close to 0" or "Ψ close to 90° by seeing how actuator demands scale with ψ. Although in the actual implementation, it would probably be easier to invert (4.4) numerically, the symbolic inversion yields an insightful solution for τ_w:

$$\tau_w = \tau_{c2} + \left(\frac{\beta_1}{\beta_2}\tau_{c1} + \frac{\beta_3}{\beta_2}\tau_{c3} \right) = \frac{1}{\beta_2}[\beta_1 \quad \beta_2 \quad \beta_3] \begin{bmatrix} \tau_{c1} \\ \tau_{c2} \\ \tau_{c3} \end{bmatrix} \tag{4.5}$$

But consideration of Figure 4.1 shows that:

$$\cos(\psi) = \frac{\beta_2}{\|\beta\|} \tag{4.6}$$

Let k be the cosine of the included angle between β and τ_c, then (4.5) can be written simply:

$$\tau_w = \frac{1}{\cos(\psi)}\left(k\|\tau_c\|\right) \tag{4.7}$$

and we see that the "efficiency" of full authority mode scales with the secant of Ψ.

Similarly, consider the torque generated by the spin axis coil m_2, which from (4.2) is seen to be:

$$\tau_{m2} \equiv m_2\beta_p \qquad \beta_p \equiv \begin{bmatrix} \beta_3 \\ 0 \\ -\beta_1 \end{bmatrix} \tag{4.8}$$

where β_p is defined to be the perpendicular of the projection of β to the spin plane. Again from Figure 4.1, it is seen that:

$$\sin(\psi) = \frac{\|\beta_p\|}{\|\beta\|} \tag{4.9}$$

and since the direction of τ_{m2} is fixed to be along β_p, the vector equation (4.8) can be solved for the scalar m_2:

$$m_2 = \frac{1}{\sin(\psi)}\left(\frac{\|\tau_{m2}\|}{\|\beta\|}\right) \tag{4.10}$$

and it is seen that the "efficiency" of the partial authority mode scales with the cosecant of Ψ. Note that $\|\beta\|$ appears in (4.10) and not in (4.7). This is expected because (4.10) is an expression of the coil command magnitude necessary to achieve τ_{m2}, whereas (4.7) is an

expression of needed wheel torque. The analogous expressions to (4.10) for mode 1 can be derived by literally inverting (4.4) to solve for m, but the expressions are cumbersome do not yield additional insights.

Equations (4.7) and (4.10) quantify the penalty of being in mode 1 or 2 as a function of local field configuration. The dependence on the secant and cosecant of Ψ implies that a strong need to switch actuation modes only exists near the extremes of the ψ range, as shown in Figure 4.2. This is fortunate because even though the full authority mode yields greater disturbance rejection performance (see Chapter 5) than mode 2, control-law switching always excites undesirable transients. The philosophy, then, is to stay in the current mode as long as possible, until forced to switch by the field vector direction. This suggests a Schmidt-trigger type of switching law. The Schmidt trigger form of the switching law also has usual the advantage of preventing control mode "flutter" when Ψ is on the verge of a transition point. Good trigger points are found by simulation to be about when the penalty reaches a factor of 3, which from Figure 4.2 implies 20°/160° and 70°/110°.
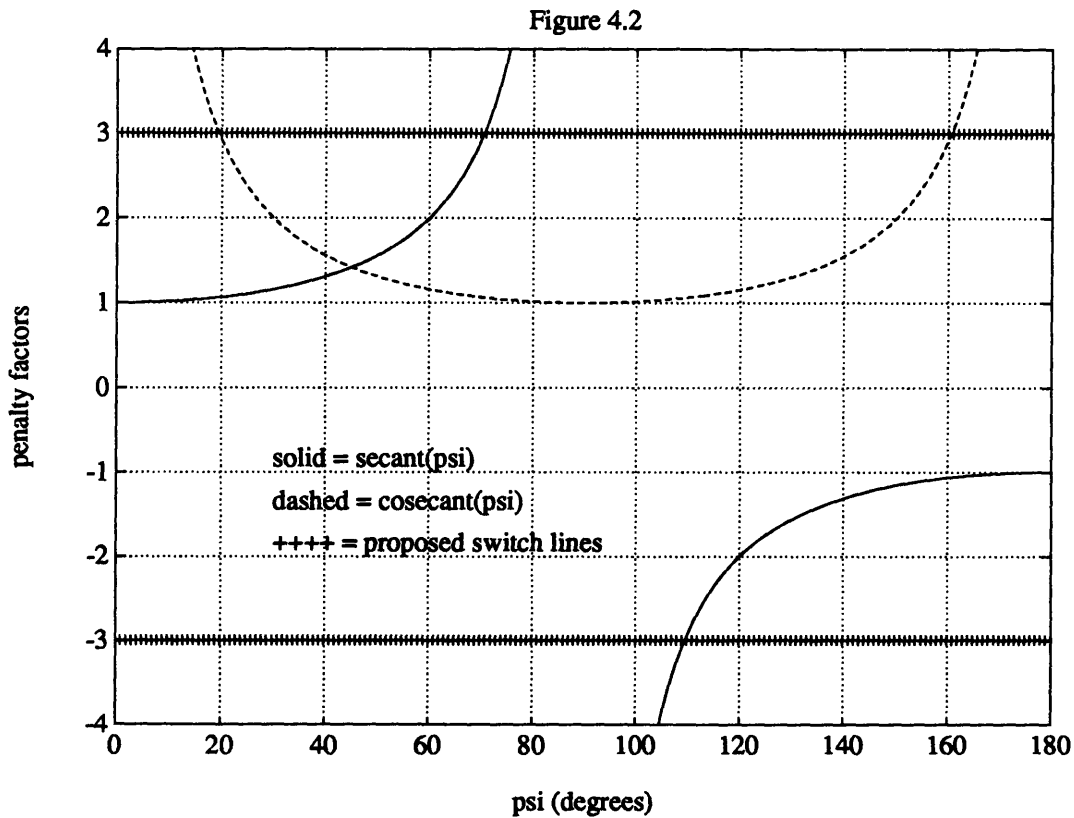
Figure 4.2

Figure 4.2: penalty factors for full authority (solid) and partial authority (dashed) actuator modes. The horizontal lines at ± 3 are the proposed switch points, when the penalty reaches a factor of 3.
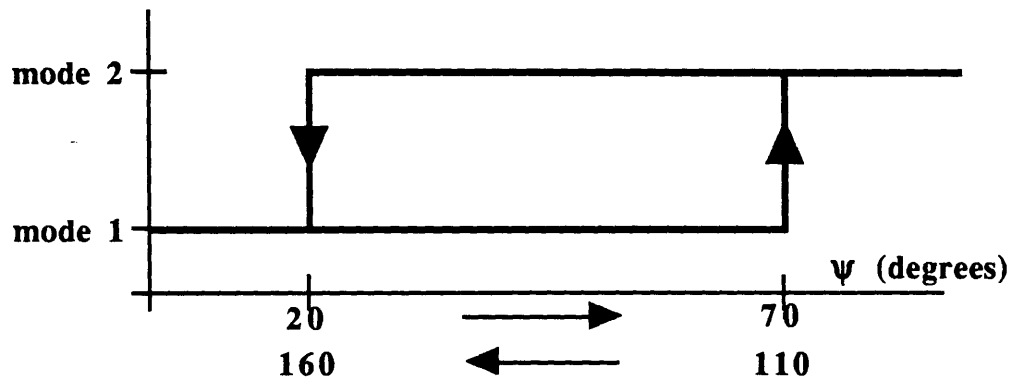
Figure 4.3: Schmidt trigger switching law to avoid changing control laws until necessary

In summary, the dual actuator mode strategy outlined compensates for local field direction changes by in effect emphasizing the spin plane coils to generate magnetic torque when Ψ is near 0, spin axis coil when Ψ is near 90°. The additional benefit of summing the wheel and mag torques when Ψ is near 0 is a bonus which is easily taken advantage of.

Figure 4.3 may suggest the implementation of a set of control laws gain-scheduled continuously to Ψ to achieve a "blending" of actuator modes 1 to 2, perhaps by "weighting" the coupling of the wheel to the magnets as a function of Ψ. This has the intuitive appeal of avoiding sudden mode switches. However, except in rare cases where control gains are nicely parametrized into continuous (or at least piecewise continuous) functions of one or two parameters, gain-scheduling implementation takes on the form of building large gain-tables as functions of the parameters which define the operating conditioning. Therefore, effort is spent minimizing the number of operating conditions necessary for which to design controllers to string together. Since it will be shown in Chapter 5 that, with intelligent use of strategies to reduce transients induced by the mode switch, two Ψ operating conditions suffice to meet specifications, we would in a circuitous fashion return to the currently proposed design. In short, the 2-actuator-mode control strategy proposed is justified on the basis of sufficiency.

4.1.4 Optimal Coil Commands for Momentum Removal

As mentioned, wheel desaturation occurs during orbit day and while employing the partial authority actuator mode. From (4.2), it is seen that the spin plane coils m_1 and m_3 generate torque according to:

$$\underline{\tau} = \begin{bmatrix} 0 \\ m_3\beta_1 - m_1\beta_3 \\ 0 \end{bmatrix} + \beta_2 \begin{bmatrix} -m_3 \\ 0 \\ m_1 \end{bmatrix} \equiv \underline{\tau}_{ds} + \underline{\tau}_d \qquad (4.11)$$

where the subscripts of the defined vectors $\underline{\tau}_{ds}$ and $\underline{\tau}_d$ suggest the view that $\underline{\tau}_{ds}$ is the desired desaturation torque and $\underline{\tau}_d$ is a disturbance in the spin plane caused by desaturation. The question of the optimal choice for the spin plane coil commands can then be posed as a linear Lagrange multiplier problem:

Suppose it is desired to generate some $\underline{\tau}_{ds} = [0 \ \tau_{ds} \ 0]^T$, then minimize:

$$\left\| \underline{\tau}_d \right\|^2 = [m_1 \ m_3] \begin{bmatrix} \beta_2^2 & 0 \\ 0 & \beta_2^2 \end{bmatrix} \begin{bmatrix} m_1 \\ m_3 \end{bmatrix} \qquad (4.12)$$

subject to the constraint:

$$[-\beta_3 \ \beta_1] \begin{bmatrix} m_1 \\ m_3 \end{bmatrix} = \tau_{ds} \qquad (4.13)$$

Imbedding the constraint results in the following matrix equation:

$$\begin{bmatrix} 2\beta_2^2 & 0 & -\beta_3 \\ 0 & 2\beta_2^2 & \beta_1 \\ -\beta_3 & \beta_1 & 0 \end{bmatrix} \begin{bmatrix} m_1 \\ m_3 \\ \lambda \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \tau_{ds} \end{bmatrix} \qquad (4.14)$$

whose solution for the coil commands is simply:

$$\begin{bmatrix} m_1 \\ m_3 \end{bmatrix} = \left(\frac{\tau_{ds}}{\beta_1^2 + \beta_3^2} \right) \begin{bmatrix} -\beta_3 \\ \beta_1 \end{bmatrix} \qquad (4.15)$$

Thus spin plane coil commands chosen according to (4.15) produce the spin axis torque τ_{ds} while minimizing the associated disturbing torque given a particular $\underline{\beta}$. The structure of the solution affords further insights, however. Substitution of (4.15) into (4.11) shows that the disturbance torque is:

$$\underline{\tau}_d = \beta_2 \left(\frac{\tau_{ds}}{\beta_1^2 + \beta_3^2} \right) \begin{bmatrix} -\beta_1 \\ 0 \\ -\beta_3 \end{bmatrix} \tag{4.16}$$

It is thus seen that $\underline{\tau}_d$ points <u>along</u> the projection of $\underline{\beta}$ on the spin plane, orthogonal to the direction which the spin-axis coil m_2 can generate torques.

We summarize by tracing out in figure 4.4 the relevant directions on the spin-plane B_1-B_3. Recall equation (4.2), repeated here for convenience:

$$\begin{bmatrix} \tau_{c1} \\ \tau_{c2} \\ \tau_{c3} \end{bmatrix} = \begin{bmatrix} -m_3\beta_2 \\ m_3\beta_1 - m_1\beta_3 \\ m_1\beta_2 \end{bmatrix} + m_2 \begin{bmatrix} \beta_3 \\ 0 \\ -\beta_1 \end{bmatrix} + \begin{bmatrix} 0 \\ \tau_w \\ 0 \end{bmatrix} \tag{4.2 repeat}$$
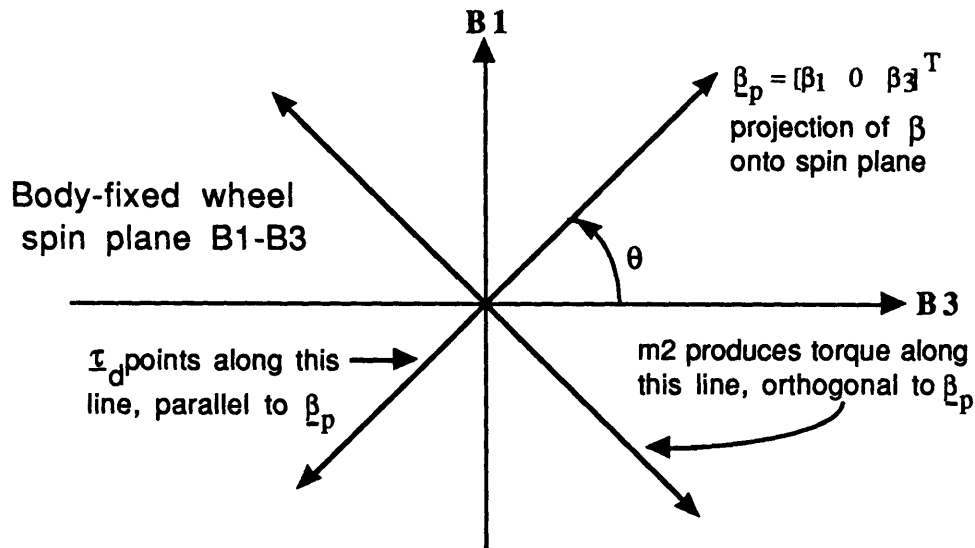
Figure 4.4: relationship of torques produced on the spin plane by the coils.

Since $\underline{\tau}_d$ is orthogonal to torques produced by m_2, the desaturation induced disturbance is necessarily uncontrollable in partial authority mode. The question is then raised: could we choose m_1 and m_3 such that the overall disturbance torque magnitude $||\underline{\tau}_d||$ is larger than that given by (4.16), but has a smaller component orthogonal to the torques produced by m_2? The idea, of course, it to then cancel the rest of the disturbance

torque with m_2 itself. This question can be posed as a modification to the Lagrange multiplier problem above, with (4.12) being changed to include the control action of m_2:

minimize the quadratic:
$$\left\| \underline{\tau}_d \right\|^2 = \left\| \beta_2 \begin{bmatrix} -m_3 \\ 0 \\ m_1 \end{bmatrix} + m_2 \begin{bmatrix} \beta_3 \\ 0 \\ -\beta_1 \end{bmatrix} \right\|^2$$

subject to the constraint:
$$\begin{bmatrix} -\beta_3 & \beta_1 \end{bmatrix} \begin{bmatrix} m_1 \\ m_3 \end{bmatrix} = \tau_{ds}$$

The algebra is more tedious, but the result is that the optimum m_2 is zero, and the optimal m_1, m_3 are as given in (4.15). The interpretation is that choosing coil commands other than according to equation (4.15) would produce a disturbance torque $\underline{\tau}_d$ which, even if its component orthogonal to $\underline{\beta}_p$ were completely cancelled by appropriate m_2 commands, would still have a left-over disturbance component no smaller than that given in (4.16). There is therefore no reason to choose any other desaturation coil commands.

It doesn't come as a surprise at this point that the desaturation efficiency of the spin plane coils is again scaled to a trignometric function of the angle ψ. The desaturation "efficiency" can be expressed as the ratio of disturbance to desaturation torque:

$$\frac{\left\| \underline{\tau}_d \right\|}{\left\| \underline{\tau}_{ds} \right\|}$$

with zero therefore being the optimum value. We derive expressions for both the numerator and the denominator. Recall from (4.11):

$$\underline{\tau}_d = \beta_2 \begin{bmatrix} -m_3 & 0 & m_1 \end{bmatrix}^T, \quad \left\| \underline{\tau}_d \right\| = \beta_2 \sqrt{m_1^2 + m_3^2} \qquad (4.17)$$

Recall also, by definition, $\left| \left| \underline{\tau}_{ds} \right| \right| = \tau_{ds}$. Equation (4.13) expresses the scalar τ_{ds} in the form of a vector dot product. But since (4.15) states that m_1 and m_3 are to be chosen such that:

$$\begin{bmatrix} -\beta_3 \\ \beta_1 \end{bmatrix} || \begin{bmatrix} m_1 \\ m_3 \end{bmatrix} \qquad (4.18)$$

the cosine of the included angle of this dot product is 1, and we can write:

$$\|\tau_{ds}\| = \sqrt{\beta_1^2 + \beta_3^2}\,\sqrt{m_1^2 + m_3^2}$$ (4.19)

Consideration of Figure 4.1 then produces the desired form:

$$\frac{\|\tau_d\|}{\|\tau_{ds}\|} = \frac{\beta_2}{\sqrt{\beta_1^2 + \beta_3^2}} = \frac{1}{\tan(\psi)}$$ (4.20)

Thus the efficiency of the desaturation coil commands scale with with the co-tangent of ψ, with ψ = 90° being optimal. As seen in figure 4.5, there is again a fortunately wide range of ψ when the penalty is acceptable. The angle ψ is monitored on-line to limit the control authority of the desaturation loop when necessary, as detailed in Chapter 5.
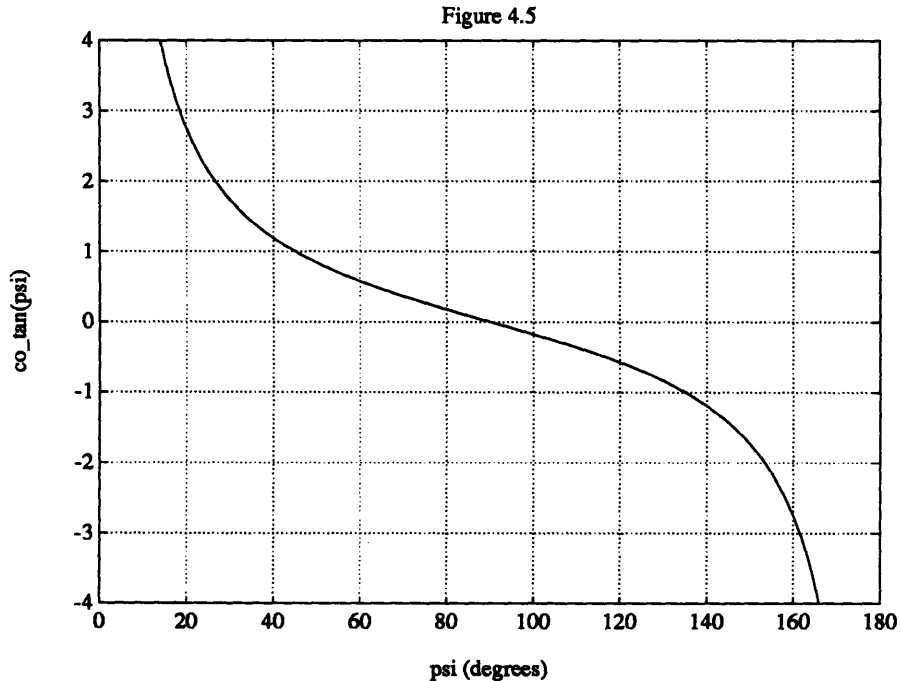
Figure 4.5: Efficiency of spin-plane coils at generating desaturation torques according to (4.15) and as a function of ψ.

4.2 Kinematics and Other Sensor Issues

Before the derivation of equations of motion, kinematics must be addressed. There are two issues to consider: the choice of the kinematic variables themselves, and the choice of the kinematic reference frame from which they describe the attitude of the body frame at the current time. It is seen that the latter is different for the orbit day and night phases.

Errors in the measurement of the angle θ, as defined in the previous section, will be important in the design of the control loop in Chapter 5. Bounding this error as a function of magnetometer bias is discussed in this section as well.

Quaternions (also known as Euler Parameters) are chosen as the kinematic parameters. While control law design in Chapter 5 uses small angle approximations where little difference exists between the families of kinematic variables (eg. Euler angles etc.), the slew maneuver makes it desirable to use a variable set without singularities at particular orientations. Without duplicating the extensive coverage in [Hugh86], a brief sketch of quaternion properties follows:

Reference frame: $\mathcal{F}_A$
Current Frame: $\mathcal{F}_B$
Euler axis/angle in $\mathcal{F}_A$ and $\mathcal{F}_B$: $\mathbf{a}, \phi$

<u>Definition:</u> $[\varepsilon_1\ \varepsilon_2\ \varepsilon_3]^T = \mathbf{a}\sin(\phi/2)$ $\eta = \cos(\phi/2)$ (4.21)

<u>Direction Cosines Relations:</u>

$$
C_{BA} = \begin{bmatrix} 1-2\left(\varepsilon_2^2+\varepsilon_3^2\right) & 2(\varepsilon_1\varepsilon_2+\varepsilon_3\eta) & 2(\varepsilon_1\varepsilon_3-\varepsilon_2\eta) \\ 2(\varepsilon_1\varepsilon_2-\varepsilon_3\eta) & 1-2\left(\varepsilon_1^2+\varepsilon_3^2\right) & 2(\varepsilon_2\varepsilon_3+\varepsilon_1\eta) \\ 2(\varepsilon_1\varepsilon_3+\varepsilon_2\eta) & 2(\varepsilon_3\varepsilon_2-\varepsilon_1\eta) & 1-2\left(\varepsilon_2^2+\varepsilon_1^2\right) \end{bmatrix}
$$

$$
\cong \begin{bmatrix} 1 & 2\varepsilon_3 & -2\varepsilon_2 \\ -2\varepsilon_3 & 1 & 2\varepsilon_1 \\ 2\varepsilon_2 & -2\varepsilon_1 & 1 \end{bmatrix} \text{ for infinetesmal rotation}
$$

(4.21)

$$\begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \end{bmatrix} = \frac{1}{4\eta} \begin{bmatrix} c_{23}-c_{32} \\ c_{31}-c_{13} \\ c_{12}-c_{21} \end{bmatrix} \qquad \eta = \pm\frac{1}{2}\sqrt{1+\text{trace}(C_{BA})} \qquad (4.22)$$

Angular Velocity Differential Relations:

$$\frac{d}{dt}\begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} \eta & -\varepsilon_3 & \varepsilon_2 \\ \varepsilon_3 & \eta & -\varepsilon_1 \\ -\varepsilon_2 & \varepsilon_1 & \eta \end{bmatrix}\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} \cong \frac{1}{2}\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} \text{ for infinitesimal } \varepsilon \qquad (4.23)$$

$$\frac{d(\eta)}{dt} = -\frac{1}{2}\begin{bmatrix} \varepsilon_1 & \varepsilon_2 & \varepsilon_3 \end{bmatrix}\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix}$$

$$\cong 0 \text{ for infinitesmal } \varepsilon \ (\therefore \eta \equiv 1) \qquad (4.24)$$

4.2.1 Orbit Night Kinematics

Since the attitude instrument for the orbit night is the CCD UV camera (more specifically, the star ID algorithm processing the CCD data), and since neither the configuration nor the software for the instrument has been finalized, little is said here about orbit night kinematics. It is simply assumed in this thesis that full attitude determination is attainable and that the information is available in quaternion form for feedback to the control loop.

The choice of the kinematic reference frame from which to measure displacement can be decided, however. Although it is quite possible to determine the direction of the sun-vector with probably greater precision than with the sun sensors, it should be kept in mind that the primary purpose of the orbit night attitude control loop is to minimize attitude rate in the presence of disturbances. As such, imposing a tracking requirement during orbit night works against this goal. Therefore, the kinematic reference frame during orbit night (null kinematic variable attitude) should simply be the attitude of the spacecraft at the time the orbit night controller first turns on. This of course does not imply that constant tracking of the attitude with respect to the sun-vector and a desired roll angle about it isn't useful- on the contrary, it is the only independent measurement of that roll angle available. The suggestion is simply that this information should not be fed back to the control law.

4.2.2 Sun Vector Tracking and Slew Kinematics

More can be said about kinematics during orbit day, since attitude control concerns drive the configuration of the sun-sensor constellation, unlike the case with the CCD camera. Since a slew maneuver and subsequent nutation damping and control are to be performed during orbit day, a general formulation for the quaternions as functions of the sun-sensor outputs and the desired sun-vector azimuth/elevation is presented here. It is noted that only two linearly independent expressions for the 3 quaternions (we derive a small angle formulation with $\eta = 1$) are available from one sun sensor.

Nominally, the goal is to align $\mathbf{B_3}$ to the sun-vector. After a slew, however, the goal is to align $\mathbf{B_3}$ to some vector offset from the sun vector. Differently oriented primary sun-sensors are likely to be used before and after a slew. These issues are formalized by the introduction of the following reference frames, in additional to the body fixed frame $\mathcal{F}_\mathbf{B}$:

$\mathcal{F}_\mathbf{I}$: inertially fixed (relative to vehicle dynamics time scale) frame centered at origin of $\mathcal{F}_\mathbf{B}$, $\mathbf{I_3}$ = sun vector

$\mathcal{F}_\mathbf{T}$: inertially fixed "track" frame- the goal of the control loop is to make $\mathcal{F}_\mathbf{B} = \mathcal{F}_\mathbf{T}$. Will define $\mathcal{F}_\mathbf{T}$ in terms of $\mathcal{F}_\mathbf{I}$ and $\mathcal{F}_\mathbf{B}$

$\mathcal{F}_\mathbf{Q}$: sun-sensor frame ("Q" for quad cell, since "S" is taken)

$\mathcal{F}_\mathbf{Q}$ is defined as in Section 2.2.3, and the sun-sensor outputs γ_1 and γ_2 are defined in Figure 4.6. Note that γ_1 and γ_2 are simply the $\mathbf{Q_1}$ - $\mathbf{Q_2}$ plane components of the sun-vector $\mathbf{I_3}$ in $\mathcal{F}_\mathbf{Q}$, since $\mathbf{I_3}$ is unit-length. As such, γ_1 and γ_2 match s_1 and s_2 in Section 2.2.3. Note also that γ_1 and γ_2 as defined below are not directly the measurements from the quad-cell current differences, but are processed outputs which make the formulations of this section easier, as mentioned in Section 2.2.3.

As in described in Chapter 2, we propose a constellation of sun sensors with their boresight axes on the spin plane, as shown in Figure 4.7. Let the boresight $\mathbf{Q_3}$ of a particular sun sensor be mounted at an azimuth angle a_q on the spin-plane with respect to $\mathbf{B_3}$. Also, to investigate the effect of sensor misalignment, let infinitesimal misalignment elevation and azimuth angles be δe_q and δa_q respectively. Then we can define the direction cosine matrix:

$$C_{QB} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & \delta e_q \\ 0 & -\delta e_q & 1 \end{bmatrix} \begin{bmatrix} \cos(a_q + \delta a_q) & 0 & -\sin(a_q + \delta a_q) \\ 0 & 1 & 0 \\ \sin(a_q + \delta a_q) & 0 & \cos(a_q + \delta a_q) \end{bmatrix} \tag{4.25}$$

where the infinitesimal form for the elevation misalignment rotation matrix is used.
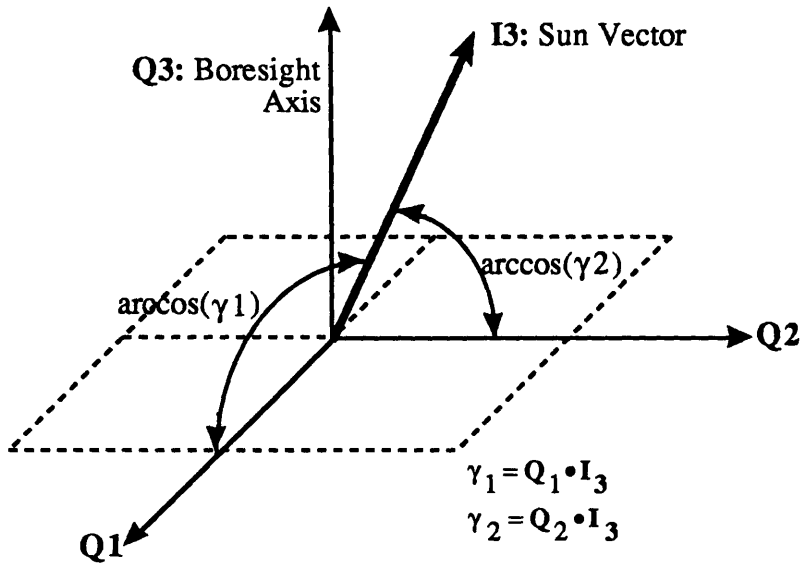
Figure 4.6: definition of the sun sensor frame and sensor outputs

$\gamma_1 = Q_1 \cdot I_3$

$\gamma_2 = Q_2 \cdot I_3$
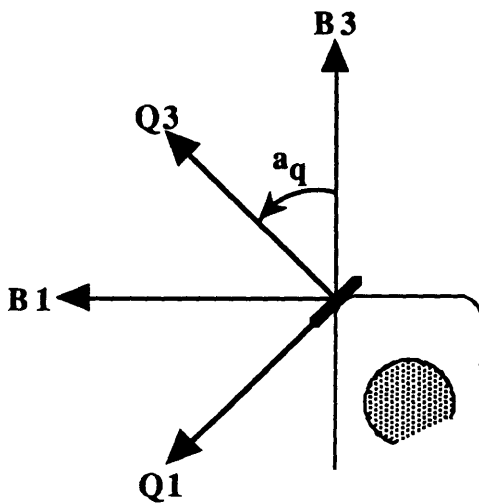
Figure 4.7: relation of body frame to sun sensor frame, where the sun-sensor is mounted at an azimuth angle aq. Case shown assumes no elevation misalignment of the boresight, which would be defined to obey the right hand rule about Q1, to be consistent with equation (4.25)

As mentioned, we now define a "tracking frame" $\mathcal{F}_T$ such that maintaining $\mathcal{F}_B$ = $\mathcal{F}_T$ fulfills the tracking requirement. Suppose there exists a desired sun vector I_3 located at azimuth a_0 and elevation e_0 with respect to $\mathcal{F}_B$ - thus we would like the actual sun vector to track I_3. Nominally both a_0 and e_0 would be zero, and even after a slew, the desired e_0 would still be 0. Since tracking a single vector specifies only 2 of 3 required degrees of freedom to uniquely define a reference frame, we choose a rotation sequence to locate $\mathcal{F}_I$ from $\mathcal{F}_T$ which makes it easy to discard the third degree of freedom. The transformation C_{IT} is essentially the Euler angle sequence 2-1-3 with the third angle set to zero:
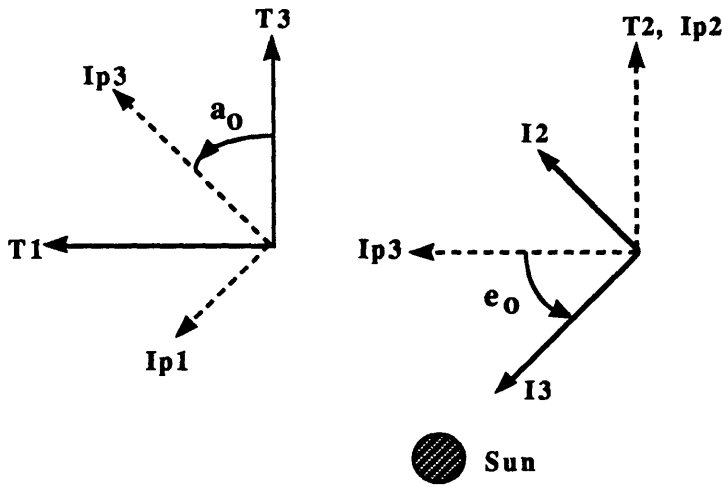
Figure 4.8: rotation sequence defining the sun (I) frame and track (T) frame. The sequence is shown from left to right, first by a0, then by e0. Note the intermediate frame (Ip) defined for clarity sake.

$$C_{IT} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(e_0) & \sin(e_0) \\ 0 & -\sin(e_0) & \cos(e_0) \end{bmatrix} \begin{bmatrix} \cos(a_0) & 0 & -\sin(a_0) \\ 0 & 1 & 0 \\ \sin(a_0) & 0 & (a_0) \end{bmatrix} \qquad (4.26)$$

$\mathcal{F}_T$ provides the kinematic reference frame to measure the attitude of $\mathcal{F}_B$ from. Therefore, C_{BT} is the same as the quaternion-direction cosine relation (4.21). Since controller design will employ a linearized model with $\mathcal{F}_B$ nearly = $\mathcal{F}_T$, the infinitesimal form is used:

$$C_{BT} = \begin{bmatrix} 1 & 2\varepsilon_3 & -2\varepsilon_2 \\ -2\varepsilon_3 & 1 & 2\varepsilon_1 \\ 2\varepsilon_2 & -2\varepsilon_1 & 1 \end{bmatrix} \qquad (4.27)$$

where the ε's now form the kinematic parameters (recall for infinitesimal attitude deviations $\eta = 1$). It then remains to simply write the definitions of sun sensor outputs γ_1 and γ_2 in scalar form:

$$\gamma_1 = Q_1 \bullet I_3 \qquad \gamma_2 = Q_2 \bullet I_3$$

$$\begin{bmatrix} \gamma_1 \\ \gamma_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} C_{QB} C_{BT} C_{TI} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \qquad (4.28)$$

The intermediate steps are algebra intensive and are left out. The results are:

$$\gamma_1 = -2\varepsilon_2 \cos(e_o)\left\{\cos(a_o)\cos(a_q + \delta a_q) + \sin(a_o)\sin(a_q + \delta a_q)\right\}$$

$$-2\sin(e_o)\left\{\varepsilon_1 \sin(a_q + \delta a_q) + \varepsilon_3 \cos(a_q + \delta a_q)\right\}$$

$$+\cos(e_o)\left\{\cos(a_q + \delta a_q)\sin(a_o) - \sin(a_q + \delta a_q)\cos(a_o)\right\} \qquad (4.29)$$

$$\gamma_2 = 2\varepsilon_1\left\{\cos(e_o)\cos(a_o) + \delta e_q \sin(e_o)\cos(a_q + \delta a_q)\right\}$$

$$-2\varepsilon_3\left\{\cos(e_o)\sin(a_o) + \delta e_q \sin(e_o)\sin(a_q + \delta a_q)\right\}$$

$$+2\varepsilon_2 \delta e_q \cos(e_o)\left\{\sin(a_o)\cos(a_q + \delta a_q) - \cos(a_o)\sin(a_q + \delta a_q)\right\}$$

$$+\delta e_q \cos(e_o)\left\{\sin(a_o)\sin(a_q + \delta a_q) + \cos(a_o)\cos(a_q + \delta a_q)\right\} - \sin(e_o) \qquad (4.30)$$

The results (4.29) and (4.30) are messy, and require some interpretation:

First of all, it is important to recall that a_o and e_o are defined as the <u>desired</u> sun-vector positions, which we are free to choose. The γ's are the sun-sensor measurements, and the ε's, as will be seen, form the state variables in the plant models used for controller design. Therefore, (4.29) and (4.30) are seen as the "output equations" (the "C" matrix) used in control design, and are general in a_o, e_o, and a_q.

For (4.29), note that the bias term disappears for $a_o = a_q + \delta a_q$. This corresponds tracking the sun-vector with the boresight of the sun-sensor at the appropriate azimuth $a_q + \delta a_q$. It is important to realize that the value of δa_q need not be known to "choose" $a_o = a_q + \delta a_q$. We simply design a controller to null the form of (4.29) which assumes $a_o =$

$a_q+\delta a_q$, and the effect of the misalignment δa_q is simply to degrade the azimuth tracking a little. The rationale for the sun sensor layout proposed in Chapter 2, where a set of fine sun-sensors are ringed about the $+B_3$ semicircle to provide a discrete set of "notches" to slew to, is precisely to facilitate choosing $a_0 = a_q+\delta a_q$. Furthermore, since the desire sun-vector is always on the spin-plane, e_0 is zero, and (4.29) boils down to:

$$\gamma_1 = -2\varepsilon_2 \tag{4.31}$$

It is seen that with $a_0 = a_q+\delta a_q$ and $e_0 = 0$, equation (4.30) becomes:

$$\gamma_2 = 2\{\varepsilon_1\cos(a_0) - \varepsilon_3\sin(a_0)\} + \delta e_q \tag{4.32}$$

The effect of the sun-sensor elevation misalignment δe_q is simply a bias on the output γ_2. Therefore, like δa_q, the effect of δe_q is to compromise elevation tracking slightly. It is significant to note that neither δe_q or δa_q causes any coupling of γ_1 to γ_2 or other signal degradation, provided $a_0 = a_q+\delta a_q$ and $e_0 = 0$. Since we do not know the value of δe_q, the form of (4.32) assuming $\delta e_q = 0$ is used in control design, with the knowledge that elevation tracking will be at best accurate to within δe_q- not a big deal.

That γ_2 is a linear combination of ε_1 and ε_3 for a finite azimuth a_0 is expected from the geometry of a slew maneuver, and consideration of the boundary cases $a_0 = 0$ or $90°$ provide sanity checks of the result.

4.2.3 Bounding θ Measurement Errors

It was shown in Section 4.1 that the angles θ and Ψ play important roles in the actuator strategy. Since the magnetometers which measure the local field are not perfect, errors in the calculated value of these angles are expected. It has been shown that Ψ is used primarily in open-loop fashion to determine actuator mode switching criteria and monitor the efficiency of momentum removal coil commands to limit the momentum management loop control authority. However, it will be shown in Chapter 5 that θ is a scheduling parameter for the attitude control loop, and that as such large enough errors in θ measurement can lead to closed-loop instability. Therefore, bounding the error on θ is addressed here.

The magnetometer output offset from the true field vector can be broadly split into unitary and non-unitary transformations. Unitary offsets are pure rotations of an

orthogonal magnetometer-fixed reference frame with respect to $\mathcal{F}_B$. Such a rotation can be characterized as an angle ϕ about an Euler axis $\underline{a}$. Worst case for θ then occurs when $\underline{a}$ = B_2, in which case all of ϕ translates to an error in θ. The worst case θ error from unitary offset is then easy to bound and, from [Pers91], estimated to be about 2 to 3 degrees.

Non-unitary offsets can be modelled as biases along each axis of $\mathcal{F}_B$ from the true field vector, which in lumped form represent individual magnetometer element biases and the result of non-orthogonal ("stretching") misalignments due to differences in scale factors along different axes. The θ measurement error due to these are more complicated, as it depends on the magnitude of the field vector being measured. We derive an expression for the worst case error here.

Projecting everything to the B_1-B_3 plane, we define:

actual field vector projection to spin-plane:

$$\underline{\beta}_p \equiv \begin{bmatrix} \beta_1 \\ \beta_3 \end{bmatrix}$$

lumped magnetometer bias vector on spin plane:

$$\underline{\delta} \equiv \begin{bmatrix} \delta_1 \\ \delta_3 \end{bmatrix}$$
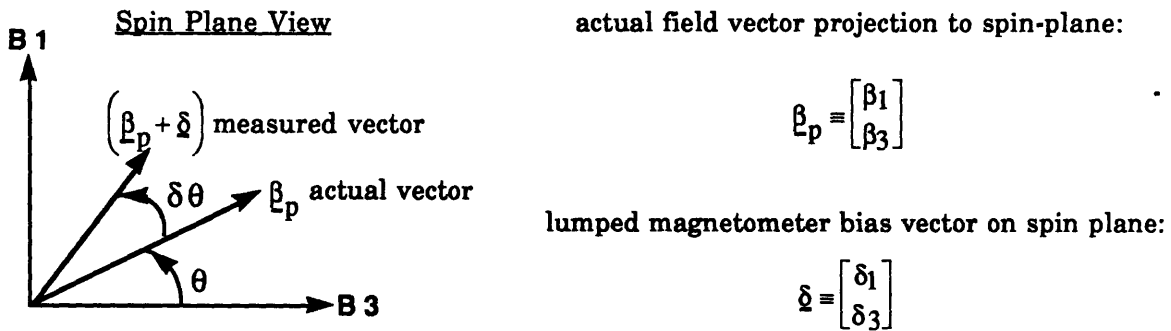
Figure 4.9: define quantities projected to the B_1 B_3 plane.

We don't claim to know the individual components of the bias vector $\underline{\delta}$, but assume that we can place a bound on the maximum magnitude $| \underline{\delta} |$. From 4.9, we can write:

$$\tan(\theta) = \frac{\beta_1}{\beta_3} \qquad \tan(\theta + \delta\theta) = \frac{\beta_1 + \delta_1}{\beta_3 + \delta_3} \tag{4.33}$$

From the trig identity:

$$\tan(\theta + \delta\theta) = \frac{\tan(\theta) + \tan(\delta\theta)}{1 - \tan(\theta)\tan(\delta\theta)} \tag{4.34}$$

we can solve for:

$$\tan(\delta\theta) = \frac{(\beta_3\delta_1 - \beta_1\delta_3)}{\beta_1^2 + \beta_3^2 + (\beta_1\delta_1 + \beta_3\delta_3)} \tag{4.35}$$

but note:

$$(\beta_3\delta_1 - \beta_1\delta_3) = \underline{\beta}_p \times \underline{\delta} = \left|\underline{\beta}_p\right|\left|\underline{\delta}\right|\sin(\lambda)$$

$$(\beta_1\delta_1 + \beta_3\delta_3) = \underline{\beta}_p \bullet \underline{\delta} = \left|\underline{\beta}_p\right|\left|\underline{\delta}\right|\cos(\lambda)$$

(4.36 a, b)

where λ is an (undetermined) separation angle between $\underline{\beta}$ and $\underline{\delta}$. Now define:

$$r \equiv \frac{\left|\underline{\beta}_p\right|}{\left|\underline{\delta}\right|}$$

(4.37)

and (4.35) can be re-written:

$$\tan(\delta\theta) = \frac{\sin(\lambda)}{r + \cos(\lambda)}$$

(4.38)

Recall the purpose of this calculation is to find $\max(\delta\theta)$. Noting that, for the interval $-90° < \delta\theta < +90°$, maximizing $\tan(\delta\theta)$ is the same as maximizing $\delta\theta$, we find stationary points of (4.38) with respect to λ:

$$\frac{d(\tan(\delta\theta))}{d\lambda} = \frac{(r + \cos[\lambda])\cos(\lambda) + \sin^2(\lambda)}{(r + \cos[\lambda])^2}$$

(4.39)

The stationary point is:

$$\cos(\lambda) = -\frac{1}{r}, \quad \sin(\lambda) = \sqrt{1 - \left(\frac{1}{r}\right)^2}$$

(4.40)

from which the desired expression is found by substitution into (4.38):

$$\delta\theta_{max} = a\tan\left\{\frac{\sqrt{1 - \left(\frac{1}{r}\right)^2}}{r - \frac{1}{r}}\right\} \qquad r \equiv \frac{\left|\underline{\beta}_p\right|}{\left|\underline{\delta}\right|}$$

(4.41)

Equation (4.41) is plotted in Figure 4.10, which gives a handle on the worst case θ error given the magnitude of the vector being measured and the expected bias magnitude.

From [Pers91] and [Lern79], a reasonably conservative estimate for $|\underline{\delta}|$ is $3\mu T$, which we shall use in subsequent discussion.

It is noted again that the ratio r is defined as $|\underline{\beta}_p|/|\underline{\delta}|$, which makes the effect of Ψ implicit, since $\underline{\beta}_p$ is the spin-plane projection of $\underline{\beta}$. Clearly, as Ψ approaches zero, r approaches zero as well.
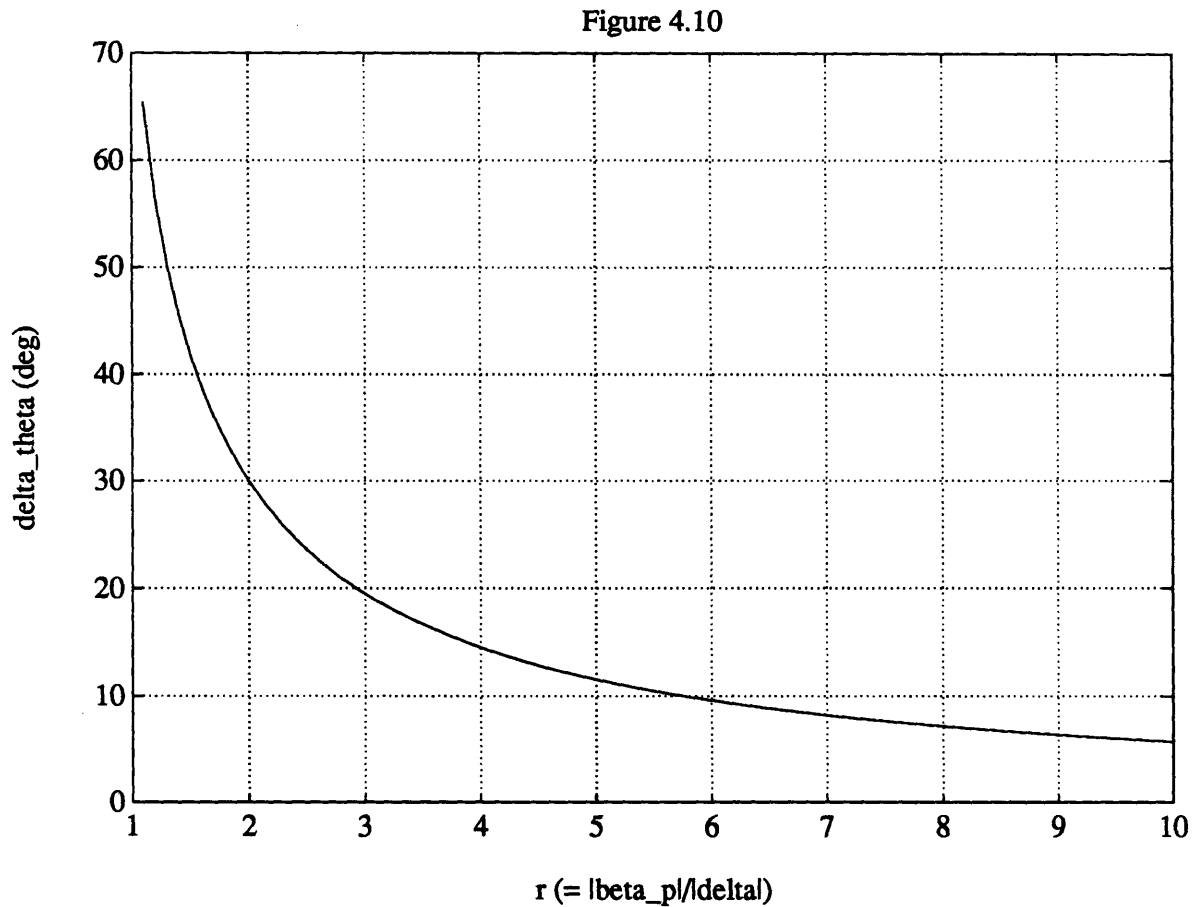
Figure 4.10

Figure 4.10: plot of equation (4.41). It is noted that for r<1, worst case $\delta\theta$ is $\pm 180°$, as is clear from geometric arguments.

4.3 Dynamics, Observability, Controllability

4.3.1 Assumptions & Mass Property Specifications

A rigid body model is assumed for the derivation of the equations of motion. This is justifiable in view of the low bandwidth of the controller (sampling frequency during orbit night is 0.25 Hz, during orbit day 1 Hz) relative to an estimated 31 Hz for the 1st flex mode of the solar panels [Pers91]. No passive nutation damper is assumed to exist. It is also assumed that fuel slosh is not a factor, either because the fuel was never on-board, or because it has been spent in an orbit raising maneuver. Estimates of the fundamental slosh frequency of low bond number fluids in an ellipsoidal tank are given in [Conc69].

We define a body fixed principal reference frame $\mathcal{F}_P$ such that P_1 is the principal axis closest to B_1, P_2 is the principal axis closest to B_2, and P_3 is the principal axis closest to B_3. Appendix A-3a gives code for the calculation of the principal inertias and the transformation C_{PB} from the inertia matrix expressed in $\mathcal{F}_B$. It is absolutely necessary in the construction of the spacecraft that $\mathcal{F}_B$ and $\mathcal{F}_P$ be closely aligned to minimize induced nutation during a slew maneuver, and this also allows decoupling of control of attitude about the wheel axis from the transverse attitude, as well as many other simplifications. Equations of motion are derived below for $\mathcal{F}_P \neq \mathcal{F}_B$ which are used in simulation, but the subset of them which assumes $\mathcal{F}_P = \mathcal{F}_B$ are used in control design for simplicity, and also because the anticipated difference between $\mathcal{F}_P$ and $\mathcal{F}_B$ after vehicle balancing will be in the region of measurement uncertainty, making it pointless to chase after.

However, we still wish to generate some worst case bound for a body-frame offset for use in simulations. To put a specification on the offset of $\mathcal{F}_P$ from $\mathcal{F}_B$, we note that C_{PB} is a rotation matrix, implying the existence of an Euler axis and angle of rotation. Dynamically the worst case rotation is one which perturbs P_2 from B_2 the most, implying that the Euler axis is somewhere on the B_1-B_3 plane. A convenient specification of the principal-body offset, then, is an angle ϕ about some random spin-plane unit vector. It is then fairly straightforward to generate, for a given ϕ, a worst case C_{PB} and body inertia matrix. [Hugh86] gives, assuming $\underline{a}$ is the Euler axis,

$$C_{BP} = \cos(\phi)I_3 + (1-\cos(\phi))\underline{a}\,\underline{a}^T - \sin(\phi)\underline{a}^\times \tag{4.42}$$

$$I_B = C_{BP}I_PC_{PB} \tag{4.43}$$

where: I_3 = 3x3 identity matrix

I_B = inertial matrix in $\mathcal{F}_B$

I_P = inertia matrix in $\mathcal{F}_P$

Appendix A-3b shows the subroutine for generating the worst case offset mass properties. For the mass properties given in Chapter 2, the following are the associated principal inertia properties. These represent the latest estimates and are used in subsequent numerical work.

$$I_P = \begin{bmatrix} 4.86 & 0 & 0 \\ 0 & 5.41 & 0 \\ 0 & 0 & 4.08 \end{bmatrix} \quad (Kg \cdot m^2)$$

4.3.2 Equations of Motion

Given all the assumptions above, the derivation is a straightforward application of Euler's equations. The transformations C_{PB} and C_{BP} are used liberally. Define some constants:

$$C_{PB} \equiv \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix}$$

I_1, I_2, I_3 = principal moments of inertia of spacecraft + wheel system

I_w = axial moment of inertia of wheel

Ω_0 = nominal wheel spin rate

$\delta\Omega$ = deviation from Ω_0

Then the angular momentum of the body+wheel system can be written in $\mathcal{F}_P$ as:

$$\underline{H} = \begin{bmatrix} I_1 & & \\ & I_2 & \\ & & I_3 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} + C_{PB} \begin{bmatrix} 0 \\ I_w(\Omega_0 + \delta\Omega) \\ 0 \end{bmatrix} \qquad (4.44)$$

Where $[\omega_1 \; \omega_2 \; \omega_3]^T$ is the spacecraft angular velocity vector in $\mathcal{F}_P$. Application of Euler's equation (4.45) then yields the motion equations (4.46):

$$\frac{d(\underline{H})}{dt} + \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \underline{H} = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} \tag{4.45}$$

$$I_1\dot{\omega}_1 + \omega_2\omega_3(I_3 - I_2) + h_w(c_{32}\omega_2 - c_{22}\omega_3) = \tau_1 \tag{4.46a}$$

$$I_2\dot{\omega}_2 + \omega_1\omega_3(I_1 - I_3) + h_w(c_{12}\omega_3 - c_{32}\omega_1) = \tau_2 \tag{4.46b}$$

$$I_3\dot{\omega}_3 + \omega_1\omega_2(I_2 - I_1) + h_w(c_{22}\omega_1 - c_{12}\omega_2) = \tau_3 \tag{4.46c}$$

$$\text{where:} \quad h_w \equiv I_w(\Omega_0 + \delta\Omega)$$

which describes a configuration termed the "Kelvin gyrostat" in [Hugh86]. It is noted that the external torque vector above is the lumped result of the control torques (as per equation (4.2)) and disturbance torques (Chapter 4):

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = C_{PB}\left\{ \begin{bmatrix} -m_3\beta_2 \\ m_3\beta_1 - m_1\beta_3 \\ m_1\beta_2 \end{bmatrix} + m_2 \begin{bmatrix} \beta_3 \\ 0 \\ -\beta_1 \end{bmatrix} + \begin{bmatrix} 0 \\ I_w\dfrac{d(\delta\Omega)}{dt} \\ 0 \end{bmatrix} \right\} + \text{disturbances} \tag{4.47}$$

Integration of attitude kinematics is done via (4.23), with the subtlety that it is the attitude of $\mathcal{F}_B$ which we are interested in tracking, so that in simulation it is $C_{BP}(\underline{\omega})$ which is integrated forward in time according to (4.23).

Linearization of (4.46) is straightforward. The equilibrium condition is assumed to be $\delta\Omega = 0$, Ω_0 constant, $\underline{\omega}_{ref} = 0$ (thus $\omega_n\omega_m$ terms dropped). The linearized equations in state variable form, with the infinitesimal form of kinematic relations (4.23) appended, is written:

$$\frac{d}{dt}\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \end{bmatrix} = \left[\begin{array}{ccc|c} 0 & \dfrac{-c_{32}h_w}{I_1} & \dfrac{+c_{22}h_w}{I_1} & \\ \dfrac{+c_{32}h_w}{I_2} & 0 & \dfrac{-c_{12}h_w}{I_2} & 0_{3\times3} \\ \dfrac{-c_{22}h_w}{I_3} & \dfrac{+c_{12}h_w}{I_3} & 0 & \\ \hline & \frac{1}{2}C_{BP} & & 0_{3\times3} \end{array} \right] \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \end{bmatrix} + \left[\begin{array}{ccc} \dfrac{1}{I_1} & 0 & 0 \\ 0 & \dfrac{1}{I_2} & 0 \\ 0 & 0 & \dfrac{1}{I_3} \\ \hline & 0_{3\times3} & \end{array} \right] \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} \tag{4.48}$$

As mentioned, the simplified form of (4.48) assuming $C_{PB} = C_{BP} = I_{3x3}$ is used as the design plant model in control design.

Small displacement natural motion can be investigated using the linearized equations as well. The characteristic equation of the A matrix in (4.48) is:

$$s^4\left(s^2 + \left(\frac{c_{12}^2}{I_2 I_3} + \frac{c_{22}^2}{I_1 I_3} + \frac{c_{32}^2}{I_1 I_2}\right)h_w^2\right) = 0 \qquad (4.49)$$

which upon numerical examination of the eigenstructure of A with the off-diagonal elements of C_{PB} small but $\neq 0$ reveals the natural motion to be:

s = 0,0,0   poles with eigenvectors pointing at ε_1 ε_2 ε_3 because of equation (4.23)

s = 0   pole with eigenvector pointing mostly at ω_2 .
                (almost double integrator dynamics along P_2 axis)

$$s = \pm j\sqrt{\left(\frac{c_{12}^2}{I_2 I_3} + \frac{c_{22}^2}{I_1 I_3} + \frac{c_{32}^2}{I_1 I_2}\right)h_w^2}$$

nutation poles with eigenvectors asymmetrically distributed mostly between (ε_1, ε_3, ω_1, ω_3), indicating elliptical nutation motion (in agreement with [Hugh86]).

The conflicting requirements driving the selection of the nominal wheel speed are the desire to have as much stiffness as possible versus power consumption and, more importantly, the ability to actively damp nutation. The last requires that the nutation frequency be within the control bandwidth, which is in turn limited by the CCD camera sampling rate (0.25 Hz). Thus Figure 4.11 plots the nutation pole solutions of (4.49) assuming the mass properties given. As discussed in Chapter 2, the configuration evaluated uses an Ithaco Scanwheel as the design momentum wheel, whose axial inertia was given in Chapter 2. We select Ω_0 = 3000 RPM with I_w = 6.3x10^-3 Kg•m² as the value of h_w considered in subsequent numerical work. The choice will be justified in light of various issues in subsequent sections and chapters.

Figure 4.11

wheel RPM (assuming Iw = 6.3e-3 Kg*m^2)
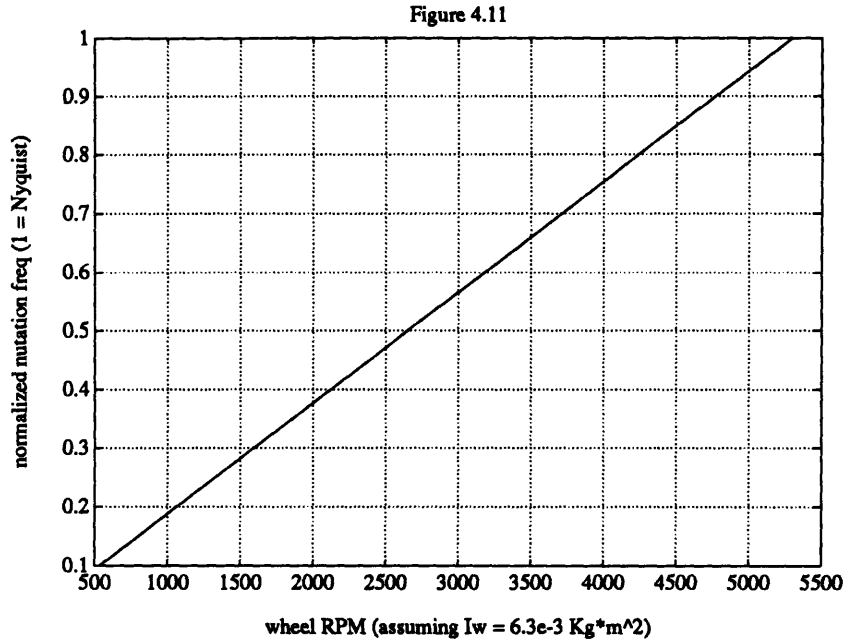
Figure 4.11: nutation frequency as a function of wheel speed, frequency normalized according to a 1/8 Hz Nyquist rate (1/4 Hz loop sample rate).

4.3.3 Observability and Controllability

The observability and controllability of the natural modes of (4.48) can be demonstrated numerically or proven algebraically by examining the eigenstructure of the state matrix as well as the measurement and control distribution matrices implied by the sensor kinematics models and actuator strategy definitions. However, that approach is cumbersome and not worth the effort here, as the dynamics are simple enough to argue the following on physical grounds alone.

Attitude observability with the CCD camera is complete. With the sun sensor, the angle about the boresight is unobservable. In full authority actuator mode (mode 1), all three attitude parameters are independently controllable. In partial authority mode, angle about the spin axis (infinitesimally, ε_2) is arbitrarily controllable with the wheel, while consideration of figure 4.4 and remembering the existence of gyroscopic coupling shows that the linear combination of ε_1 and ε_3 which corresponds to rotation about β_p is controllable.

Rate observability with the CCD camera is again complete on kinematic grounds alone. With the sun sensors, (4.29) shows that ω_2 is always observable through ε_2, while gyroscopic coupling renders ω_1 and ω_3 observable if any linear combination of ε_1 and ε_3 is observable, which is always the case with the sun sensors, as derived in (4.30). As for controllability, ω_2 is arbitrarily controllable with the wheel, while ω_1 and ω_3 are controllable (though not necessarily independently) whenever a torque can be generated on the spin plane, again due to gyroscopic coupling.

The two significant conclusions reached are:

Nutation damping (ω_1 and ω_3 observable and controllable) is always possible except if we choose to remain in actuator mode 2 as ψ approaches 0 or 180°, which we do during orbit day. This will be discussed further in Chapter 5.

While using a sun-sensor, a loop to control rotation about the boresight axis cannot be closed due to observability considerations. However, when the boresight azimuth equals θ, the only rotation achievable using the spin-axis coil is precisely about the boresight axis. We therefore expect a loss of tracking capability (i.e. a pole-zero cancellation- see equation (5.57)) when $\theta = a_0$. This will be explored further in Chapter 5 as well.

4.3.4 Active and Passive Disturbance Rejection

Having established the observability and controllability properties of the proposed configuration, we argue on dynamical grounds for the suitability of the chosen actuator strategy to this mission. The steady state response of a momentum bias spacecraft to a disturbance torque in the spin plane is quickly analyzed[1]. As shown in Chapter 3, the environmental disturbance torques have spectral components two decades below the spacecraft nutation frequency selected. They can therefore be approximated as constants here. The (small) natural motion of the spacecraft consists of the spin axis nutating in an elliptical path about the body+wheel system angular momentum vector H, which is dominantly the same as the wheel angular momentum vector h_w. From the definition of $\mathcal{F}_B$ it is seen that the motion of the spin axis relative to the inertial frame $\mathcal{F}_I$ is the vector

---

[1]Since the vehicle dynamics about the spin-axis is a double integrator if B_2 is close to P_2, an active regulator loop around the spin axis must be assumed to exist. Therefore we consider spin-plane disturbance only here.

$^I\omega^B$. Now if it is assumed that nutation is damped, then in the steady state the spin axis tracks the angular momentum vector. Then we can state:

$$\frac{^B d}{dt}H = 0 \qquad\qquad (4.50)$$

From elementary dynamics,

$$\frac{^I d}{dt}H = U = \frac{^B d}{dt}H + {}^I\omega^B \times H \qquad\qquad (4.51)$$

where the vector **U** is an applied torque in the spin-plane. Since it has been argued that $^I\omega^B$ tracks **H**, substitution of (4.50) into (4.51) and considering magnitudes yields:

$$\left|{}^I\omega^B\right| = \frac{|U|}{|H|} \qquad\qquad (4.52)$$

where of course $|H| = I_w\Omega_0$. Thus the passive rate disturbance rejection property of momentum bias stiffness is seen to be the attenuation of a spin-plane disturbance torque by a factor of $|H|$.

Since the control and disturbance signals in this case are the same quantities (torques), the action of a feedback regulator loop is in essense to deduce the disturbance torque vector from observation of the vehicle attitude and command control torques to cancel them, assuming that the desired state vector has been achieved. This is possible provided that:

- the disturbance spectra are within the control bandwidth
- the vector negative of the disturbance torque is achievable

In the full authority mode (mode 1), potentially both conditions are met, and we can therefore expect performance to be limited mostly by sensor/actuator imperfections. In the partial authority mode (mode 2), the second condition is not met, as control torques can only be generated along the spin axis and orthogonal to the projection of the field vector onto the spin plane. Thus the component of the disturbance torque along the projection of the field vector onto the spin plane is un-cancellable. However, the magnitude of this component is usually much smaller than the full disturbance vector itself, and it is the one

which enters (4.52). Thus we see what active control buys us- at worst, an augmentation of gyroscopic stiffness by reducing the magnitude of the torque challenging the stiffness[1].

The primary purpose of the attitude control system for this mission is the rejection of disturbance torques to meet the rate specifications. The shortcomings of simply using a high momentum bias alone to accomplish this are the difficulty of re-orienting the momentum vector to track the sun and set up a slew maneuver, and the increased power consumption. Furthermore, raising the nutation frequency past the control bandwidth would require the use of passive dampers, while space and weight (especially space) are at a premium. The opposite extreme of using a zero momentum bias suffers from the lack of appropriate actuators- namely commercially available reaction wheels of sufficiently small size. The use of magnetic torquers in a zero momentum bias situation alone is unacceptable due to the 1/s dynamics about the field vector which would be driven by the un-cancelled disturbance component. Thus the complementary use of magnetic torquing and momentum bias allows overcoming the disadvantages of using each individually, and makes achieving the specifications outlined in Chapter 1 plausible using proven off-the-shelf components.

---

[1]This will be seen numerically in Chapter 5. When the disturbance rejection frequency response of the closed loop system in the partial authority mode is presented as singular value plots, the plot of σ_{max} shows an attenuation by a factor of $1/(I_w\Omega_0) = -6dB$, with the associated input singular vector pointing along the direction of the field vector projection to the spin plane. Note the -6dB figure is of angular rate dimensions [rad/s], being the product of [torque]/[angular momentum].

V. Control Design for Observation Phase

This chapter details the design of controllers to meet specifications for the observation phase of the mission. Requirements for orbit night rate control, day sun tracking, momentum management, and slew about the spin axis[1] are met. Stability robustness, implementation issues, and performance evaluations are discussed, along with simulation results. The sensor/actuator imperfections outlined in Chapter 2 are accounted for in the designs and simulations; their specifications, which have been simply stated in Chapter 2, are justified here.

The following is a review of relevant attitude control specifications:

Orbit Night Phase: instruments require inertial drift rates below 2 °/hr about each
body axis (1 °/hr = $\pi/(3600 \cdot 180)$ rad/s); no strict pointing requirements;
loop sample rate is 0.25 Hz (to match CCD camera frame rate).

Orbit Day Phase: track B_3 to within 5° cone about sun-vector for solar arrays;
maintain inertial drift rates to below 20°/hr about each axis to permit CCD
camera image (and hence attitude) capture during transition to orbit night;
design loop sample rate is 1 Hz.

Slew Maneuver: minimize power consumption in maneuver; damp induced
nutation to orbit day specs within 5 minutes to allow transition to orbit night
(slews are to be performed near the end of orbit day to facilitate battery
recharging for the majority of the day).

As the desire for slewing about an arbitrarily oriented wheel spin axis arose only recently after a meeting with the science team, the orbit day controller presented in this chapter does not take into account of additional requirements imposed by the need to control or re-orient the spin axis roll angle about the sun vector to set up the slew maneuver. However, the ability to perform the slew maneuver itself is demonstrated. Changes and additional work necessary to implement spin-axis direction control are discussed in Chapter 6. Discussion of the initial attitude acquisition is also left to Chapter 6, primarily because it is an application of magnetic de-tumble and spin-up techniques known in the literature. The orbit raising phase, which at the time of writing stands a good chance of being eliminated as the result of choosing an expendable launch vehicle in place of the Shuttle, is not considered in this thesis.

---

[1]As in Chapter 4, "spin" is assumed to mean "wheel spin" unless otherwise noted. Thus the "spin axis" is
B_2 and the "spin plane" is the B_1-B_3 plane.

5.1 Design Plant Model

Compensators are designed with the assumption that $\mathcal{F}_B = \mathcal{F}_P$, which permits simplification of the design plant model (DPM) and explicit decoupling of the spin-axis controller from the spin-plane controller when it is desirable to do so. It is assumed that the spacecraft will be designed and balanced such that offsets of the principal frame from the body frame result from the limitations of mass properties measurements. As such, there is no point in attempting to use an unknown offset to gain a small advantage in control design. An assumed worst case offset, as quantified in (4.42) and (4.43), is used in simulations to evaluate controller performance.

To relate the actuator and sensor strategies outlined in the previous chapter to the DPM, recall the state equations (4.48), repeated below:

$$\dot{\underline{x}} = A\underline{x} + B\underline{u} \qquad\qquad y = C\underline{x} \tag{5.1}$$

$$\underline{x} = \begin{bmatrix} \omega_1 & \omega_2 & \omega_3 & \varepsilon_1 & \varepsilon_2 & \varepsilon_3 \end{bmatrix}^T \tag{5.2}$$

$$A = \left[\begin{array}{ccc|c} 0 & 0 & \dfrac{h_w}{I_1} & \\ 0 & 0 & 0 & 0_{3\times3} \\ \dfrac{-h_w}{I_3} & 0 & 0 & \\ \hline \dfrac{1}{2}I_{3\times3} & & & 0_{3\times3} \end{array}\right] \tag{5.3}$$

For the full authority actuator mode (mode 1), the control vector and control distribution matrix are:

$$B = \left[\begin{array}{ccc} \dfrac{1}{I_1} & 0 & 0 \\ 0 & \dfrac{1}{I_2} & 0 \\ 0 & 0 & \dfrac{1}{I_3} \\ \hline & 0_{3\times3} & \end{array}\right] \qquad \underline{u} = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} \tag{5.4}$$

where $\underline{u}$ is related to coil dipole and wheel acceleration by equation (4.4).

For the partial authority mode (mode 2), the wheel generates $[0 \ \tau_2 \ 0]^T$ independently, while the spin axis coil m_2 generates the transverse torque τ_t according to equation (4.2):

$$\tau_t = \begin{bmatrix} \beta_3 \\ 0 \\ -\beta_1 \end{bmatrix} m_2 \qquad (5.5)$$

The coil current's dependence on the field strength makes for an undesirable variance which is easily eliminated. Define a pseudo-control signal τ_{tn} to put (5.5) in the more elegant (and easier to schedule) form: (recall figure 4.1)

$$\tau_t = \begin{bmatrix} \dfrac{\beta_3}{\sqrt{\beta_1^2+\beta_3^2}} \\ 0 \\ \dfrac{-\beta_1}{\sqrt{\beta_1^2+\beta_3^2}} \end{bmatrix} m_2\sqrt{\beta_1^2+\beta_3^2} = \begin{bmatrix} \cos(\theta) \\ 0 \\ -\sin(\theta) \end{bmatrix} \tau_{tn} \qquad (5.6)$$

Therefore, in the partial authority actuator mode:

$$B = \begin{bmatrix} \dfrac{1}{I_1} & 0 & 0 \\ 0 & \dfrac{1}{I_2} & 0 \\ 0 & 0 & \dfrac{1}{I_3} \\ \hline & 0_{3\times3} & \end{bmatrix} \begin{bmatrix} \cos(\theta) & 0 \\ 0 & 1 \\ -\sin(\theta) & 0 \end{bmatrix} \qquad \underline{u} = \begin{bmatrix} \tau_{tn} \\ \tau_2 \end{bmatrix} \qquad (5.7)$$

As for the sensors, the CCD camera output is assumed to give a measurement vector and matrix of:

$$\underline{y} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \end{bmatrix} = C\underline{x} \qquad C = [0_{3\times3} \mid I_{3\times3}] \qquad (5.8)$$

Following the recommendations of section 4.2.2, the desired sun azimuth angle is assumed to equal the sun sensor boresight azimuth ($a_0 = a_q + \delta a_q$), while the desired sun elevation (with respect to the spin plane) is $e_0 = 0$. Further, assume the sensor elevation misalignment $\delta e_q = 0$, and that the negative sign in (4.29) is reversed in software for convenience. Defining the processed sun sensor outputs as:

Azimuth error $= \lambda_1$
Elevation error $= \lambda_2$

equations (4.29) and (4.30) yield for the day measurement matrix:

$$\underline{y} = \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} = C\underline{x} \quad C = \begin{bmatrix} 0_{2\times3} & \begin{vmatrix} 0 & 2 & 0 \\ 2\cos(a_0) & 0 & -2\sin(a_0) \end{vmatrix} \end{bmatrix} \tag{5.9}$$

Discrete time (DT) versions of the DPM are necessary for the implementation of DT controllers. The transformation is performed on the state and control distribution matrices using the standard zeroth order hold equivalence method, as detailed in [Fran80]. The transformation basically solves the continuous time equations for one sample period assuming a constant control signal, resulting in:

$$\underline{\dot{x}}(t) = A\underline{x}(t) + B\underline{u}(t) \quad \rightarrow \quad \underline{x}[n+1] = \Phi\underline{x}[n] + \Gamma\underline{u}[n] \tag{5.10}$$

$$\Phi = e^{AT} \qquad \Gamma = \left[\int_0^T e^{A\eta} d\eta \right] B \qquad T = \text{sampling period (sec)} \tag{5.11}$$

Note the equation for Γ has B as the post-multiplying matrix. This is convenient because, for actuator mode 2, the B matrix has the form given in (5.7). But the 3x2 matrix containing θ, which is time varying as the local field vector changes, need not be included in the transformation but can be multiplied with the Γ derived with B being that in (5.4), which contains only time-invariant elements. Thus (5.11), which is usually performed with a series approximation and is computationally intensive, need not be calculated on-line in either simulation or implementation of gain-scheduled compensation. The time-varying part can simply be multiplied on the tail.

5.2 Orbit Night Controller Design

5.2.1 Design for Full Actuator Authority Mode

This is the most trivial design case, where full attitude observability and arbitrary control torque generation are assumed. The only stringent constraints are that the loop sample period is 4 seconds long, with an assumed 1 second delay between the CCD camera capturing an image in its buffer and the output of new coil/wheel commands (note computation of the attitude from a portion of the image is assumed to take place in this interval). The delay is found in simulation to have negligible effect on the performance of the controller. Of the various possible compensation design methods, straight LQG is chosen primarily for the overall ease of design and discrete-time implementation. Since the LQR and estimator gain cost function minimizations are posed in discrete time, their solutions yield directly valid controllers at the design sample rate, assuming the design plant is satisfactorily discretized.

Numerical values of the DPM in continuous time and discrete time assuming a 4 sec. sample period are given below. The DT versions go into the implementation of the model based LQG compensator. The measurement matrix C is as given in (5.8).

$$
A =
$$

0	0	0.4073	0	0	0
0	0	0	0	0	0
-0.4854	0	0	0	0	0
0.5000	0	0	0	0	0
0	0.5000	0	0	0	0
0	0	0.5000	0	0	0

$$
B =
$$

0.2058	0	0
0	0.1847	0
0	0	0.2453
0	0	0
0	0	0
0	0	0

Φ =

-0.2062	0	0.8962	0	0	0
0	1.0000	0	0	0	0
-1.0683	0	-0.2062	0	0	0
1.1004	0	1.2424	1.0000	0	0
0	2.0000	0	0	1.0000	0
-1.4809	0	1.1004	0	0	1.0000

Γ =

0.4528	0	0.6094
0	0.7390	0
-0.6094	0	0.5398
0.6277	0	0.4545
0	0.7390	0
-0.4545	0	0.7482

Steady state solutions are found for the relevant Ricatti equations so as to implement time-invariant compensators. The regulator cost function minimized is (n is the discretized time variable; all functions in (5.12) are functions of n):

$$J = \frac{1}{2} \sum_{n=0}^{\infty} \left\{ \omega_1^2 + \omega_2^2 + \omega_3^2 + 10^{-2} \left(\varepsilon_1^2 + \varepsilon_2^2 + \varepsilon_3^2 \right) + \tau_1^2 + \tau_2^2 + \tau_3^2 \right\} \qquad (5.12)$$

It would be legitimate to question the non-zero weight on the epsilons, since the specifications do not call for tracking a particular attitude and, in fact, it has been argued in Section 4.2.1 that the attitude should simply be measured with respect to the reference frame which corresponds to the attitude of the spacecraft when the orbit night phase is initially entered. The answer is simply that rejection of disturbance torques on rate perturbations, the whole purpose of the controller, is greatly diminished if the regulator gain matrix ignores attitude perturbations (i.e. has zero elements in the corresponding entries, which would be the case it the epsilon weights were set to zero). This is demonstrated in Section 5.2.4 below.

The filter portion of the compensator is designed to have poles slightly faster than the regulator poles, and to weight measurements more heavily than propagated internal dynamics in the innovation. Continuing with the LQG design methodology, the filter gain is calculated as a steady state Kalman filter, with assumed values of process and measurement noise intensities to achieve the desired poles. Defining the white process noise to be w[n] and the white measurement noise to be v[n], the intensities chosen are:

$$E\left[w[n]w[n]^T\right] = I_{3\times3} \qquad E\left[v[n]v[n]^T\right] = \frac{1}{10}I_{3\times3} \qquad (5.13)$$

The process noise is assumed to be disturbance torques, so its distribution matrix is the same as the control distribution matrix. In discrete time, the filter is implemented in the form of a current estimator [Fran80]; that is, with separate estimation and prediction steps.

Numerical values for the resultant filter gain matrix H and regulator gain matrix G are given below:

```
H =

        0.5050    -0.0000     0.3319
        0.0000     0.5710     0.0000
       -0.3569     0.0000     0.5304
        0.9315    -0.0000     0.0021
        0.0000     0.9403    -0.0000
        0.0021    -0.0000     0.9425


G =

        0.5392    -0.0000     0.4837     0.0225    -0.0000    -0.0629
       -0.0000     0.8669    -0.0000    -0.0000     0.0638     0.0000
       -0.5482    -0.0000     0.4673     0.0601    -0.0000     0.0199
```

Note the positions of the zeros in both gain matrices indicate the complete decoupling of the spin axis loop from the spin plane loop, as expected. This decoupling is left implicit in the compensator implementation to streamline the form of presentation. It is also noted that since the controller estimates rates which it then feeds back with proportions in the G matrix, the LQG method amounts in this case to be a systematic way of designing lead compensators. The compensator implementation equations are summarized below:

- previous compensator state: $\bar{x}[n]$
- measurement vector: $\underline{c}[n]$
- error signal (compensator input):   e=r-$\underline{c}[n]$ (r = [0 0 0]^T)   (5.14)
- current estimate: $\hat{x}[n] = \bar{x}[n] - H(e + C\bar{x}[n])$ (5.15)
- compensator output: $\underline{u}[n] = -G\hat{x}$ (5.16)
- prediction step: $\bar{x}[n+1] = \Phi\hat{x}[n] + \Gamma\underline{u}[n]$ (5.17)

Matrix equation (4.4) is then inverted with the control output from (5.16) and the current magnetometer reading to form commands for the wheel torque and coil dipoles.

The separation principle applies, allowing the poles of the closed loop to be separated into regulator and filter loop poles:

<div align="center">

Filter Poles: eig(Φ-ΦHC)   (5.18)

Regulator Poles: eig(Φ-ΓG)   (5.19)

</div>

For the purpose of maximizing insight, these poles are mapped to the s-plane via $s = Log(z)/T$. Numerical values (rad/s) are given below:

```
        filter              regulator

    -0.4137 + 0.2879i   -0.1925 + 0.4452i
    -0.4137 - 0.2879i   -0.1925 - 0.4452i
    -0.2785 + 0.7415i   -0.0225 + 0.0010i
    -0.2785 - 0.7415i   -0.0225 - 0.0010i
    -0.3523 + 0.4350i   -0.1727
    -0.3523 - 0.4350i   -0.0523
```

MATLAB code to generate the design is included in Appendix A-5a for cross-checking.

5.2.2 Design for Partial Actuator Authority Mode

In mode 2, the spin axis coil and the wheel are commanded. The use of only the spin axis coil limits the directionality of the possible spin-plane torque, as accounted for in (5.7). The spin-plane torque is limited to lie along $\theta \pm 90°$ on the spin plane. A satisfactory time-invariant control law which is robust to arbitrary θ variations has not been found. Since θ changes slowly compared to the spacecraft dynamics (Chapter 3) and is measured with the magnetometers, the approach of using regulator gains scheduled to θ has been used with success.

It was found that the same filter loop- defined as the steady state LTI Kalman filter- with the assumed process and measurement noises covariances given in (5.13)- gives satisfactory performance. The H matrix is therefore not recomputed. The regulator loop is tuned to θ variations. The control vector and distribution matrices are as defined in (5.7) and repeated below for convenience:

$$B_0 \equiv \left[\begin{array}{ccc} \dfrac{1}{I_1} & 0 & 0 \\ 0 & \dfrac{1}{I_2} & 0 \\ 0 & 0 & \dfrac{1}{I_3} \\ \hline & 0_{3\times3} & \end{array}\right] \qquad B_{dir} \equiv \left[\begin{array}{cc} \cos(\theta) & 0 \\ 0 & 1 \\ -\sin(\theta) & 0 \end{array}\right] \qquad B = B_0 B_{dir} \qquad \underline{u} = \left[\begin{array}{c} \tau_{tn} \\ \tau_2 \end{array}\right] \qquad (5.20)$$

The time invariant portion of the DPM is discretized using (5.11); the time varying control distribution matrix is then constructed by multiplication, as mentioned. Therefore:

$$\Phi = e^{AT} \qquad \Gamma_0 = \left[\int_0^T e^{A\eta} d\eta\right] B_0 \qquad \Gamma(\theta) = \Gamma_0 B_{dir} \qquad (5.21)$$

and is is seen that Φ is unchanged and $\Gamma = \Gamma_0$ from the previous section.

The cost function (5.23) as applied to the DPM defined by (5.22) is then solved for many discrete θ's in the range $0 \le \theta \le 2\pi$ to generate a family of gain solutions $G(\theta)$:

$$\underline{x}[n+1] = \Phi\underline{x}[n] + \Gamma(\theta)\underline{u}[n] \qquad \underline{u} = \left[\begin{array}{c} \tau_{tn} \\ \tau_2 \end{array}\right] \qquad (5.22)$$

$$J = \frac{1}{2}\sum_{k=0}^{\infty}\left\{\omega_1^2 + \omega_2^2 + \omega_3^2 + 10^{-2}\left(\varepsilon_1^2 + \varepsilon_2^2 + \varepsilon_3^2\right) + \tau_{tn}^2 + \tau_2^2\right\} \qquad (5.23)$$

Note the weights for ω_2, ε_2 and τ_2 are the same as in (5.12) and, since the DPM does not change for the rows and columns related to ω_2 and ε_2, we expect the same gain elements for the wheel control loop. The compensator implementation equations are similar to before, with θ being computed on line from the magnetometers:

- previous compensator state: $\bar{\underline{x}}[n]$
- measurement vector: $\underline{s}[n]$
- error signal (compensator input): $e=r-\underline{s}[n]$ $\qquad (r = [0\ 0\ 0]^T)$ $\qquad (5.24)$
- current estimate: $\hat{\underline{x}}[n] = \bar{\underline{x}}[n] - H(e + C\bar{\underline{x}}[n])$ $\qquad (5.25)$
- compensator output: $\underline{u}[n] = -G(\theta)\hat{\underline{x}}$ $\qquad (5.26)$
- prediction step: $\bar{\underline{x}}[n+1] = \Phi\hat{\underline{x}}[n] + \Gamma_0 B_{dir}\underline{u}[n]$ $\qquad (5.27)$

We expand the control law (5.26) to define the non-zero elements of $G(\theta)$:

$$\tau_2 = g_{22}\omega_2 + g_{25}\varepsilon_2 \tag{5.28}$$

$$\tau_{tn} = g_{11}\omega_1 + g_{13}\omega_3 + g_{14}\varepsilon_1 + g_{16}\varepsilon_3 \tag{5.29}$$

where the ω's and ε's are from the estimator output $\hat{\underline{x}}$ as calculated in (5.25). The gain elements of (5.28) are constant and the same as in the actuator mode 1 controller. The gain elements of (5.29) depend on θ and are plotted in figures 5.1 and 5.2. Their clear sinusoidal shapes suggest that it would be quite easy to compute them on line as fitted functions of θ, with a minimum of storage required. For simulations in this thesis, the method adopted is a polynomial fit, since MATLAB provides a canned routine well suited to this purpose. The code to calculate and fit the gains is given in Appendix A-5a.

It is worthwhile to examine the eigenstructure of the closed loop system as functions of θ to further understand the controller design. Again, recall the filter closed loop poles do not change from those in mode 1 (equation 5.18 and subsequent numerical values), since H, C and Φ are all unchanged. The eigenstructure of the closed loop regulator matrix is examined numerically:

$$\text{eig}[\Phi - \Gamma(\theta)G(\theta)] \tag{5.30}$$

The spread of the eigenvalues (poles) over the full 2π θ range is plotted in figure 5.3, again mapped to the s-plane to facilitate insight. It is useful to recall the open loop poles of the DPM: 4 integrators and 2 nutation poles on the $j\omega$ axis. On figure 5.3, the poles labeled "wheel" have eigenvectors with non-zero elements only along ω_2 and ε_2, showing the expected decoupled behavior. They represent two of the four open loop integrators and are moved by the spin axis loop. The poles labeled "nutation" have complex conjugate eigenvectors with components along ω_1, ω_3, ε_1, and ε_3. They are the damped open loop nutation poles. Of the remaining two open loop integrators, one is moved to $\sim$ s = -0.02. The remaining integrator is not moved and has a closed loop eigenvector oriented at $\pm \cos(\theta)$ in the $\varepsilon1$ component, and $\pm \sin(\theta)$ in the $\varepsilon3$ component. This is the uncontrollable mode using the partial authority actuator, which has a clear geometric interpretation, as shown in Figure 5.4. Note that transmission zeros are not plotted in Figure 5.3, but that one certainly exists at s = 0.

It is significant that the uncontrollable mode contains no ω components. This is

consistent with the fact that gyroscopic stiffness prevents rates from having 1/s dynamics about any axis but the spin axis, which in the closed loop is clearly asymptotically stabilized. A disturbance torque component along the axis $\theta \pm 90°$ on the spin plane will produce growing perturbations in attitude and a steady state constant rate perturbation of magnitude attenuated by the angular momentum of the wheel according to equation (4.52). This is examined further in Section 5.2.4. Note, of course, that the analyses given here have assumed a constant θ, which is in fact changing slowly on this time scale as the spacecraft traverses its orbit. Thus the uncontrollable direction changes.
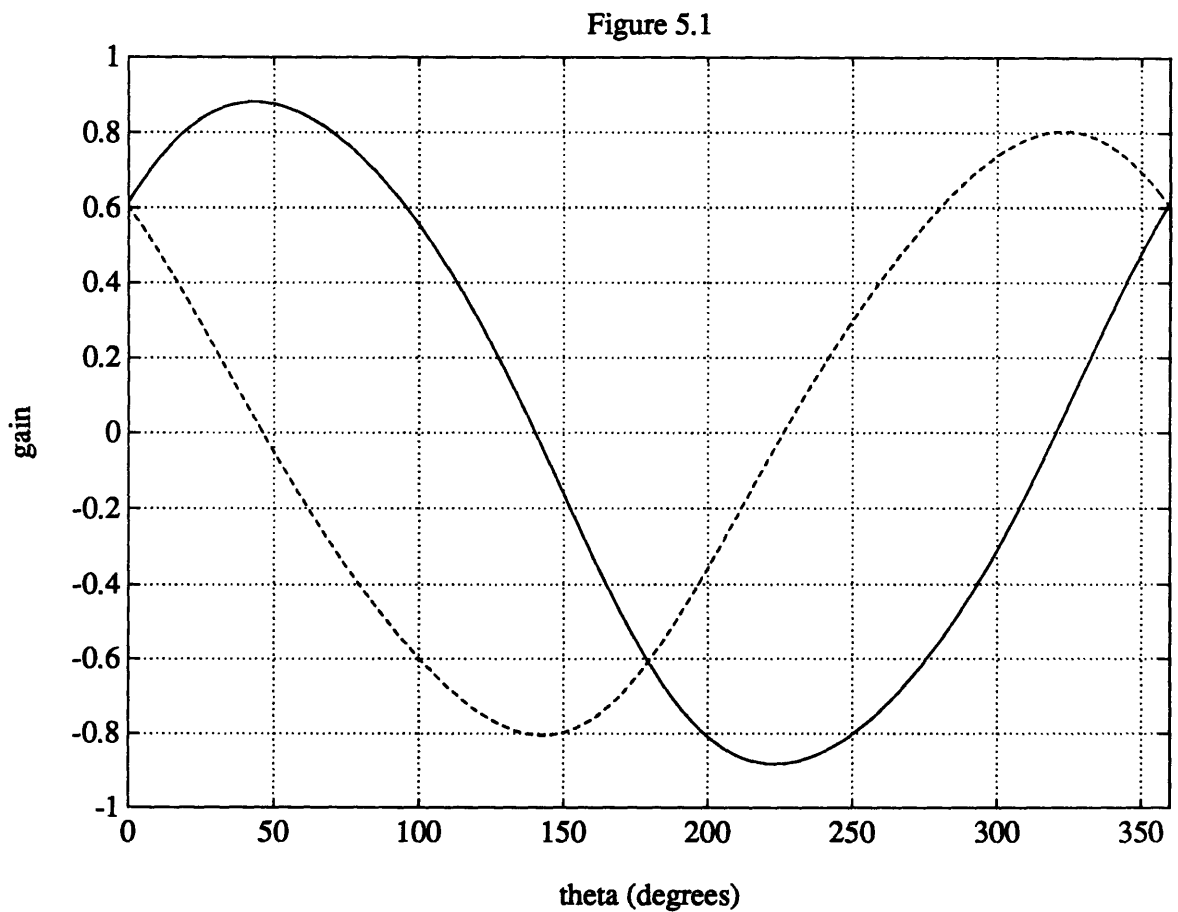
Figure 5.1

Figure 5.1: LQR solutions for the ω feedback gains g11 (solid) and g13 (dashed) as functions of θ.

Figure 5.2
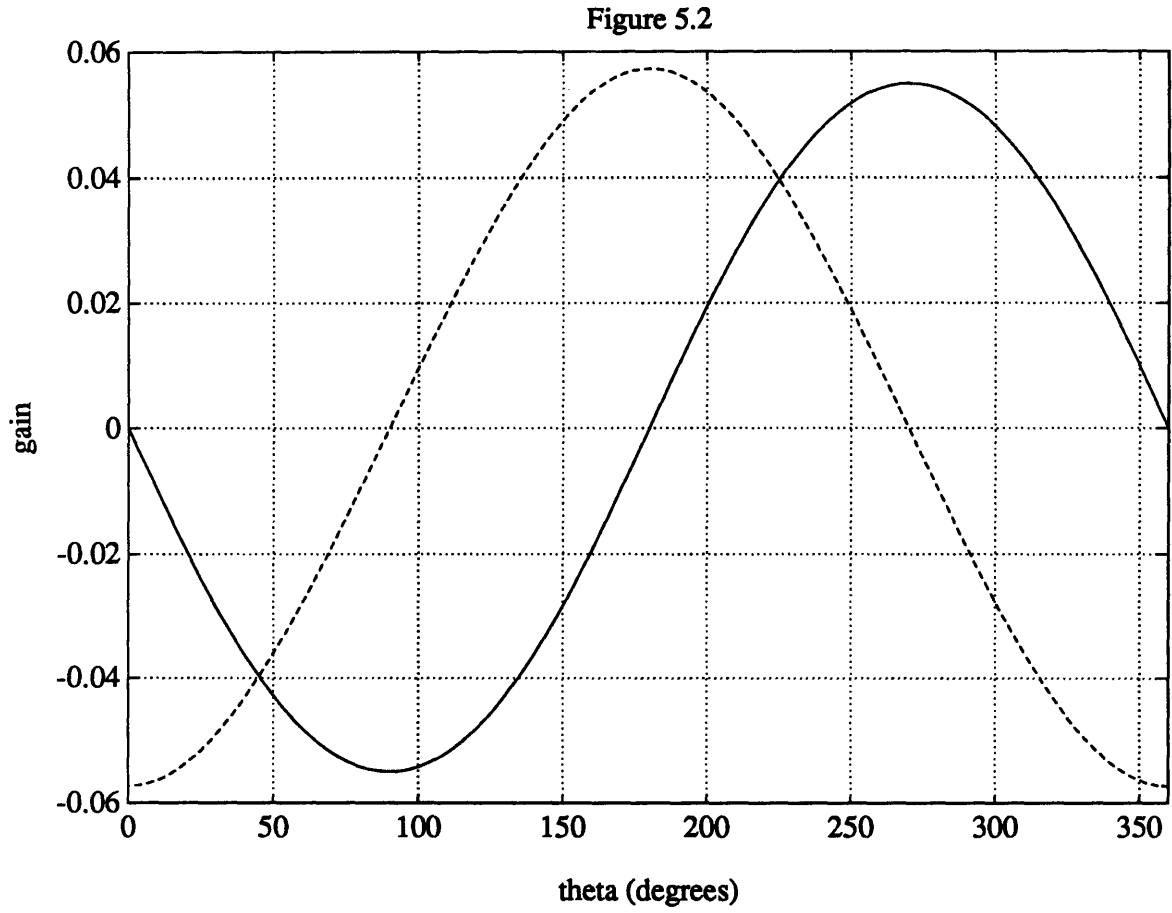

Figure 5.2: LQR solutions for the ε feedback gains g_{14} (solid) and g_{16} (dashed) as functions of θ.
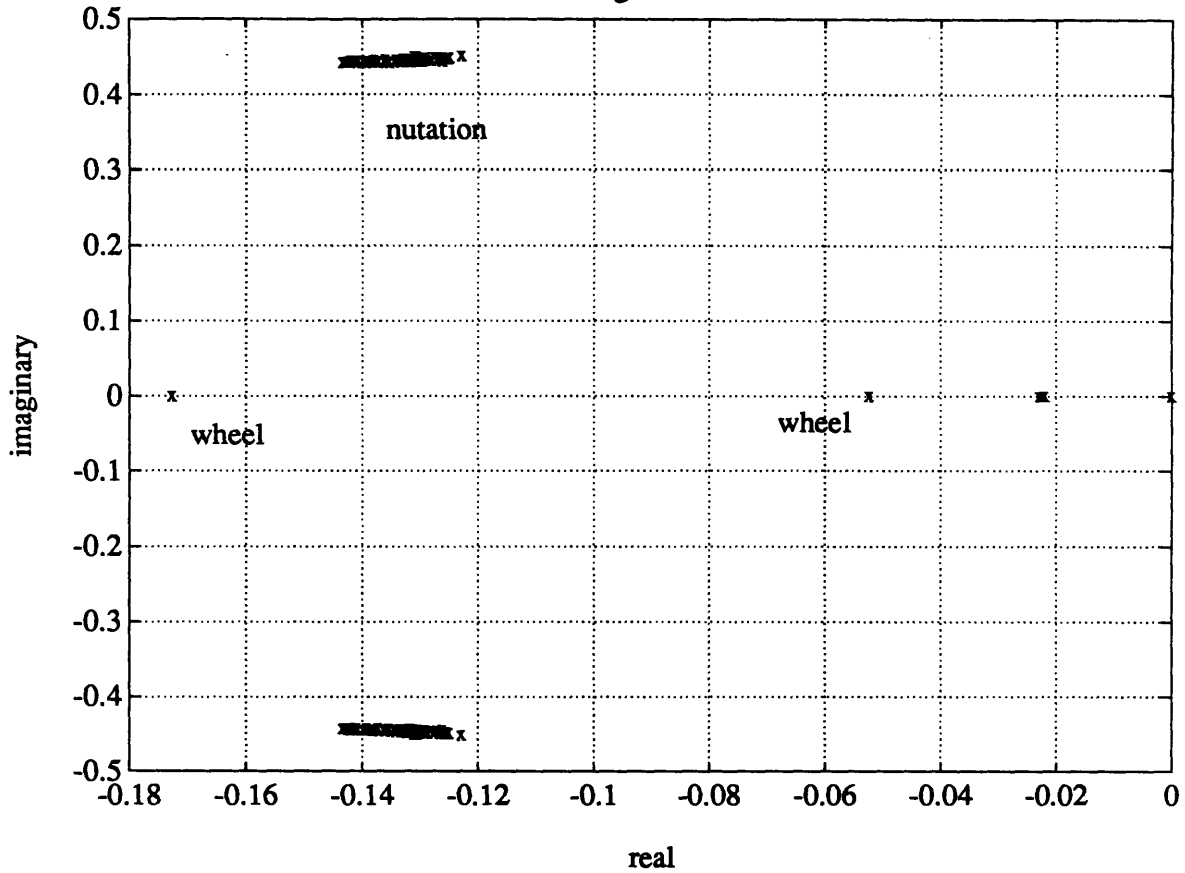
Figure 5.3

Figure 5.3: closed loop regulator root locus of night mode 2 controller, mapped to the s-plane via the relation $z = e^{st}$. The parameter θ is varied over a 2π range.
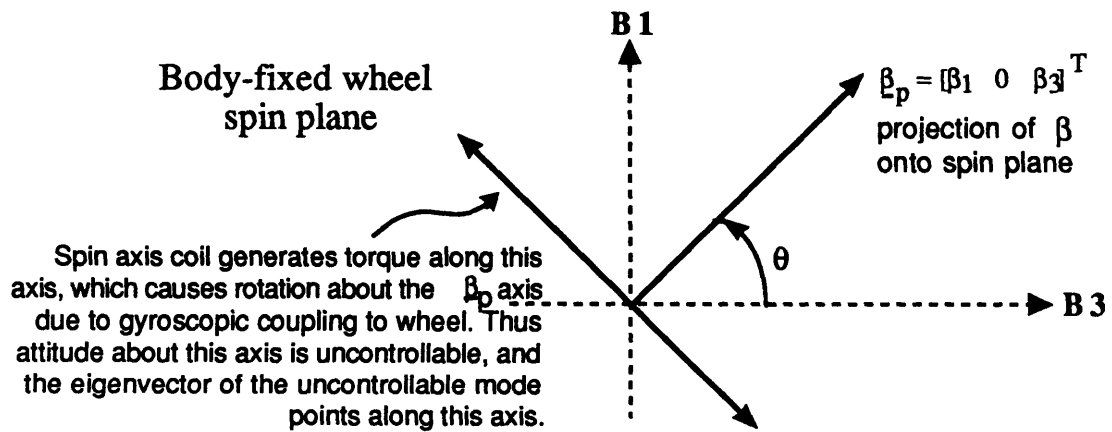
Figure 5.4: geometric interpretation of the uncontrollable mode

5.2.3 Stability Robustness Properties

Actual plant uncertainties are classified broadly as structured (parametric) and unstructured. Unstructured plant perturbations challenge the adequacy of the design plant model equations of motion and are dealt with assuming worst-case phase uncertainty, which is necessarily conservative. The stability of a linear feedback control system is robust to plant perturbations which have dynamics significantly above the closed-loop bandwidth. Since the implementation is discrete time, the closed loop bandwidth is limited to be no higher than the loop Nyquist rate of 1/8 Hz (π/4 rad/s). The possible unmodeled plant uncertainties are:

- Structural resonances: the cantilevered natural frequency of the solar array paddles should be the lowest frequency structural mode around, and measurements from the same paddles built for a previous satellite[1] put a lower bound of around 30 Hz [Pers91] on the eigenfrequency. This is 240 times the Nyquist frequency.

- Actuators: inner control loops for the actuators are the current regulation loop for the magnetic torquers and the phase-locked loop to command wheel accelerations. The current regulators can safely be assumed to be much faster than the wheel torque control loop. A time constant of 1/30 sec. (95% rise time 1/10 sec. to a commanded step in acceleration) for the wheel torque inner loop is reasonable, which puts its frequency of 30 rad/s at 40 times the Nyquist rate.

Thus without rigorous proof, we are justified from spectral separation arguments in considering only parametric uncertainties to evaluate the practical stability robustness of the control designs presented, stating in effect that Euler's equations in the form presented in Chapter 4 adequately captures the dynamics in the frequency range of interest. This simplifies the analysis greatly. Furthermore, the standing assumption for the gain-scheduled controller is that θ varies sufficiently slowly compared to the control bandwidth to justify using linear time-invariant stability criteria. The test of this claim is left to simulation.

---

[1]ALEXIS from Aero/Astro Inc.

It is seen that the most important parameter variations of the actual plant from the design plant model are:

• Nutation frequency and open loop gain changes cause by mass properties differences and off-nominal wheel angular momentum.

• For actuator mode 2, θ offsets from inaccurate magnetometer measurements, causing mismatch of the DPM to the actual plant and resulting in the wrong gains being computed. Note that ψ measurements offsets determine actuator mode switches but do not affect closed loop stability.

Spin Axis Loop

We continue the assumption that effects due to principal/body frame separation are small. Decoupling of the spin axis control loop from the spin plane loop is exploited. The gain/phase margins of the SISO spin axis loop is easily examined. From the fact that the plant transfer function is:

$$\frac{\varepsilon_2(s)}{\tau_2(s)} = \frac{1}{2I_2 s^2} \tag{5.31}$$

it is clearly seen that the wheel speed and θ have no effect on the loop, as expected. I_2 variations cause a change in the open loop gain, so the gain margin is most significant. Figure 5.5 shows that the loop has a 10 dB upward gain margin, with the frequency response showing that the LQG solution is simply a lead compensator with the high frequency roll-off associated with DT implementation such that the upward gain margin is finite. A 10 dB upward gain margin corresponds to more than a factor of 3 drop in actual I_2 compared to the design value, which is ample headroom. The phase margin is seen to be about 50°.
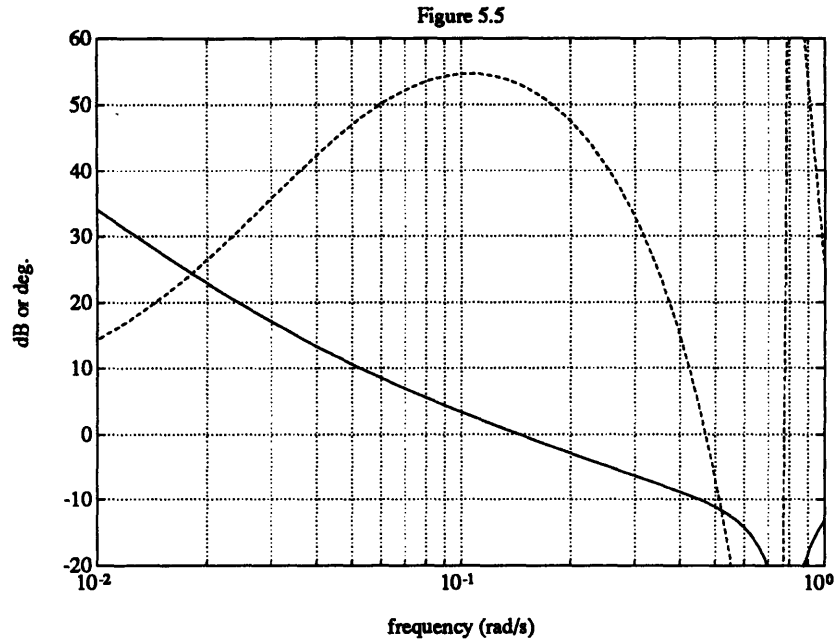
Figure 5.5

Figure 5.5: open loop Bode plots of spin-axis loop, the dashed plot representing the phase + 180° to facilitate reading the margins. The frequency response is of the discrete time transfer function for the DT stability boundary $z = e^{j\omega T}$, and has no meaning beyond the Nyquist frequency of $\pi/4$ rad/s. The upward gain margin is seen to be about 10 dB; the phase margin about 50°.
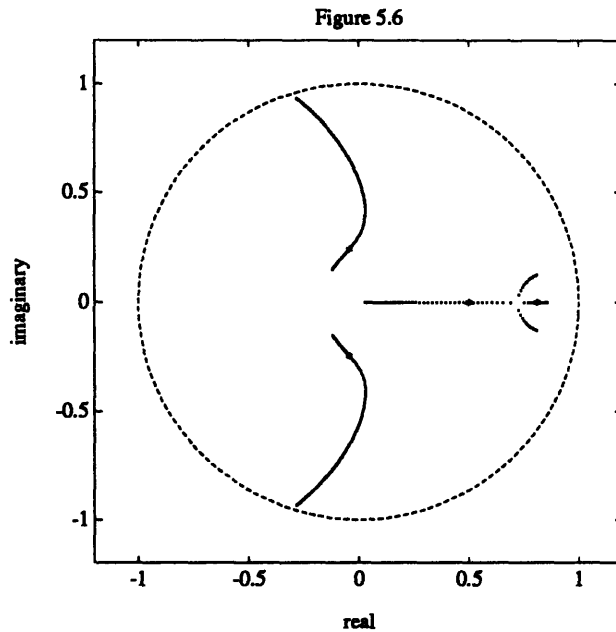
Figure 5.6

Figure 5.6: z-plane root locus for gain offset range of 60% ~ 320% of design open loop gain, validating the bode plot gain margin claim of 10 dB. The asterisks * represent nominal closed loop pole locations.

Spin Plane Loop. Full Authority Actuator Mode:

Spin plane controller stability robustness is most conveniently examined using closed loop root loci as functions of parameter variations. Again, decoupling is exploited to eliminate consideration of the spin-axis controller. Root loci are presented in the z-plane; the unit circle is the stability boundary. For the full authority actuator mode, the parameters varied are the wheel angular momentum h_w and principal inertias I_1-I_3. Recall from (4.49) that the open loop nutation frequency is approximately:

$$\omega_n = \frac{h_w}{\sqrt{I_1 I_3}} \qquad (5.32)$$

Furthermore, from the equations of motion it is seen that changes in h_w affect the natural nutation frequency (phase @ crossover), but changes in the body inertias also affect the control distribution matrix (open loop gain). Thus we expect less tolerance to mass properties offset. Figures 5.7 thru 5.9 detail the cases examined. Note that in all cases both filter and regulator poles are plotted- thus the plant-compensator system is of order 8. The general conclusion is that higher plant nutation frequencies are more de-stabilizing, so that it is better to underestimate than overestimate the inertias.

Spin Plane Loop. Partial Authority Actuator Mode:

Now the additional parameter θ is involved. Again, root loci is employed, this time plotted with respect to the full 2π range of θ, while discrete worst cases of θ, h_w and inertia offsets are in effect. The roots are plotted on the z-plane and include both regulator and filter poles, in contrast to figure 5.3. It is seen in figures 5.10 thru 5.12 that less headroom exists for the gain scheduled controller than the full actuator authority controller. It is observed again that tolerance is reduced for nutation frequency increases, so that it is better to underestimate the inertias. The worst cases considered are extreme and unlikely to be permitted to occur; overall, the conclusion is that the designs presented are viable from a stability robustness point of view.
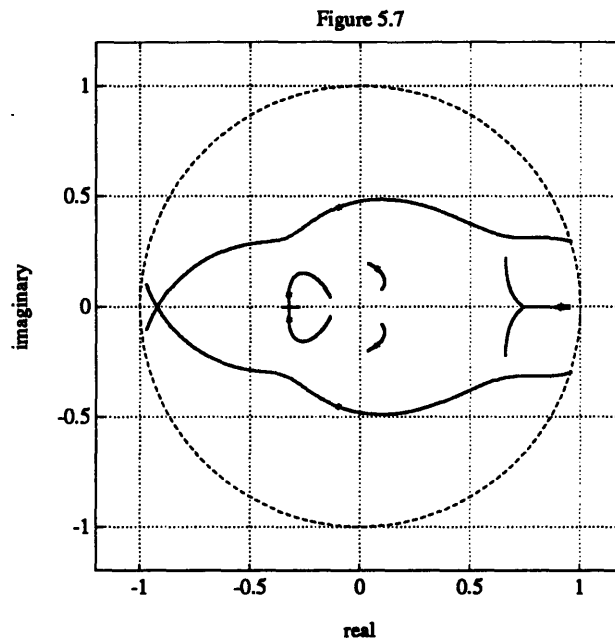
Figure 5.7

Figure 5.7: Night actuator mode 1 spin-plane controller closed loop root locus vs. wheel momentum. Plant h_w is varied in range 40% ~ 160% of design value; mass properties are not offset. Asterisks * represent nominal closed loop pole locations.
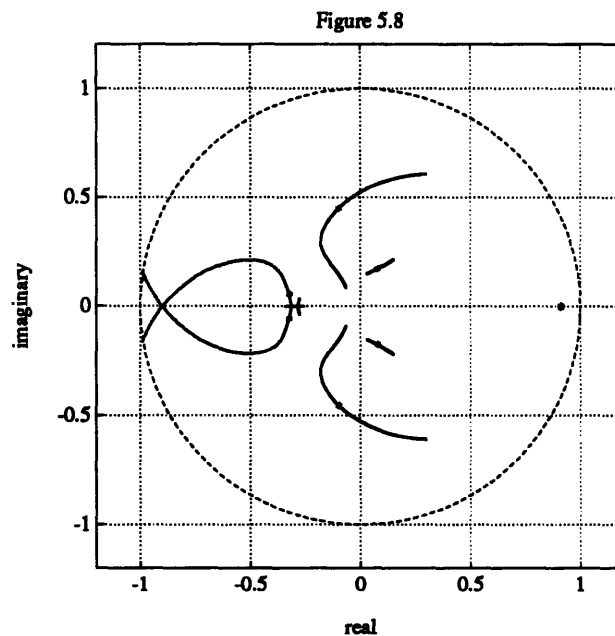
Figure 5.8

Figure 5.8: Night actuator mode 1 spin-plane controller closed loop root locus vs. plant inertial offsets. Plant principal inertias I_1 and I_3 are simultaneously varied in the range 65% ~ 135% of their design values, h_w is not offset. Asterisks represent design closed loop pole locations. The branch directions about to go unstable correspond to reduced inertias.
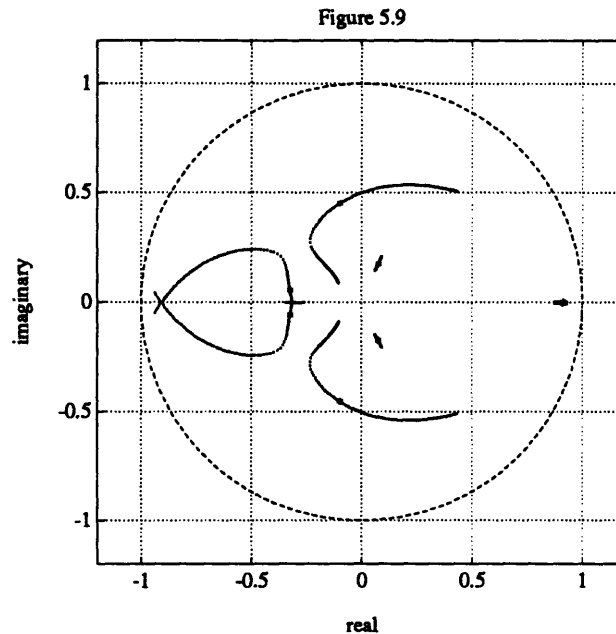
Figure 5.9

Figure 5.9: Night actuator mode 1 spin-plane controller closed loop root locus vs. plant perturbations; h_w and I_1 I_3 are varied simultaneously in worst case combinations: h_w from 80% to 120% while I_1 and I_3 from 120% to 80% individually. This gives open loop nutation frequency offsets from 67% to 150%. The branch directions about to go unstable represent increased plant nutation frequency. Asterisks again represent design closed loop pole locations.
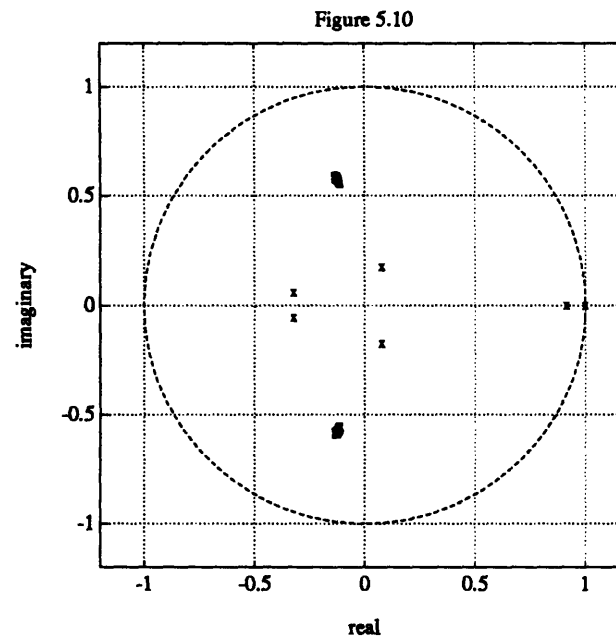
Figure 5.10

Figure 5.10: Night actuator mode 2 gain-scheduled spin plane controller, closed loop poles vs. θ, no plant perturbations modelled. This is the z-plane version of figure 5.3, with the inclusion of the filter poles, which are designed to be about 3 times faster than the regulator poles.

Figure 5.11a
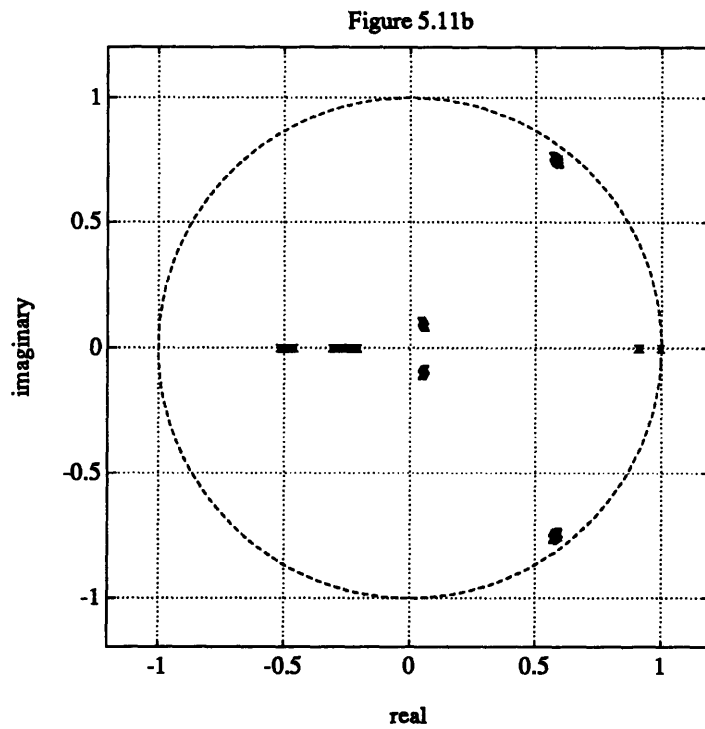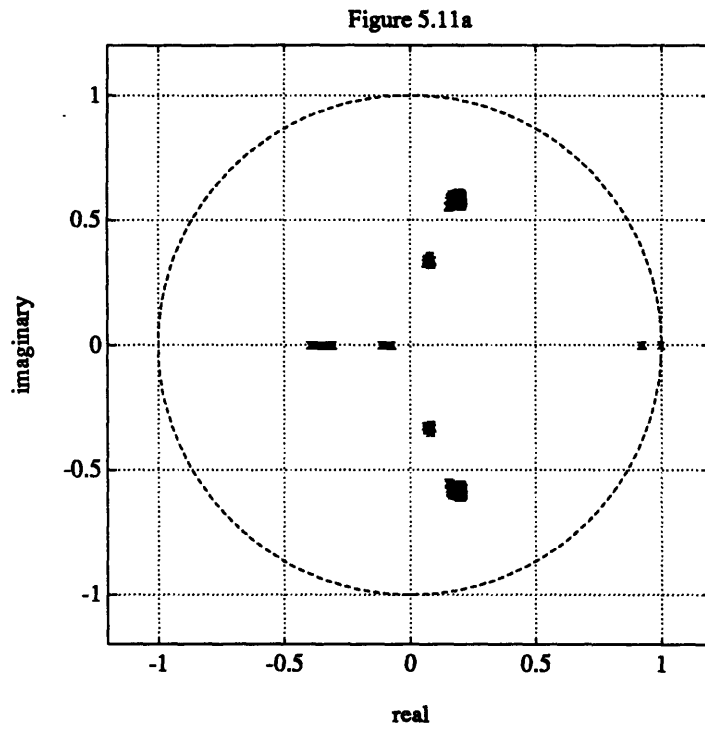
Figure 5.11b

Figure 5.11a,b: Night actuator mode 2 gain-scheduled spin plane controller, closed loop poles vs. θ. Plant perturbations modelled: I_1 and I_3 up 20% individually, h_w down 20%- resultant nutation frequency drop = 33%. θ measurement offset modelled: **5.11a:** $\Delta\theta = +40°$; **5.11b:** $\Delta\theta = -40°$.

Figure 5.12a
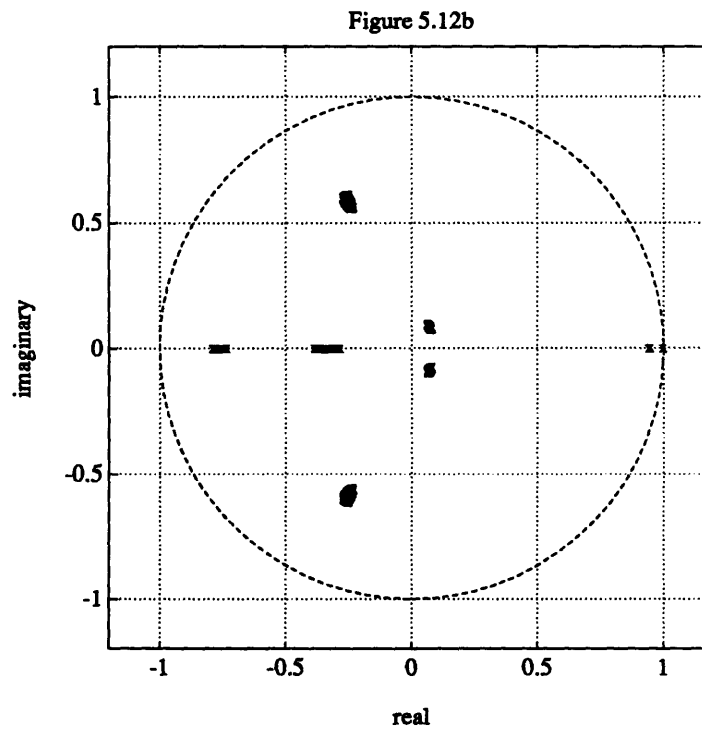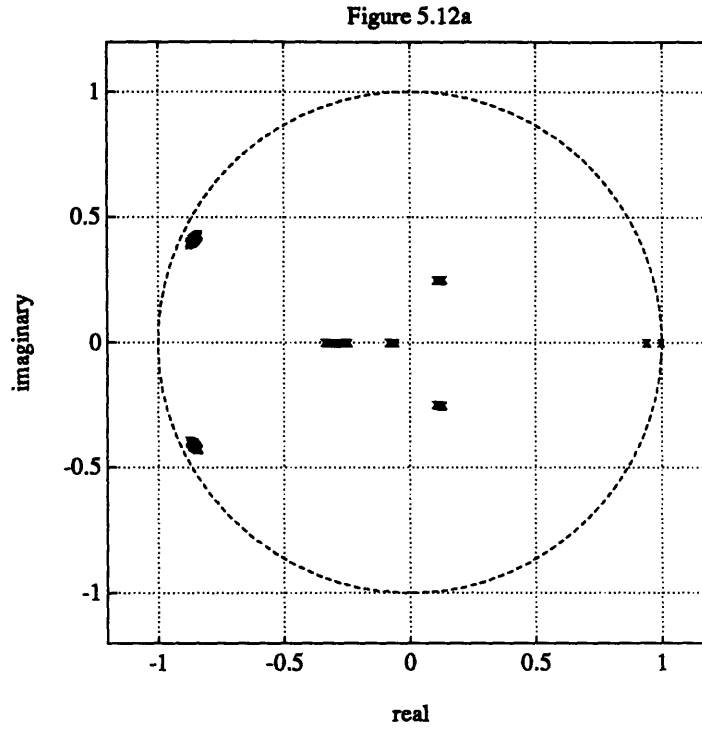
Figure 5.12b

Figure 5.12: Night actuator mode 2 gain-scheduled spin plane controller, closed loop poles vs. θ. Plant perturbations modelled: I_1 and I_3 down 15% individually, h_w up 10%- resultant nutation frequency rise = 29%. θ measurement offset modelled: **5.12a:** Δθ = +30°; **5.12b:** Δθ = -40°. Note the reduced tolerance for Δθ for increased nutation frequency.

The plant perturbations considered in this analysis were isolated because of their dominant effect on closed loop stability as seen on physical grounds. Other perturbations involving sensor/actuator imperfections mostly contribute to performance degradation. The most important of these- sensor and actuator output quantization due to the digital implementation of their inner control loops, is addressed in detail in later sections. Other effects are modelled and left to simulation study.

Code for generating the results of this section is included in Appendix A-5b.

5.2.4 Disturbance Rejection Properties

The primary purpose of the orbit night controller is to reduce the effect of disturbance torques on body rate buildup. An effective analysis method of controller effectiveness is to consider the closed loop frequency response from disturbance torque to body rate. It is noted that this analysis is necessarily optimistic- it assumes linear operation, which means no actuator saturations are allowed, and it assumes no plant perturbations. However, the margins achieved are large for the specifications and the disturbance spectrum analysis in Chapter 3.

This is the counterpart to Section 3.3.1. The singular value gain-band generalization of the MIMO frequency response is used, with the transfer functions derived in z and evaluated along the unit circle $z = e^{j\omega T}$. The algebra is sketched below:
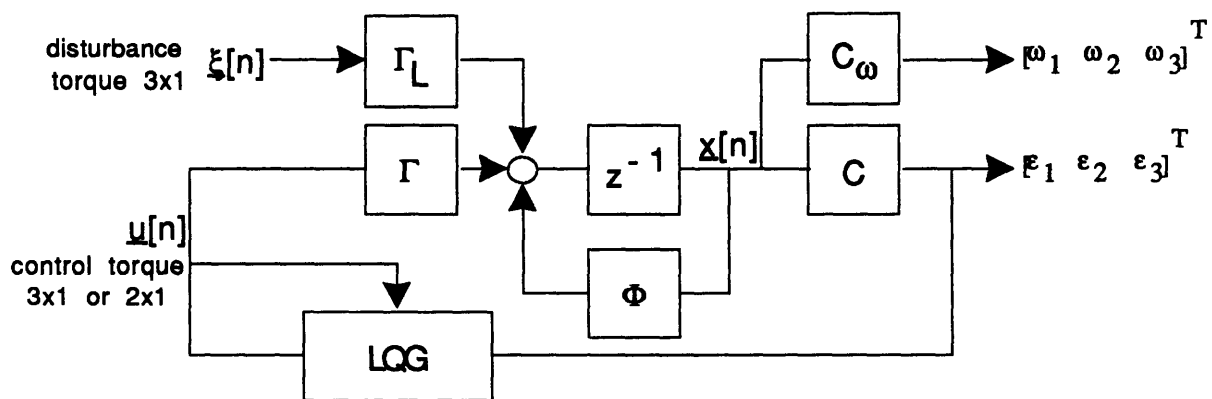
Figure 5.13: block diagram showing injection of disturbance torque.

Define: $\underline{\omega}(z) = G_d(z)\underline{\xi}(z)$

$$\Gamma_L = \begin{bmatrix} \frac{1}{I_1} & & \\ & \frac{1}{I_2} & \\ & & \frac{1}{I_3} \\ \hline & 0_{3\times3} & \end{bmatrix} \qquad C_\omega = [I_{3\times3} \mid 0_{3\times3}]$$

Γ as appropriate for actuator mode (eq. (5.4) or (5.7))

Compensator:

state estimate: $\underline{\hat{x}}[n] = (I - HC)\underline{\bar{x}}[n] - H\underline{e}[n]$ (5.33)

loop closure: $\underline{e}[n] = -C\underline{x}[n]$ (5.34)

control output: $\underline{u}[n] = -G\underline{\hat{x}}[n]$ (5.35)

compensator state: $\underline{\bar{x}}[n+1] = \Phi\underline{\hat{x}}[n] + \Gamma\underline{u}[n] = (\Phi - \Gamma G)\underline{\hat{x}}[n]$ (5.36)

Plant:

$$\underline{x}[n+1] = \Phi\underline{x}[n] + \Gamma\underline{u}[n] + \Gamma_L\underline{\xi}[n] \qquad (5.37)$$

Substitute and Append:

$$\begin{bmatrix} \underline{x}[n+1] \\ \underline{\bar{x}}[n+1] \end{bmatrix} = \begin{bmatrix} \Phi - \Gamma GHC & -\Gamma G(I - HC) \\ (\Phi - \Gamma G)HC & (\Phi - \Gamma G)(I - HC) \end{bmatrix}\begin{bmatrix} \underline{x}[n] \\ \underline{\bar{x}}[n] \end{bmatrix} + \begin{bmatrix} \Gamma_L \\ 0_{6\times3} \end{bmatrix}\underline{\xi}[n] \equiv [\Phi_d]\underline{\tilde{x}}[n] + [\Gamma_d]\underline{\xi}[n] \quad (5.38)$$

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = [C_\omega \mid 0_{3\times6}]\begin{bmatrix} \underline{x}[n] \\ \underline{\bar{x}}[n] \end{bmatrix} \equiv C_d\underline{\tilde{x}}[n] \qquad (5.39)$$

Z-transform and form disturbance transfer function relating disturbance torque input to body rate output:

$$G_d(z) = C_d(zI - \Phi_d)^{-1}\Gamma_d \qquad (5.40)$$

Equation (5.40) is viewed as a frequency dependent linear transformation with z = $e^{j\omega T}$. The input and output spaces are spanned by a singular value decomposition at each ω up to the Nyquist rate of 1/8 Hz ($\pi/4$ rad/s). The resulting minimum and maximum singular value functions comprise the gain-band of the closed loop system. It is important to note that the gain has units of (rad/s)/(N•m). For reference, some relevant figures are:

- 1 °/hr = $\pi/(3600 \cdot 180)$ rad/s = 4.85×10^{-6} rad/s
- Circular orbital rate @ 300 nautical mile: 1.1×10^{-3} rad/s

Figure 5.14 show the disturbance rejection gain band for the night full authority mode controller, where it is shown that in the design of the LQG solution the weights were chosen to circularize (i.e. match max and min singular values) the disturbance rejection dynamics. Note that the Nyquist rate for a 4 sec. sampling period is $\pi/4$ (0.785) rad/s, and that the plot is meaningful for frequencies small compared to this, since it is based on a discrete-time approximation of the continuous plant itself. The input and output spaces are in this case both of dimension 3, so 3 singular value traces exist.

The controller exhibits ample ability to handle low-frequency content disturbances. Comparison with the expected disturbance magnitudes and spectra calculated in Chapter 3 show a wide margin to meeting the 2 °/hr body rate output specifications. Again, it is important to note the limitations of the analysis. High-bandwidth disturbances due to sensor and actuator imperfections pass right through and will need to be addressed separately in the next section.

Figure 5.15 shows the corresponding plot for the partial authority controller for an arbitrary θ. The plot does not vary greatly across the 2π range for θ. Again, the input and output spaces have dimension 3- recall the input space contains disturbance torque vectors. Recall also that the right singular vectors at each frequency form an orthogonal basis for the input space, while the left singular vectors span the output space. Upon numerical examination, the maximum right singular vector which corresponds to the flat trace at about -6 dB points in the spin plane along θ for all low frequencies, precisely the direction which the spin-axis coil is unable to generate a cancelling control torque. Furthermore, -6dB corresponds to a factor of 0.5. Recall the argument presented in Section 4.3.4, equation (4.52), where the attenuation by gyroscopic stiffness of this uncanceled disturbance component is by a factor of $1/h_w$. The design value of h_w is (0.0063 Kg•m^2)•(3000 RPM) = 1.9792 N•m•s. This is a numerical demonstration of the results in Section 4.3.4. Disturbance torques components other than the one pointing along θ in the spin plane can

be expressed as linear combinations of the other two right singular vectors and are thus suppressed as effectively as by the full authority mode controller. The leftover component, whose magnitude will be significantly smaller except in perverse circumstances, is suppressed 6 dB by gyroscopic stiffness. Thus we stand a good chance of meeting specifications with the partial authority mode controller as well. The same caveats discussed in the previous paragraph apply. Appendix A-5c contains code for the analysis done in this section.
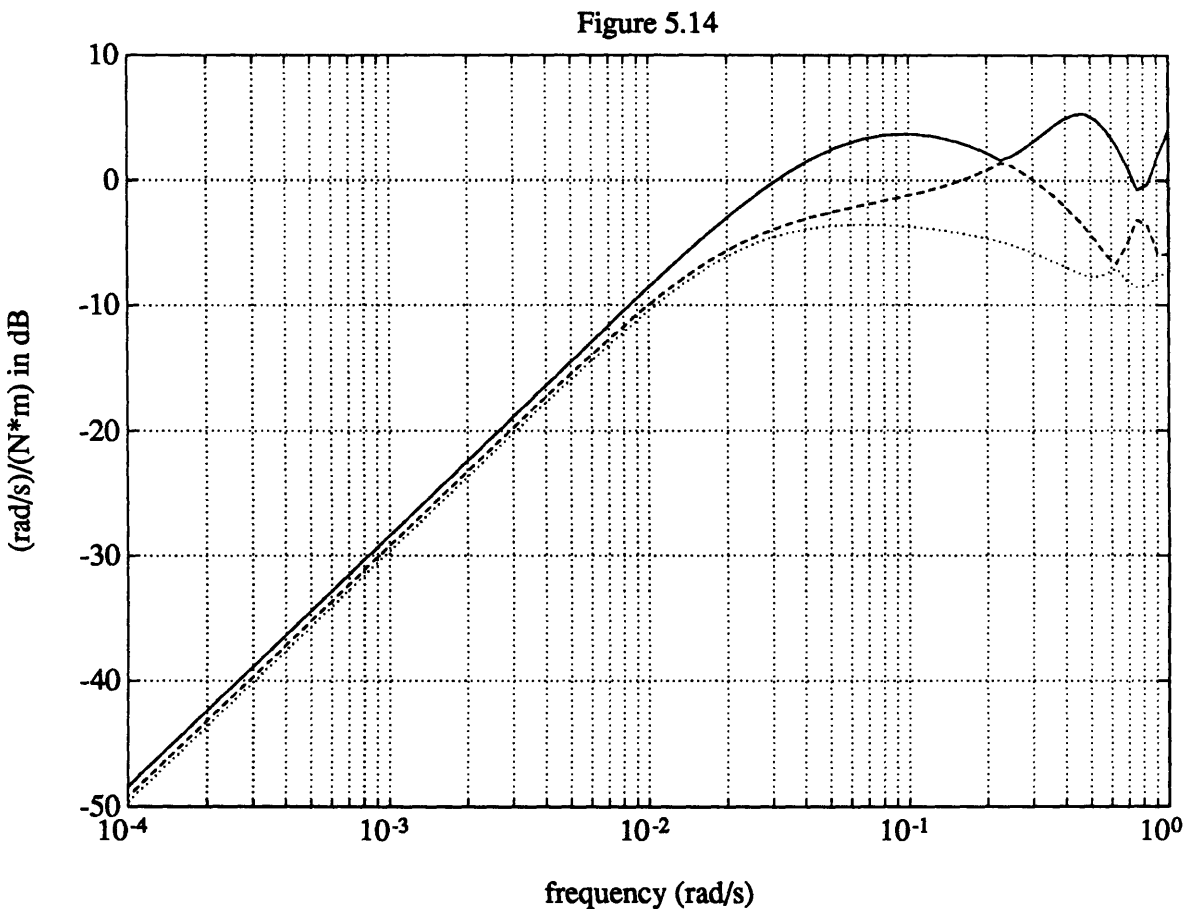
Figure 5.14

Figure 5.14: disturbance rejection frequency response of the night mode 1 controller, torque input of units N•m, body rate output of units rad/s.
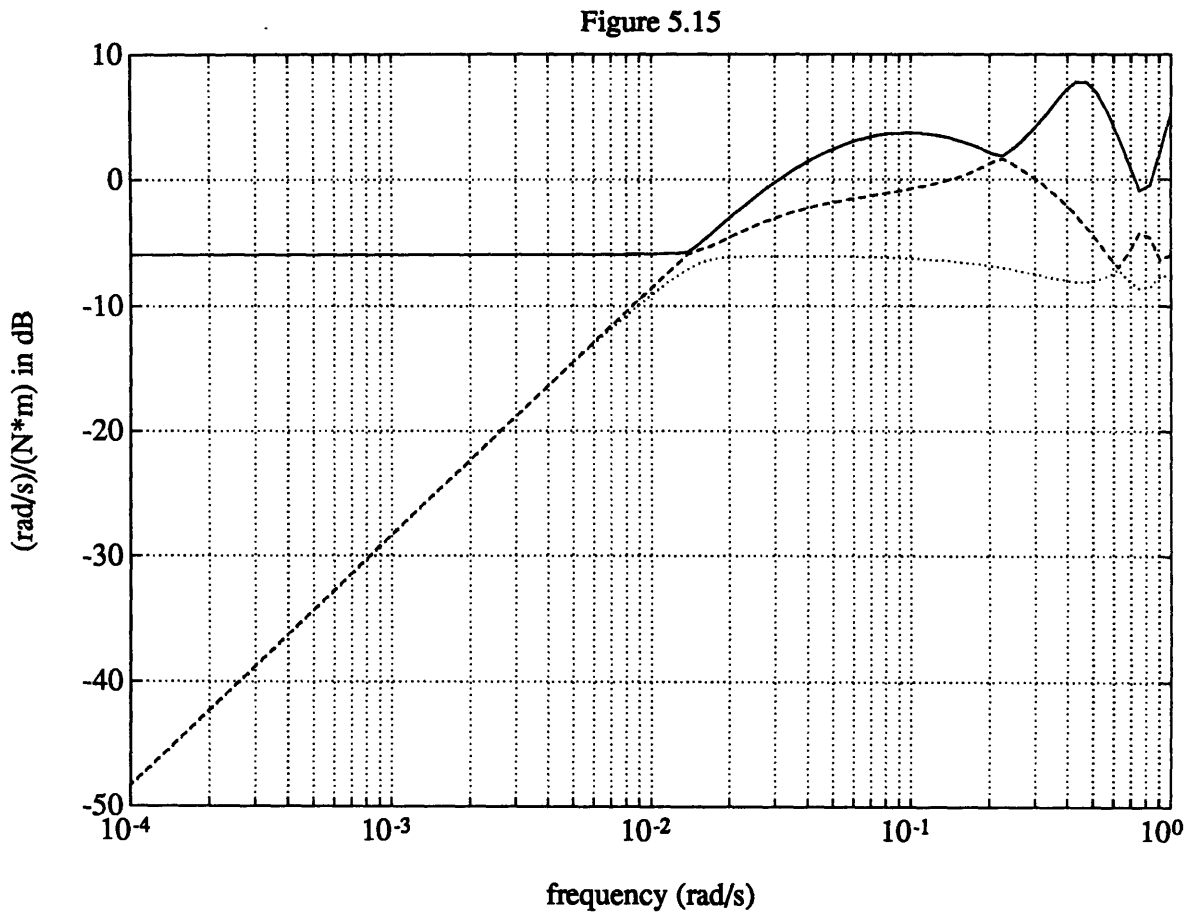
Figure 5.15

Figure 5.15: disturbance rejection of the night mode 2 controller, gains scheduled for an arbitrary θ = 330°. Disturbance torque input is of units N•m, body rate output has units of rad/s.

5.2.5 Actuator Quantization Effects on Performance

Actuators are of course never perfect, and models for the imperfections are discussed in Chapter 2. Of the various imperfections the most threatening to meeting specifications is magnitude quantization arising from the digital implementation of the inner loops controlling them. This is particularly true of the momentum wheel control loop. Assumedly, it will be a phase-locked loop arrangement with either optical or Hall-effect angular displacement sensors present. The quadrature counts provide a lower limit to the spacing between each discrete acceleration commandable. The current-control loop

for the magnetic torque coils will likely be a pulse-width modulated scheme to maximize efficiency by operating the drive transistors in their saturated regime- the spacing between the discrete pulse widths are limited by the resolution of the DAC used. Since these inner loops are to be designed, it is desirable to calculate the effects of the quantization on controller performance so as to provide design guidelines.

Quantization in the attitude measurement of CCD cameras is not considered because at the time of writing it is still early in the development of the instrument and its algorithms, making it difficult to obtain estimates of the quantization magnitude. The angular resolution to one particular star is known (approximately 3 arcsec.), but that resolution can be improved upon substantially for calculation of the attitude by observing many stars to generate an optimal fit to a catalog. It is assumed in simulations that the signals are perfect, with the knowledge that performance obtained will be degraded accordingly. This may in fact be a better assumption than any other at the time [Pers91].

The analysis approach is to regard the control quantization as a stochastic disturbance torque on top of the commanded control torque and calculate the RMS response due to the stochastic component [Fran80]. The system under consideration is the same as in the previous section, with figure 5.13 modified such that $\Gamma_L = \Gamma$. The disturbance vector $\xi[n]$ is thus such that:

$$\underline{u}_a[n] = \underline{u}[n] + \underline{\xi}[n] \qquad (5.41)$$

where: $\underline{u}[n]$ = commanded control torque from compensator
$\underline{u}_a[n]$ = actual (quantized) control torque to plant

It should be pointed out that the quantities quantized are actually the wheel acceleration and the coil currents, but the abstraction to quantized torques is used here. Two further assumptions are made to make application of Lyapunov steady state analysis possible:

• Let the quantization step be defined as q (units N•m). The quantization process is one of rounding, so at each time step n, assume each component of the random vector $\xi[n]$ is governed by a zero mean, uniform probability density function from -q/2 to +q/2. Assume further that the components are mutually uncorrelated.

• Assume the process $\xi[n]$ is a white noise sequence (process).

The first assumption is reasonable. Even though in the full authority actuator mode the torque generated is actually a combination of the coils and wheel, thereby making the disturbance torque's PDF the convolution of several uniform PDF's, this is not a detail worth chasing after. The second assumption requires that the power spectral density of $\xi[n]$ be wide compared to the closed loop bandwidth of the control system. This is believable if the commanded control signals are on the order of q, which would reduce the correlation of two sequential rounding errors.

In any case, the assumptions allow the covariance matrix of $\xi[n]$ to be stated :

$$E\left(\underline{\xi}[n]\underline{\xi}^T[n+k]\right) = 0 \qquad k \neq 0 \qquad\qquad (5.41)$$

$$E\left(\underline{\xi}[n]\underline{\xi}^T[n+k]\right) = \begin{bmatrix} \dfrac{q_1^2}{12} & 0 & 0 \\[2mm] 0 & \dfrac{q_2^2}{12} & 0 \\[2mm] 0 & 0 & \dfrac{q_3^2}{12} \end{bmatrix} \qquad k = 0 \qquad\qquad (5.42)$$

where the diagonal entries of (5.42) are recognized as the variance of a random variable with a uniform PDF. Let the autocovariance of $\xi[n]$, (5.42), be denoted by R_ξ. The closed loop system represented by (5.38) and (5.39) is then solved for the autocovariance of the augmented state $\tilde{x}[n]$, defined as

$$R_{\tilde{x}}[n] \equiv E\left(\underline{\tilde{x}}[n]\underline{\tilde{x}}^T[n]\right) \qquad\qquad (5.43)$$

and has dynamics governed by the Lyapunov equation [Kwak72]:

$$R_{\tilde{x}}[n+1] \equiv \Phi_d R_{\tilde{x}}[n]\Phi_d^T + \Gamma_d R_\xi \Gamma_d^T \qquad\qquad (5.44)$$

but in fact we are really interested in the autocovariance of the body rates:

$$R_\omega[n] = C_\omega R_{\tilde{x}}[n] C_\omega^T \qquad\qquad (5.45)$$

where Φ_d, Γ_d, and C_ω are all defined as in equations (5.38) and (5.39). When Φ_d has all its eigenvalues strictly in the unit circle, (5.44) evolves to a steady state for $R_{\tilde{x}}[\infty]$. It is noted, however, that in the partial authority actuator mode, one eigenvalue of Φ_d is on the unit circle. However, it has been demonstrated that the eigenvector of that uncontrollable mode has no components in the rate directions in state space for all gains scheduled in the full range of θ. Therefore, in actuator mode 2, the corresponding elements of equation (5.44) do not approach a steady state, but (5.45) then avoids all of them.

Since $\xi[n]$ is zero mean, the square roots of the diagonal elements of $R_\omega[\infty]$ from (5.45) are the steady state RMS value of the rate outputs given quantization widths q_1 thru q_3. The following numerical values are used:

- q_2: corresponds to spin axis control torque step. $q_2 = 5\ \mu N \cdot m$ is assumed, as mentioned in Chapter 2.

- q_1 and q_3 correspond to spin-plane torque components which are coil generated. The actual current quantization step is the 2 time the maximum capacity (for + and -) divided by the DAC resolution. As modelled in Chapter 2, the values used give $8/256\ [A \cdot m^2]$. An average value for the ambient field strength must be assumed to translate this to a torque step; $20\ \mu T$ is selected, giving $q_1 = q_3 = 0.625\ \mu N \cdot m$. This is seen to be small compared to q_2.

Numerical results: (rates have been converted to °/hr for convenience)

Actuator Mode:	RMS(ω_1):	RMS(ω_2):	RMS(ω_3):
Full Authority	0.0367	0.2751	0.0395
Partial Authority	< 0.04	0.2751	< 0.04

The RMS(ω_1) and RMS(ω_3) results for actuator mode 2 were calculated over the span of θ, hence the upper bound presentation. Note also that RMS(ω_2) is the same in either actuator mode, as it ought to be. It should be mentioned that the calculated results are surprisingly close to those obtained in simulation, making this a reasonable design decision aid. The results are encouraging- it can be argued from an ad-hoc normal PDF assumption that peak rates should not be expected to exceed 3 times the RMS value, and in fact the largest peaks observed in simulation are around twice the calculated RMS value. It is seen that the effect of the coil quantization is negligible, and that the effect of quantization of the wheel torque is the only one worth worrying about. The calculations are

repeated for a range of q_2's to facilitate design; the result is plotted in figure 5.16. As expected for a linear plant, RMS(ω_2) rises linearly with q_2. The code for this section is contained in Appendix A-5d.
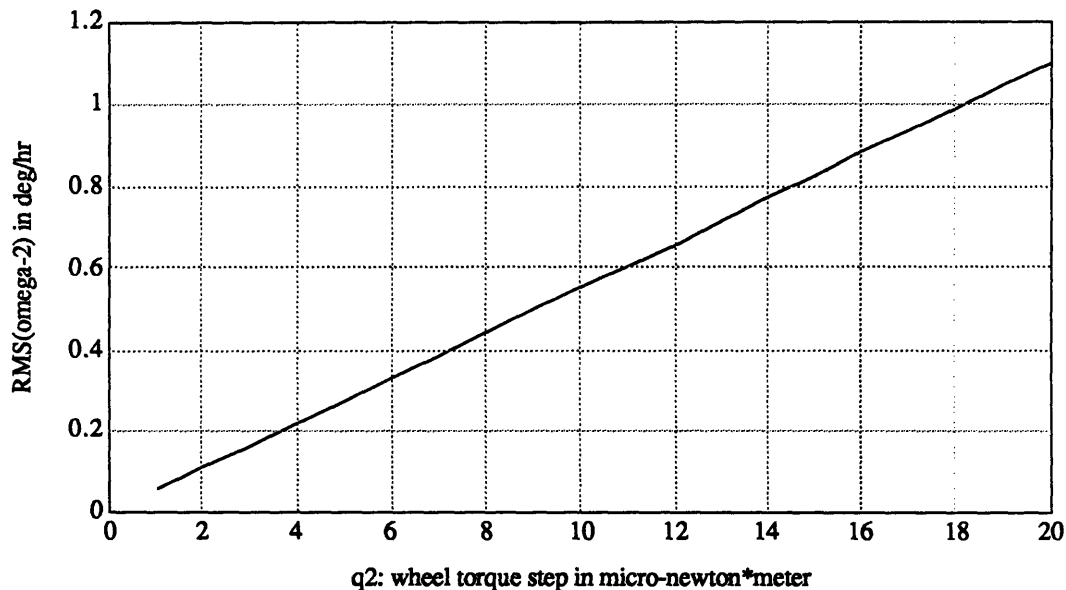
Figure 5.16: calculation of RMS(ω_2) for a range of q_2 discretization steps. It should be remembered that the RMS value is not the peak value, and that the analysis is performed assuming an ideal plant. Therefore there should be a factor of at least 2 to 3 between the rate spec and the designed RMS value when actually specifying the resolution of the wheel acceleration control loop.

5.2.6 Coil Saturation Anti-Windup

Two practical implementation issues sufficiently important to merit separate discussion are covered in this section and the next: an anti-windup strategy to deal with the eventuality that the magnetic coils will be saturated (commanded past their peak dipole capacity) during recovery from large initial conditions, and a bumpless transfer strategy to minimize rate disturbances when switching actuator modes. It is noted that both techniques have worked well in simulation and in fact would have forced re-designs of the controllers had they not been successful.

Saturation of the wheel torque capacity is not considered. The base design model momentum wheel is the Ithaco Scanwheel, which as mentioned in Chapter 2 have a

maximum torque capacity of 20 mN•m. This is about 3 orders of magnitude above the expected disturbance torque level and enough torque to accelerate the spacecraft by about 800 deg/hr/sec about its axis- far more than any anticipated need of the controller to damp transients. On the other hand, the 4 A•m^2 capacity coils generate only 80 μN•m in a 20 μT field apiece, and while this is plenty for disturbance rejection purposes, the possibility for saturation while damping transients certainly exists.

Windup is a phenomenon associated with actuator magnitude saturations and refers to large excursions of the compensator states when the actual control applied to the plant differs significantly from the commanded. Strategies of varying sophistication and theoretical rigor exist (see, for example, [Kapa88] and its references) to deal with this issue, but it is found that this particular system is sufficiently benevolent that a fairly simple strategy is adequate. Since the compensator is "model-based," the state prediction step of the estimator is fed a drive control signal which is software saturated, so that the estimator better tracks actual plant state. The software saturation is set slightly below the expected hardware saturation. The performance thus obtained is acceptable and permits the use of smaller torque coils than otherwise would be necessary. Figure 5.17 details the modification.
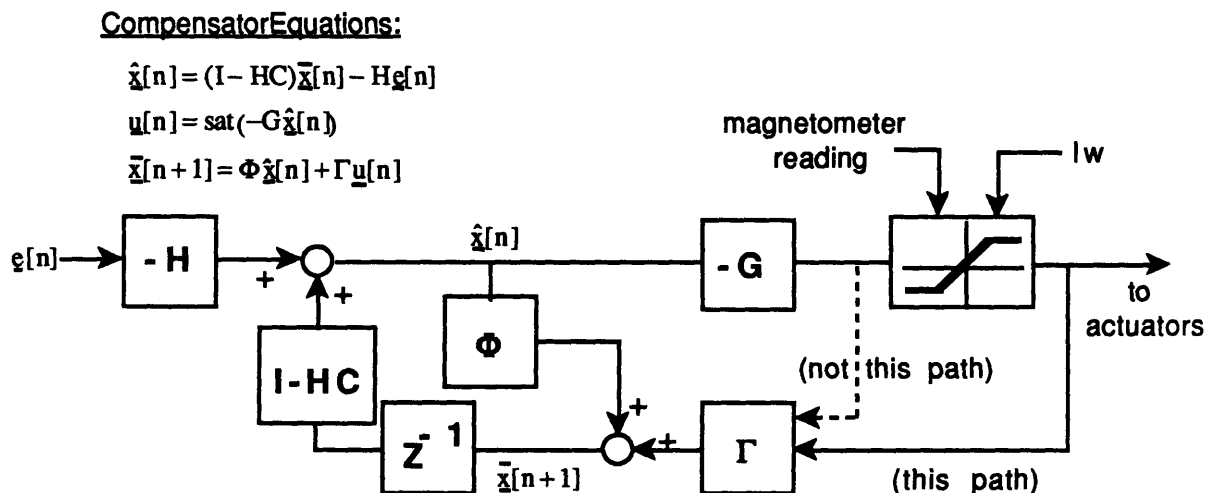
Figure 5.17: block diagram of the LQG compensator with software saturation in loop. The anti-windup fix is simply to feed the estimator prediction step with the saturated control output.

5.2.7 Bumpless Actuator Mode Switching

Chapter 4 detailed the switching of orbit night controllers as a function of the angle Ψ when the field arrangement forces a change in actuator strategy, and this chapter has presented the separate controller designs. A controller switch produces transients because of the imposed change of the eigenstructure of the closed loop system. Generally, ad-hoc but widely used gain scheduling design methods call for "blending" smoothly the gains in one regime from another when switching. These methods worked poorly when tried in simulation because they fail to address the cause of large rate transients. That cause is the tendency of the controllers to null out errors in both rate and attitude, the attitude being defined as relative to some default reference frame. When attitude errors exist, the controller nulls them out at the designed time constants and at the expense of incurring rates. This explains the observation made in simulations that actuator mode-switching transients are most severe when transitioning from partial authority to full authority mode. Recall from Section 5.5.2 that in partial actuator authority a closed loop integrator with its eigenvector pointing along ϵ_1 and ϵ_3 exists, representing the uncontrollability of attitude along θ±90° on the spin plane. The resultant attitude error buildup then accounts for the large rate transient when switching to the full actuator authority controller. In summary, the controller tries to reduce "initial value" attitude errors according to the time constants of the relevant closed loop modes. Therefore, rates are incurred accordingly, causing undesirable transients. The situation is worsened if attitude errors are large enough to cause actuator saturation while correcting for them.

An obvious solution would seem to be turning down the bandwidth of the attitude feedback loops, which in the context of the LQG design methodology would mean reducing the weights on attitude error in the LQR cost function. In fact, it can be argued that given the nature of the specifications, there is no reason to weight attitude errors at all. The problem with this approach is that disturbance rejection of disturbance torques on rates is severely compromised as the attitude error loop bandwidths are reduced. This can be shown numerically. Figures 5.18 and 5.19 show disturbance rejection frequency responses for actuator modes 1 & 2 respectively, with the weights on attitude errors set to 0 in the LQR cost functions. The resulting gain matrices have zeros in the elements which feed back states ϵ_1 thru ϵ_3. The plots are to be compared against figures 5.14 and 5.15 respectively.

However, since the computed attitude parameters' null reference frame is arbitrarily defined, we can take advantage of the loose attitude tracking specification in the following manner:

* Upon switching actuator modes, define the attitude at the transition point as the null reference frame $[\underline{\varepsilon}^T \eta] = [0\ 0\ 0\ 1]$. Feed back to controller attitude deviations with respect to this frame from there on.

* Upon transitioning from orbit day to orbit night, do the same thing.

This trick is found in simulation to work quite well, as expected.

In summary, it is seen that for the sake of minimizing the effect of disturbance torque on body rates, it is necessary to feed back in the regulator loop deviations in attitude. But since we are free to choose the reference frame from which the attitude is measured, smooth transitions are achieved by redefining the null reference frame to be the attitude at the point of transition. The effectiveness of this trick argues against the implementation of any more elaborate schemes for "continuous Ψ gain scheduling," as mentioned in Section 4.1.3. Note also this "initial value intercept" strategy is essentially another form of the common control design practice of putting a pre-filter on the servo-input signal to a control loop to reduce transients caused by high-bandwidth servo commands. An initial value is- equivalent to a step servo command at $t = 0$. The difference is that in this case the "pre-filter" has an infinite decay time. In the orbit day controller, where tracking the sun vector is important, the same problem is solved with an "initial value intercept pre-filter" of finite decay time, a strategy which is also found to work well.
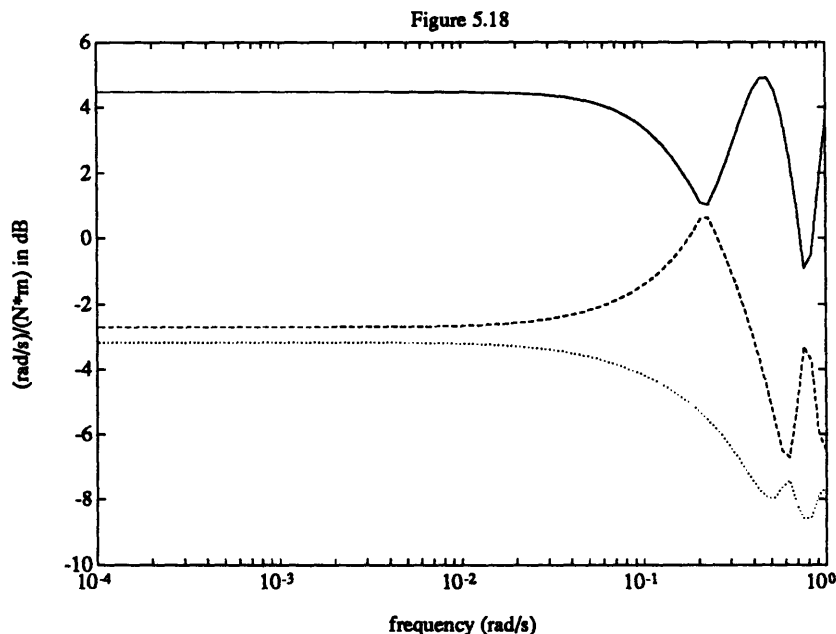
Figure 5.18

Figure 5.18: disturbance rejection frequency response for full-authority actuator mode, regulator control law designed with no feedback of attitude error. Compare with figure 5.14.
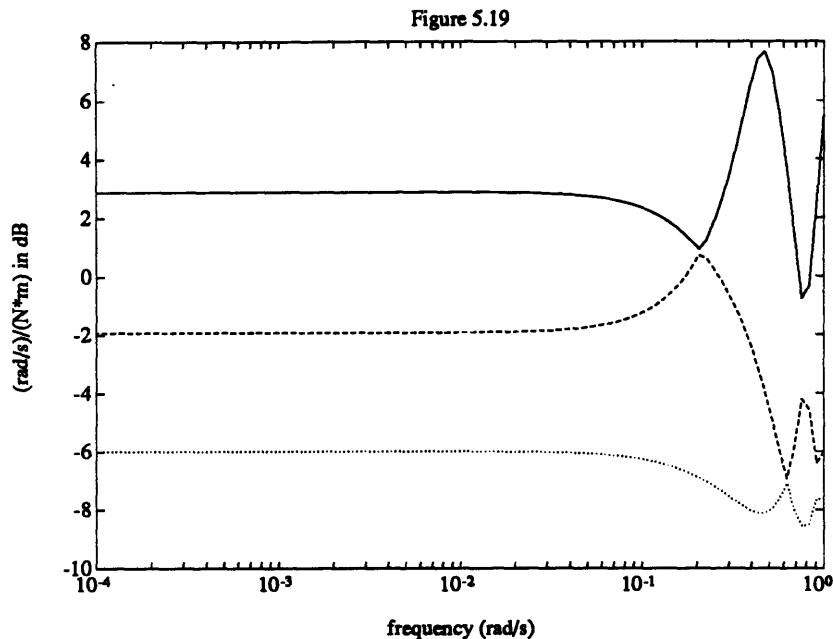
Figure 5.19

Figure 5.19: disturbance rejection frequency response of partial authority actuator mode, regulator control law designed with no feedback of attitude error. θ is arbitrarily set st 330°. Compare with figure 5.15. Note that the gyroscopic stiffness rejection of -6 dB is now the minimum singular value as opposed to the maximum.

5.3 Orbit Day Controller Design

This section discusses the attitude control and momentum management requirements of nominal orbit day operations. The subset of issues specific to implementing a slew maneuver are discussed separately in Section 5.4.

5.3.1 Control Strategy

Recall the orbit day controller specifications:

• Maintain body rates to < 20°/hr about all three axis with respect to inertial space to permit CCD cameras to obtain an attitude lock upon transitioning to orbit night.

• Maintain the sun vector to within a 5° cone about the body B_3 axis.

A design decision which is crucial to the control strategy adopted for the orbit day is:

> • With initial body rates which are within specs, gyroscopic stiffness alone- without nutation damping- is sufficient to meet the rate specification for the spin-plane angular velocity components ω_1 and ω_3 in the face of expected disturbance torques. (ω_2 clearly requires active control all the time.) Therefore, the spin-plane rates control loop can be turned off temporarily. The implications of this are discussed below.

This is to be contrasted to the orbit night. Summarily, the design factors which determine h_w, the stored angular momentum magnitude, are that it be low enough to permit active nutation damping within the sensor-dictated bandwidth constraint of the orbit night but high enough to meet this demand. The former has been demonstrated in the previous sections; we show the latter here:

Assume again decoupled behavior afforded by $\mathcal{F}_B = \mathcal{F}_P$, which permits the linearized spin-plane dynamics to simply be written as:

$$\frac{d}{dt}\begin{bmatrix} \omega_1 \\ \omega_3 \end{bmatrix} = \begin{bmatrix} & \dfrac{h_w}{I_1} \\ -\dfrac{h_w}{I_3} & \end{bmatrix}\begin{bmatrix} \omega_1 \\ \omega_3 \end{bmatrix} + \begin{bmatrix} \dfrac{1}{I_1} & \\ & \dfrac{1}{I_3} \end{bmatrix}\begin{bmatrix} \tau_1 \\ \tau_3 \end{bmatrix} \qquad (5.46)$$

The vector $\underline{\tau}(t)$ is taken to be a disturbance torque on the spin plane. Since the environmental disturbances are at frequencies comparable to orbit rate (Chapter 3), they can be approximated as DC when compared to the spacecraft's dynamics. The forced response of (5.46) can then be derived simply with transform methods. The torque vector components are represented as steps of magnitudes g_1 and g_3 respectively:

$$\begin{bmatrix} \tau_1(s) \\ \tau_3(s) \end{bmatrix} = \begin{bmatrix} \dfrac{g_1}{s} \\ \dfrac{g_3}{s} \end{bmatrix} \qquad (5.47)$$

Then the forced response is:

$$\begin{bmatrix} \omega_1(t) \\ \omega_2(t) \end{bmatrix} = L^{-1}\left\{ [sI - A]^{-1}B\underline{\tau}(s) \right\} \qquad (5.48)$$

where: A,B are as in (5.46), I is the 2x2 identity matrix

$\underline{x}(s)$ is as in (5.47)

L^{-1} is the inverse Laplace operator

which, after some algebra, yields:

$$\omega_1(t)=\frac{g_3}{h_w}+\sqrt{\left(\frac{g_3}{h_w}\right)^2+\left(\frac{g_1}{h_w}\right)^2\left(\frac{I_3}{I_1}\right)}\sin(\Omega_p t+\Phi_1) \qquad (5.49a)$$

$$\omega_3(t)=\frac{-g_1}{h_w}+\sqrt{\left(\frac{g_1}{h_w}\right)^2+\left(\frac{g_3}{h_w}\right)^2\left(\frac{I_1}{I_3}\right)}\sin(\Omega_p t+\Phi_3) \qquad (5.49b)$$

where: $\Omega_p=\frac{h_w}{\sqrt{I_1 I_3}}=$ nutation frequency

$\Phi_{1,2}$ are relative phase offsets

Let's use the current design mass properties and h_w for argument's sake:

$$I_1/I_3 = 1.192$$
$$h_w = 1.98 \ N\bullet m\bullet s$$

From Chapter 3, a worst case net disturbance torque assumption of $2x10^{-5}$ N•m along each axis seems defensible ("net" meaning the sum of all the low frequency spectral components which look like DC from 3 decades up). Assume time t such that $\sin(\Omega_p t+\Phi_3) = -1$, equation (5.49b) gives:

$$|\omega_3| = 5.2 \ °/hr$$

This is of course a fairly rough worst case estimate, but it is certainly in the range where one can feel comfortable about meeting a 20°/hr rate specification. The fact that the orbit night controller, designed with these values, meets its specifications completes the justification for choosing this h_w.

Based on these results, the orbit day control strategy chosen is summarized:

• A spin-axis controller decoupled from the spin-plane controller is designed to be used throughout the day to null the azimuth error λ_1 and maintain ω_2 to within specifications.

• A spin-plane controller using only the spin-axis coil m_2 is designed such that:

- When the field is good ("good" and "bad" are defined in detail in the next section), use it to advantage by nulling the elevation error λ_2 and damping rates.

- When the field is bad, the controller is simply turned off. The build up of elevation error during this time is insignificant, and the rates induced are tolerable, as shown.

Why confine the spin-plane controller to using the spin-axis coil only (i.e. actuator mode 2)? Because this reduces the torque demand on the wheel to only that required by the spin-axis controller, which is desirable for the sake of wheel momentum management. The looser specifications during orbit day do not justify the use of a full-authority actuator mode controller.

5.3.2 Controller Design

We begin by putting the design plant model of Section 5.1 into forms specific to this section. Recall the definitions in Section 4.2.2 for the desired azimuth a_0 of the sun-vector (assume the desired elevation $e_0 = 0$). Equations are stated general in a_0, although a_0 is non-zero only during a slew maneuver. The azimuth/elevation errors from the processed sun sensor measurements are assumed available for feedback. These relate to the quaternions (kinematic parameters) according to the measurement matrix in (5.9):

$$\lambda_1 = 2\varepsilon_2 \qquad (5.50a)$$

$$\lambda_2 = 2(\cos(a_0)\varepsilon_1 - \sin(a_0)\varepsilon_3) \qquad (5.50b)$$

Unlike the CCD camera attitude outputs, which are inherently discrete time signals computed from an image buffer every 4 seconds, the actual sun sensor outputs (quad-cell current differences) are analog, making it necessary to have a pre-filter somewhere in the loop. The control loop sample rate is assumed to be 1 Hz for the orbit day, making the Nyquist rate π rad/s. We choose, for a conservative design estimate, a pre-filter on each of the feedback quantities in (5.50) of a single pole at $\omega_{pf} = 0.75$ rad/s, which is about $\pi/4$. This is in all likelihood a lower bandwidth than necessary- the sun sensor noise spectrum shouldn't have much power at this low a frequency- but the effect of it on controller

performance is still negligible, as will be seen. Note also that a filter of this low bandwidth is difficult to implement with analog components. It would make sense to sample the actual sun-sensor at a higher integral multiple of 1 Hz using appropriate analog prefilters, then implement the prefilters at ω_{pf} digitally in front of the calculations of λ_1 and λ_2.

The pre-filters are taken into account in the design plant models. The spin-axis plant is the familiar double integrator:

$$\frac{\lambda_1(s)}{\tau_w(s)} = \left(\frac{1}{I_2 s^2}\right)\left(\frac{\omega_{pf}}{s + \omega_{pf}}\right) \tag{5.51}$$

The spin-plane plant is reduced to a minimal (i.e. with the pole-zero cancellation associated with the unobservability of rotation about the sun-sensor boresight removed) SISO realization, first in state space, then in transfer function form. Again the pre-filter is assumed to have a DC gain = 1; its state is "f" below:

$$\frac{d}{dt}\begin{bmatrix} \omega_1 \\ \omega_3 \\ \lambda_2 \\ f \end{bmatrix} = \begin{bmatrix} 0 & \frac{h_w}{I_1} & 0 & 0 \\ \frac{-h_w}{I_3} & 0 & 0 & 0 \\ \cos(a_0) & -\sin(a_0) & 0 & 0 \\ 0 & 0 & \omega_{pf} & -\omega_{pf} \end{bmatrix}\begin{bmatrix} \omega_1 \\ \omega_3 \\ \lambda_2 \\ f \end{bmatrix} + \begin{bmatrix} \frac{1}{I_1} & 0 \\ 0 & \frac{1}{I_3} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} \cos(\theta) \\ -\sin(\theta) \end{bmatrix}\left(m_2\sqrt{\beta_1^2 + \beta_3^2}\right) \tag{5.52}$$

Define the following for convenience:

$$\underline{x}_R \equiv \begin{bmatrix} \omega_1 \\ \omega_3 \\ \lambda_2 \\ f \end{bmatrix} \qquad \tau_{tn} \equiv \left(m_2\sqrt{\beta_1^2 + \beta_3^2}\right) \tag{5.53}$$

$$A_R \equiv \begin{bmatrix} 0 & \dfrac{h_w}{I_1} & 0 & 0 \\ \dfrac{-h_w}{I_3} & 0 & 0 & 0 \\ \cos(a_o) & -\sin(a_o) & 0 & 0 \\ 0 & 0 & \omega_{pf} & -\omega_{pf} \end{bmatrix} \qquad (5.54)$$

$$B_{R_0} \equiv \begin{bmatrix} \dfrac{1}{I_1} & 0 \\ 0 & \dfrac{1}{I_3} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \qquad B_R \equiv B_{R_0}\begin{bmatrix} \cos(\theta) \\ -\sin(\theta) \end{bmatrix} \qquad (5.55)$$

The plant model output is the prefilter state:

$$f = [0 \quad 0 \quad 0 \quad 1]\underline{x}_R \equiv C_R\underline{x}_R \qquad (5.56)$$

The transfer function is then, after some algebra:

$$\frac{\lambda_2(s)}{\tau_{tn}(s)} = C_R(sI - A_R)^{-1}B_R$$

$$= \left(\frac{s(I_1\sin a_o \sin\theta + I_3\cos a_o \cos\theta) + h_w(\cos\theta\sin a_o - \sin\theta\cos a_o)}{s\left(I_1 I_3 s^2 + h_w^2\right)} \right)\left(\frac{\omega_{pf}}{s + \omega_{pf}} \right) \quad (5.57)$$

Some observations on this spin-plane plant :

- From (5.53), it is seen that one way the ambient field could be "bad" is when $\sqrt{\beta_1^2 + \beta_3^2}$ is too small to achieve the commanded τ_{tn} within the limitations of m_2. This is the same as having an unfavorable Ψ and has already been discussed.

- There is a pole/zero cancellation at $s = 0$ when $a_o = \theta$ or $\theta \pm \pi$. Again, recall figure 5.4 for the geometric origin of the cancellation:

- m_2 (or τ_{tn}) can control attitude about the axis along θ on the spin plane
- sun sensor cannot observe attitude about the direction a_0 on the spin plane

The frequency of the zero in the transfer function (5.57) is near 0 when a_0 nears θ or $\theta \pm \pi$, which shows the increasing difficulty to control λ_2.

It is seen that for a range of θ the plant has very low frequency non-minimum phase zeros. Its worst feature, however, is that as θ crosses a_0 or $a_0 \pm \pi$, the low frequency phase of the transfer function shifts by 180° (or, in servo terminology, the static sensitivity / velocity constant changes sign). It is noted that the design plant model for the orbit night does not suffer from the same problem because of full attitude measurement provided by the CCD cameras. For the majority of the θ range, LQG is again a convenient method for yielding a family of compensators scheduled on θ which perform well and offer good robustness to θ measurement offsets. This is not true, however, when θ nears a_0, "near" being defined according to the θ measurement error bound analysis of Section 4.2.3. Then possibility exists for the true plant's DC phase to be 180° offset from the design plant model, and the designed compensators are not robust to this effect. This comprises the other way the field could be "bad"- when θ is near a_0 and $\sqrt{\beta_1^2 + \beta_3^2}$ is too small compared to the estimate of the magnetometer biases to be confident of the side the true value of θ is on relative to a_0. When the field is "bad" in this way, the options are to:

• Design a different compensator for this θ region which is tolerant to DC phase perturbation of 180°. This compensator will surely adopt the strategy of putting a zero on top of the pole at s = 0 to prevent it from moving as the plant zero shifts across the $j\omega$ axis, in which case its only purpose is to damp the nutation poles. It's not clear that sufficient robustness to nutation pole perturbations could still be maintained, as the design would most likely be notch-filter based.

• Change the physics of the situation by employing the full-actuator authority idea used during orbit night, which eliminates the restriction that torque can only be generated along $\theta \pm 90°$ on the spin plane.

• Turn the controller off.

The last option is the least heroic but most appealing to common sense, given the conclusion of the last section that active rate damping is really not a necessity but rather a

bonus result of tracking the sun-vector, which will come no where near violating the 5° specification as a result of being uncontrolled for a few minutes. The additional complexity of the other two options is unnecessary. This completes the justification for the control strategy chosen; it remains to fill in the details of the design.

The spin-axis controller is designed in continuous time and the resulting compensator mapped to discrete time. Since the plant is fairly benevolent, and since disturbance torques enter the loop before the plant, the compensator contains an integrator along with lead compensation to generate infinite DC gain before the disturbance entry point in the loop, removing steady-state tracking error due to the disturbance. The phase lead hump and gain are tweaked to yield the following open loop system from loop error input to filtered λ_1 output:

$$\frac{\lambda_1(s)}{e(s)} = \left(\frac{K(s+z)^2}{s(s+p)} \right) \left(\frac{1}{I_2 s^2} \right) \left(\frac{\omega_{pf}}{s+\omega_{pf}} \right) \tag{5.58}$$

Design values used : $I_2 = 5.4129 \text{ Kg} \cdot \text{m}^2$ $\omega_{pf} = 0.75 \text{ rad/s}$

$z = 0.03 \text{ rad/s}$ $p = 0.75 \text{ rad/s}$

$K = \left(\frac{p}{z^2} \right) (-63 \text{ dB})$

The resulting closed loop poles in the s-plane are at:

```
-1.0242   -0.2372   -0.1084 ± 0.0463i   -0.0218
```

Figure 5.20 shows the open-loop Bode plot, where 180° has been added to the phase to facilitate reading phase and gain margins. It is seen that the system is conditionally stable (has finite up and downward gain margins), which is confirmed with a root locus sketch. The upward and downward gain margins are tweaked to be about 20 dB each, resulting in a 45° phase margin. The gain crossover is more than a decade below the Nyquist frequency, so the discrete time version of the controller is not expected to lose much phase margin. It is easiest to implement the discrete time version of the compensator of (5.58) in state space form. The zeroth-order hold transformation is used again. However, the presence of the feedthrough term (2 zeros, 2 poles) causes the DT compensator's gain to be slightly high past 0.1 rad/s. Turning the design DC gain down 5 dB improves the gain margins.
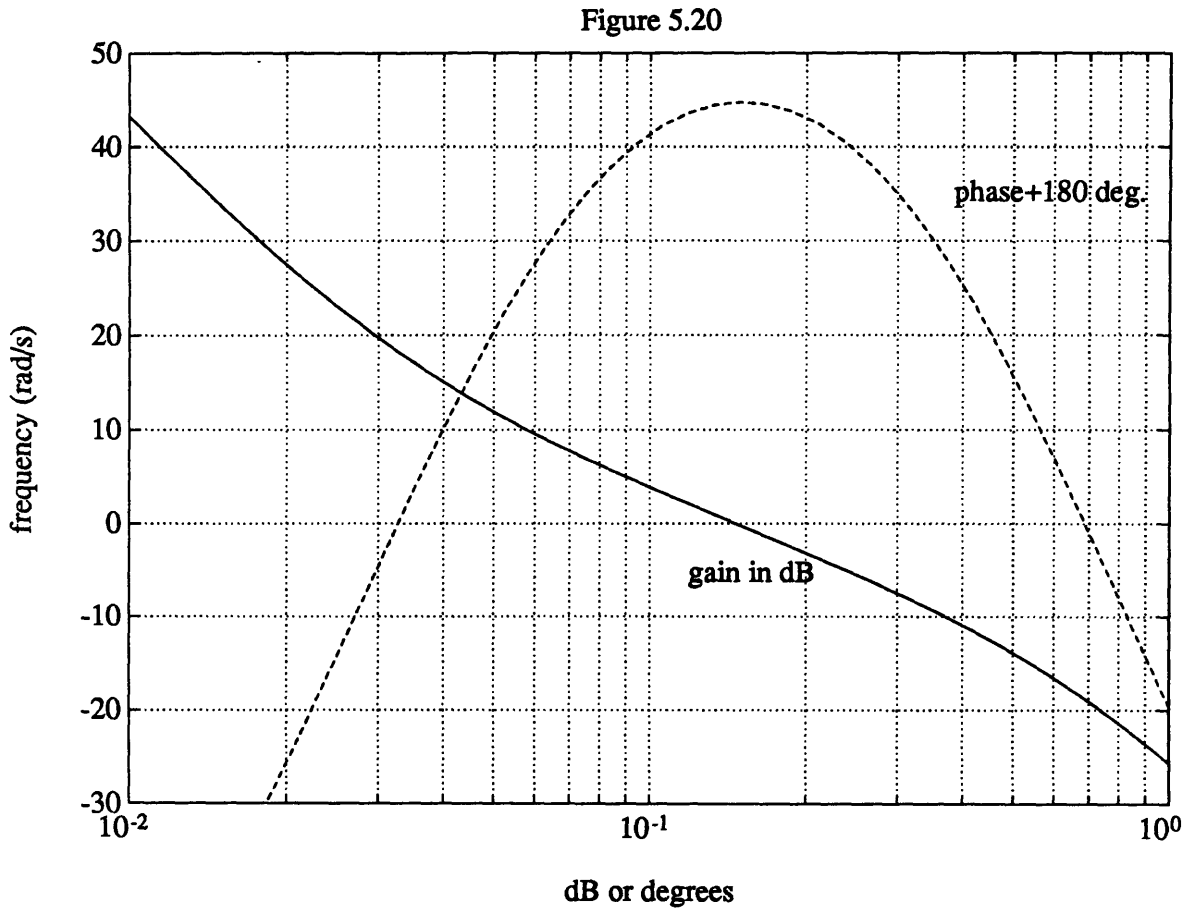
Figure 5.20

Figure 5.20: spin-axis controller Bode plot; solid = magnitude in dB, dash = phase + 180°. The margins are thus read relative to phase = 0, making the plot scaling easier.

The spin plane controller designed here is for a desired sun-vector azimuth $a_0 = 0$. Even though the design problem is SISO, the LQG compensator structure is chosen because it offers an easy way to schedule a family of compensators, with shifting pole/zero locations, to the range of θ by simply changing the gain matrix elements. The compensators were initially designed in continuous time, but it was found that using the same filter and regulator gains in the discrete time compensator, with the design plant models properly transformed, resulted in a controller whose frequency response matches that of the CT version well past the crossover frequencies for all θ. Thus the same gains are used in the DT implementation.

The spin plane design plant model (5.54) thru (5.56) are discretized at a 1 second

sampling period using the zero-th order hold technique described previously. Numerical values for the matrices are given below:

$$A_R =$$

0	0.4073	0	0
-0.4854	0	0	0
1.0000	0	0	0
0	0	0.7500	-0.7500

$$\Phi_R =$$

0.9028	0.3940	0	0
-0.4696	0.9028	0	0
0.9674	0.2003	1.0000	0
0.2912	0.0422	0.5276	0.4724

$$B_{Ro} =$$

0.2058	0
0	0.2453
0	0
0	0

$$\Gamma_{Ro} =$$

0.1991	0.0491
-0.0491	0.2373
0.1012	0.0165
0.0213	0.0027

C_R is as in (5.56). The filter design parameters used are:

- process noise intensity: $E\left[w(t)w(t)^T\right] = 10[I_{2\times2}]$

- measurement noise intensity: $E\left[v(t)v(t)^T\right] = 1$

Process noise $w[n]$ is interpreted as a disturbance torque, so its distribution matrix is B_{Ro} from (5.55). Again, the weights are chosen to obtain closed loop filter poles somewhat faster than closed loop regulator poles. Numerical values are given for the resulting filter gain matrix H, which is of dimension 4x1 since the innovation is a scalar:

$$H =$$
$$0.9133$$
$$-0.3346$$
$$2.2869$$
$$1.2482$$

The regulator design cost function minimized is:

$$J = \int_0^\infty \omega_1^2 + \omega_3^2 + 0.01\lambda_2^2 + \tau_{tn}^2 \, dt \qquad (5.59)$$

It is noted that the prefilter state f is not weighted in the cost function. Therefore, the compensator will have a zero at s = -ω_{pf} to cancel the pole. This is fine, as the ω_{pf} is of sufficiently high frequency compared to the dominant closed loop poles to make moving it in the closed loop unnecessary. A family of gains are calculated using as design plant models (5.54) thru (5.56), where (5.55) varies with θ.

The compensator implementation equations are the familiar DT LQG structure again, with the appropriate DPM:

- previous compensator state: $\bar{x}[n]$

- error signal : $e = r - \lambda_2[n]$ (r = servo command = 0)   (5.60)

- current estimate: $\hat{x}[n] = \bar{x}[n] - H(e + C_R\bar{x}[n])$ (5.61)

- compensator output: $\tau_{tn}[n] = -G(\theta)\hat{x}$ (5.62)

- prediction step: $\bar{x}[n+1] = \Phi_R\hat{x}[n] + \Gamma_{Ro}\begin{bmatrix} \cos(\theta) \\ -\sin(\theta) \end{bmatrix}\tau_{tn}[n]$ (5.63)

The regulator control law (5.62) is expanded to define its individual elements:

$$\tau_{tn}[n] = g_{11}(\theta)\omega_1[n] + g_{12}(\theta)\omega_2[n] + g_{13}(\theta)\lambda_2[n] \qquad (5.64)$$

These are plotted in Figure 5.21 as functions of θ. Note the discontinuities at $\theta = 0$ and $180°$, as expected. Since the correct gain to use is computed on line from measurements of θ, that the sign of g_{12} and g_{13} change across the discontinuities is demonstration of the assertion made earlier that the design is not robust to θ measurement offsets which may cross the boundaries 0 and $180°$ (i.e. a_0 and $a_0 \pm \pi$). Tolerance to θ measurement errors in other regions of θ should be reasonably good, as the gain curves are seen to be smooth. The gain

curves can be piece-wise fitted with either polynomials or sinusoids to reduce on-line computation greatly; this is the approach used in simulations in this thesis.The code to compute the gains and fit them to polynomials is given in Appendix A-5e.
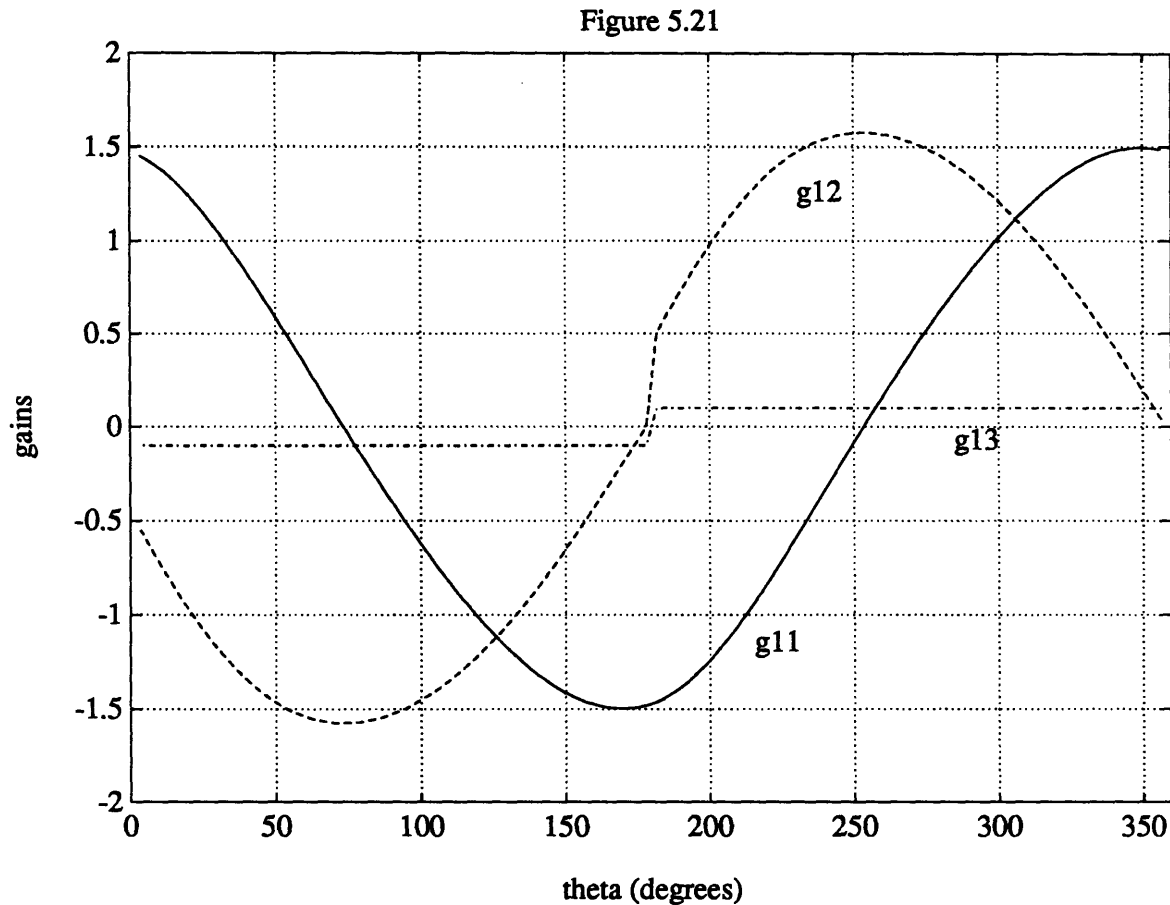
Figure 5.21: gain scheduling curves for day spin-plane controller, $a_0 = 0$.

The family of achieved closed loop poles are plotted as functions of θ in Figure 5.22a, mapped to the s plane via s = Log(z)/T (recall T =1 sec) for insight. Again the separation principle applies, allowing separation of filter and regulator closed loop poles according to equations (5.18) and (5.19). Both sets are plotted in Figure 5.22. It is seen that as the plant's zero (from transfer function in (5.57)) drops in frequency as θ approaches 0 or 180°, the compensator is less able to move the open loop λ_2 pole at s = 0 while satisfying the constraint on actuator activity imposed by the cost function, as expected from the physical arguments given previously.

For completeness, the open loop and closed loop frequency responses of the spin-plane controller are plotted over the full range of θ (except at θ = 0, 180°) in Figures 5.22 b thru d.
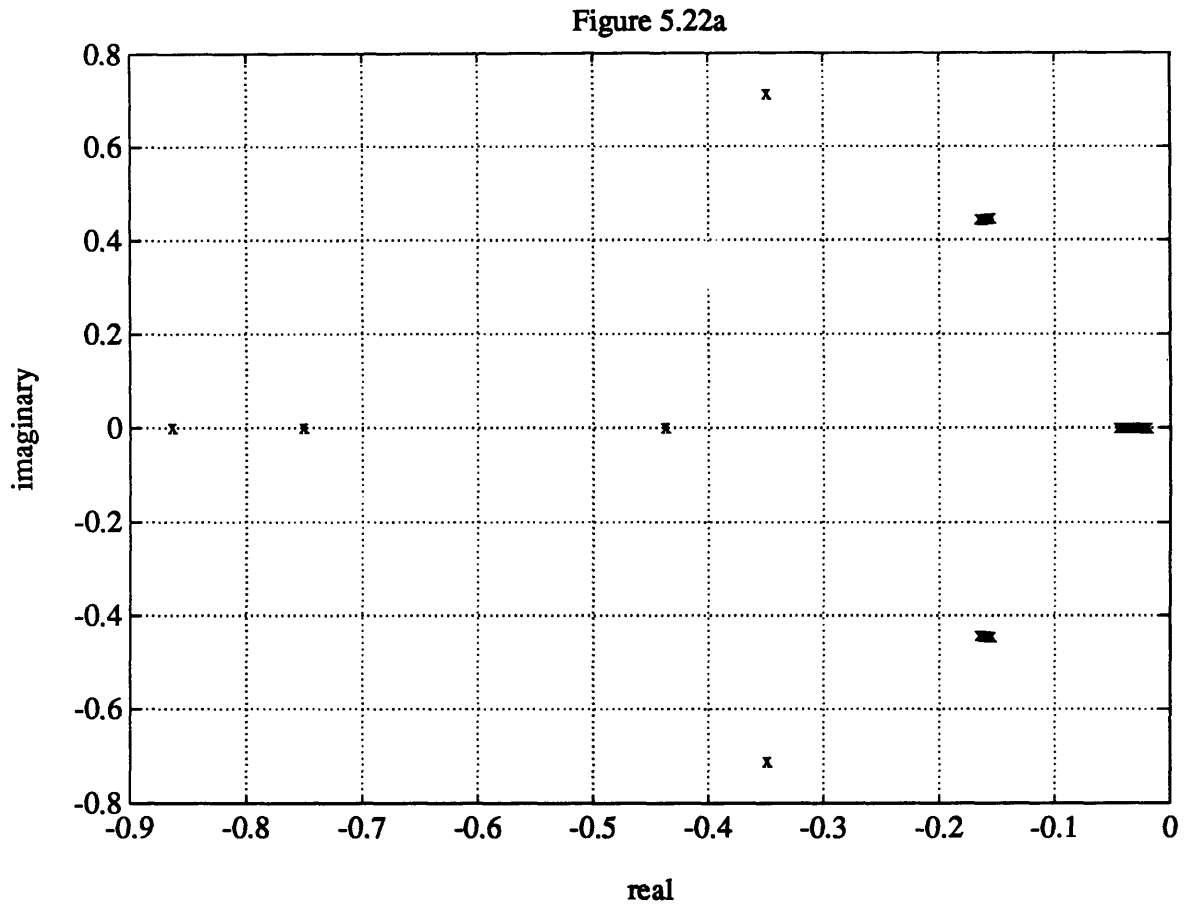
Figure 5.22a

Figure 5.22a: spin-plane controller closed loop poles as functions of θ. The poles are transformed to the s-plane from the z-plane.

Figure 5.22b
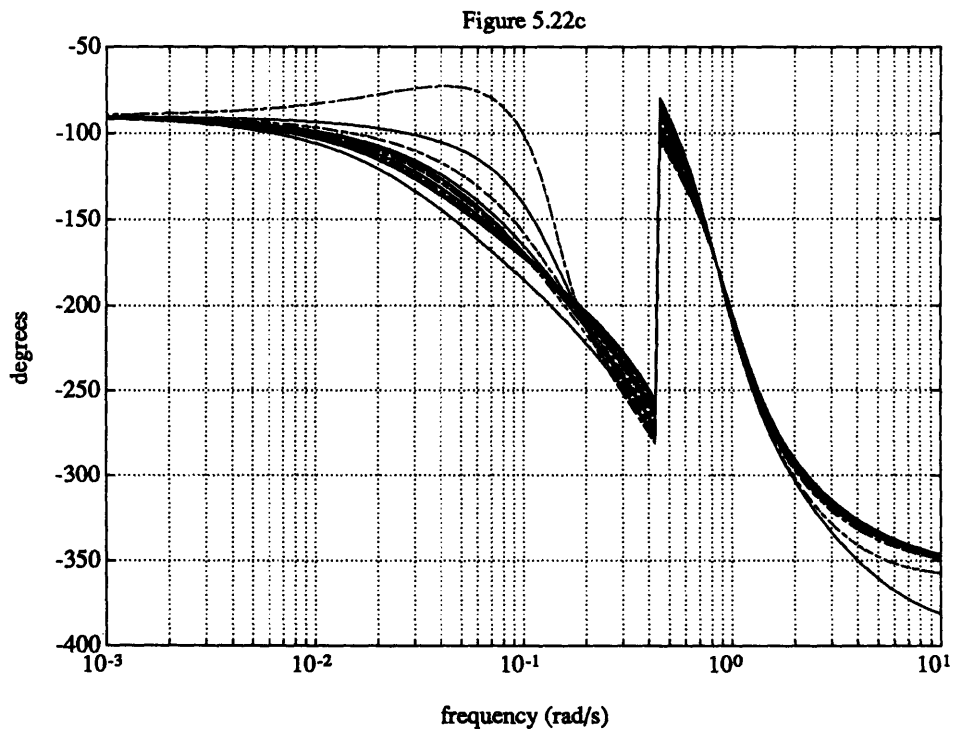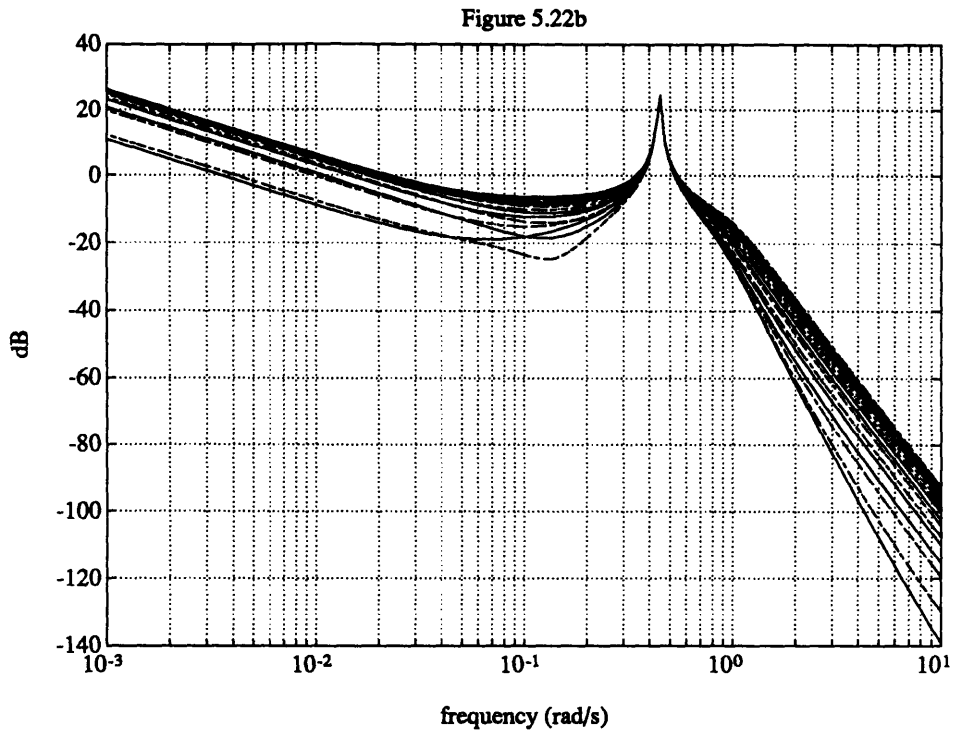
Figure 5.22c

Figure 5.22 b & c: open loop Bode plots of the compensator+plant system for the full range of θ (except θ = 0, 180°) in 10° steps.
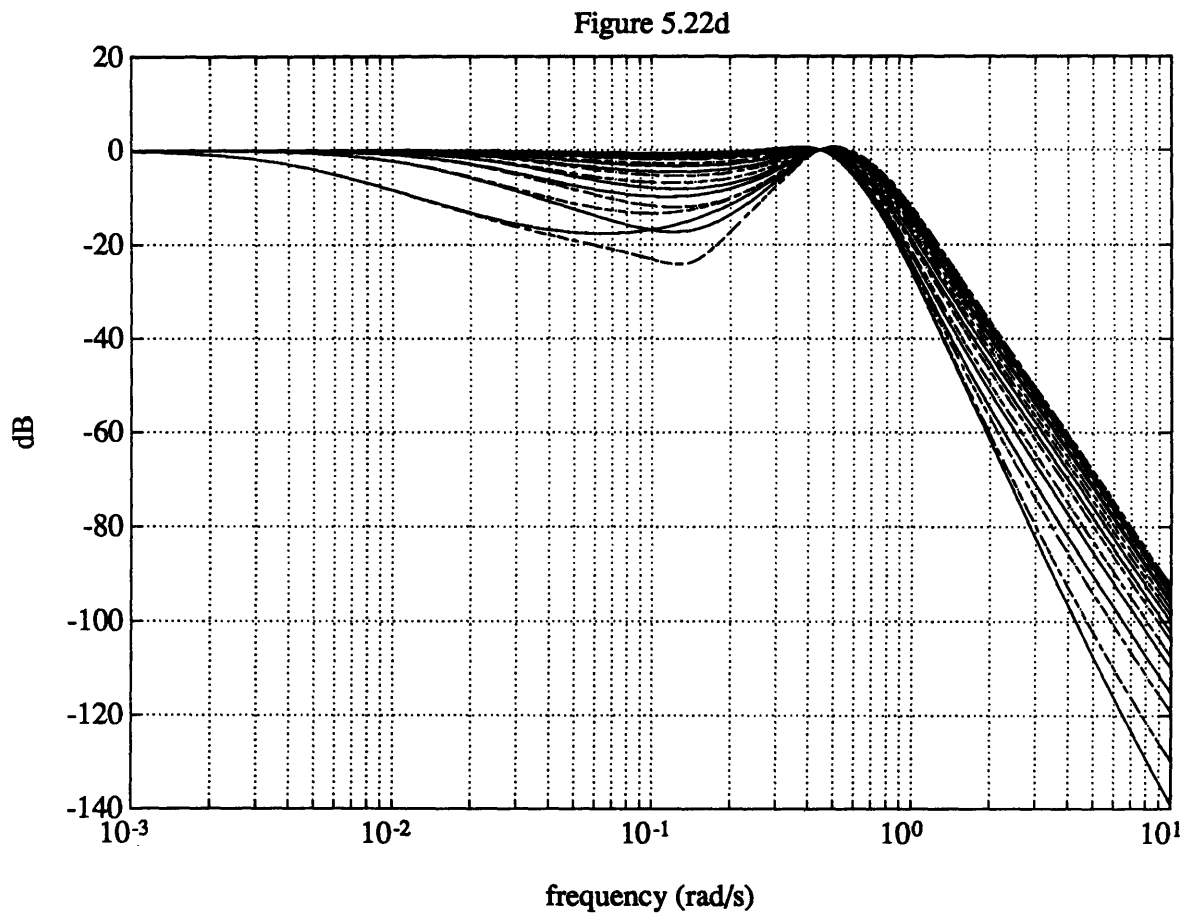
Figure 5.22d

Figure 5.22d: closed loop Bode magnitude plot of the spin-plane controller over the same range of θ as in figure 5.22 b &c. The poorer response side of the band corresponds to θ nearing 0 or 180°, as expected from the poor controllability of the plant.

5.3.3 Stability Robustness

As with the orbit night case, only robustness to structured parameter variations is considered, as no unmodeled dynamics are expected in the control bandwidth. Also, it is again assumed for the gain scheduled spin-plane controller that θ varies sufficiently slowly compared with the control bandwidth that linear time-invariant stability criteria are used.

The robustness of the spin-axis controller has already been discussed. It is seen from Figure 5.20 that about 20 dB of upward and downward gain margins exist, and that the phase margin of 45° is not compromised much by the discrete time implementation because the gain crossover is 1/15 th of the Nyquist frequency (0.5*360/15 = 12°). These are sufficient.

The spin-plane controller robustness to parameter variations are evaluated again by z-plane root loci. The following combinations of inertia, wheel momentum, and θ measurement error are analyzed:

Case #:	parameter perturbations:				nutation freq. change:
	I_1:	I_3:	h_w:	θ:	
1	0	0	0	0	none
2	-20%	-20%	+10%	0	up 37.5%
3	-20%	-20%	+10%	-20°	up 37.5%
4	-20%	-20%	+10%	+20°	up 37.5 %
5	+20%	+20%	-10%	0	down 25%
6	+20%	+20%	-10%	-20°	down 25%
7	+20%	+20%	-10%	+20°	down 25%

Recall that the plan is to shut off the controller when θ comes close to 0 or 180°. Since the θ offset is here modelled to be 20°, the root loci below are generated for θ's not in the range of 340° ~ 20°, 160° ~ 200°. In the actual implementation, the "forbidden" θ zone is determined on-line from an estimate of the total magnetometer bias magnitude and the θ error bound analysis of Section 4.2.3.

The root loci are presented in the z-plane. The system order is 8, but the plot is scaled such that the quadrants where the nutation poles come closest to the unit circle are emphasized- one of the filter closed loop poles is thus not shown. This is not significant.

Figure 5.23
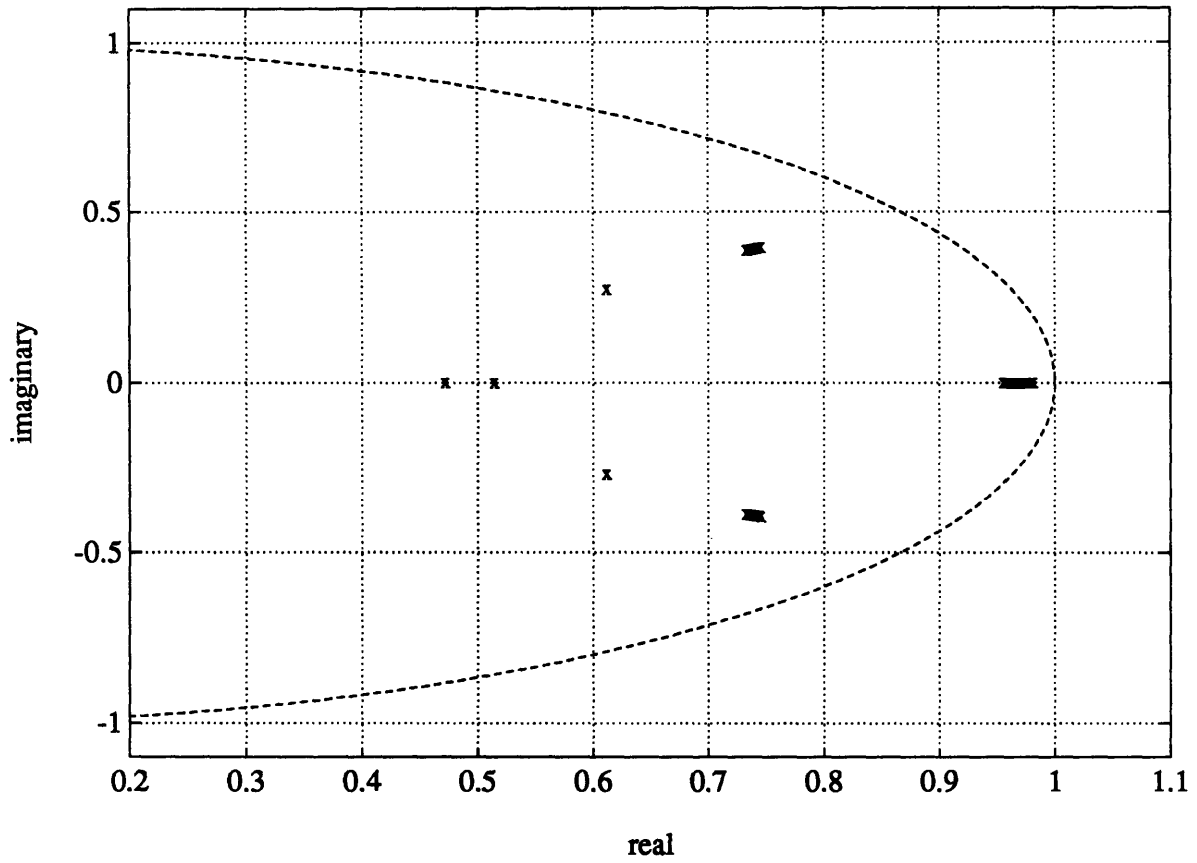
Figure 5.23: case #1: no plant perturbations. This is the z-plane version of figure 5.22, here for reference. The dashed oval is the unit circle on a scale distorted for closeup clarity.

Figure 5.24
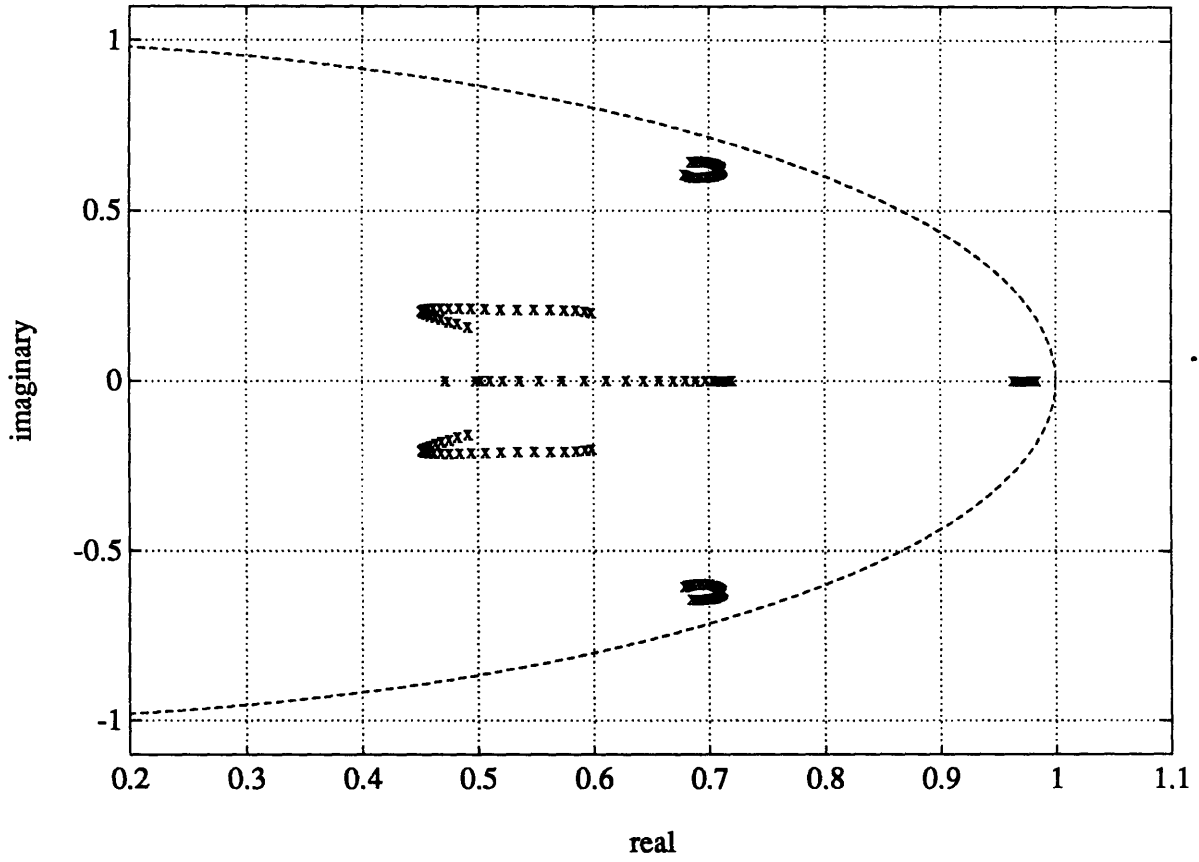
Figure 5.24: day spin-plane controller closed loop root locus vs. θ, plant perturbation case #2.

Figure 5.25
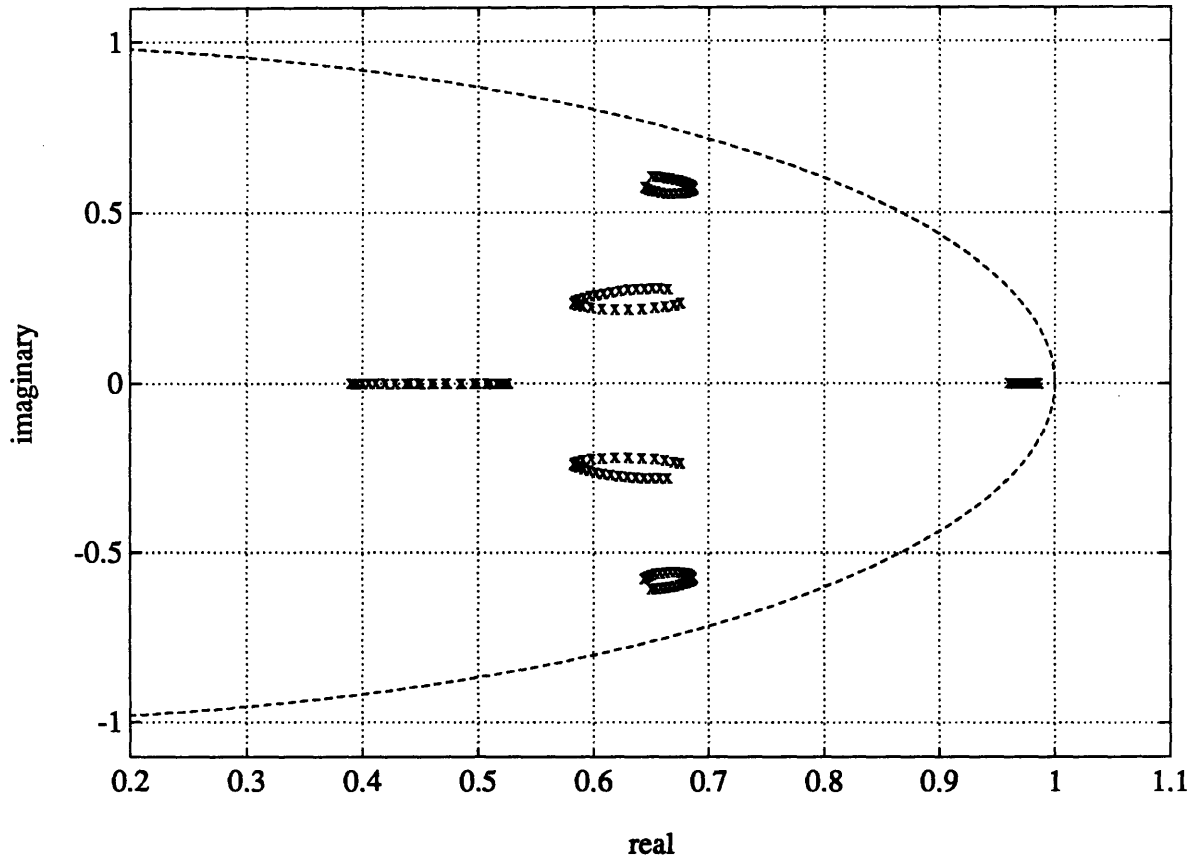
Figure 5.25: day spin-plane controller closed loop root locus vs. θ, plant perturbation case #3.

Figure 5.26
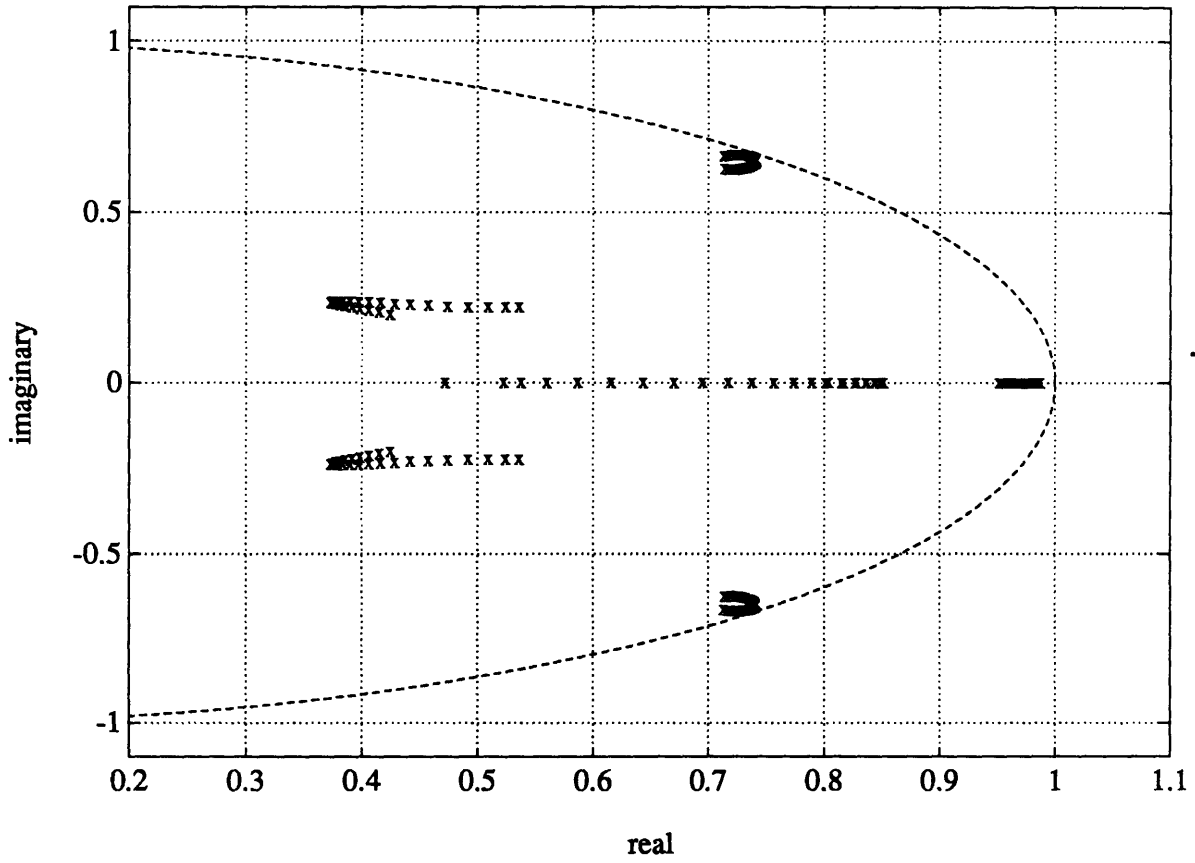
Figure 5.26: day spin-plane controller closed loop root locus vs. θ, plant perturbation case #4. This is apparently the most severe case.

Figure 5.27
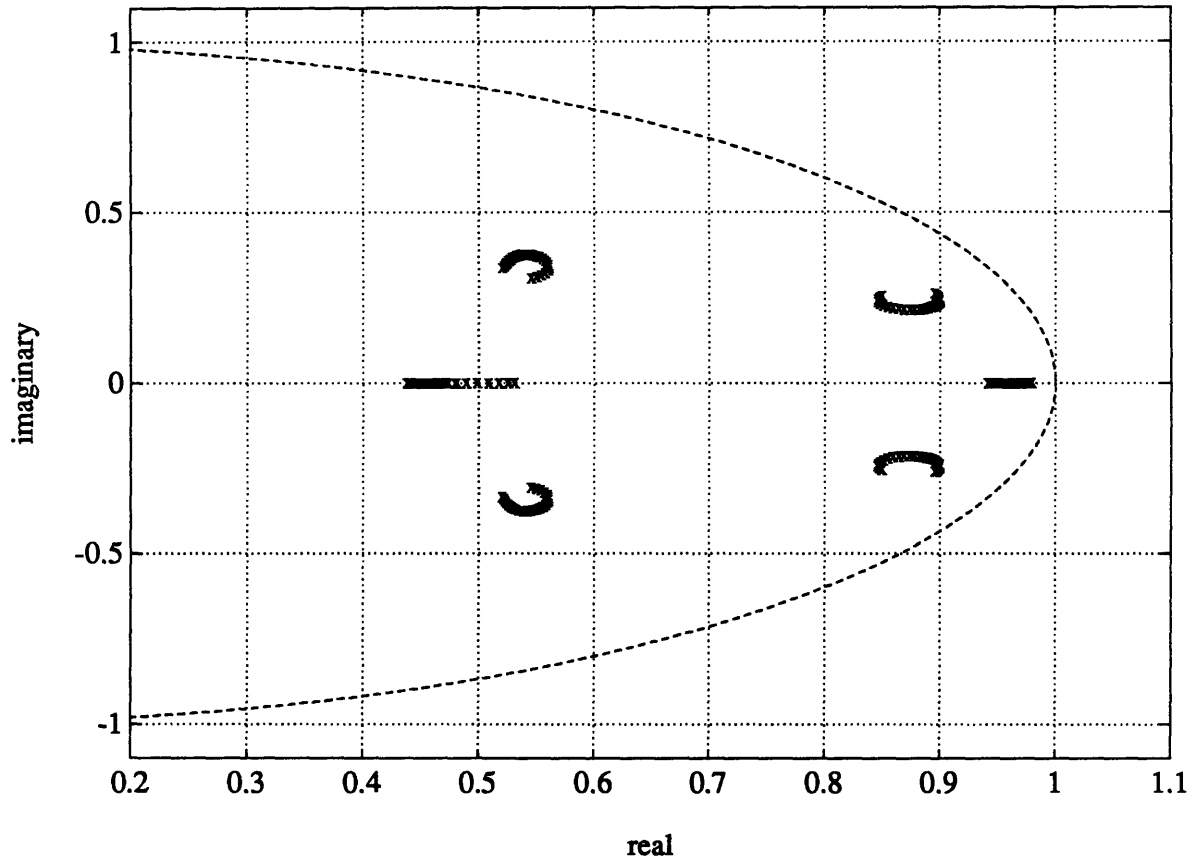
Figure 5.27: day spin-plane controller closed loop root locus vs. θ, plant perturbation case #5

Figure 5.28
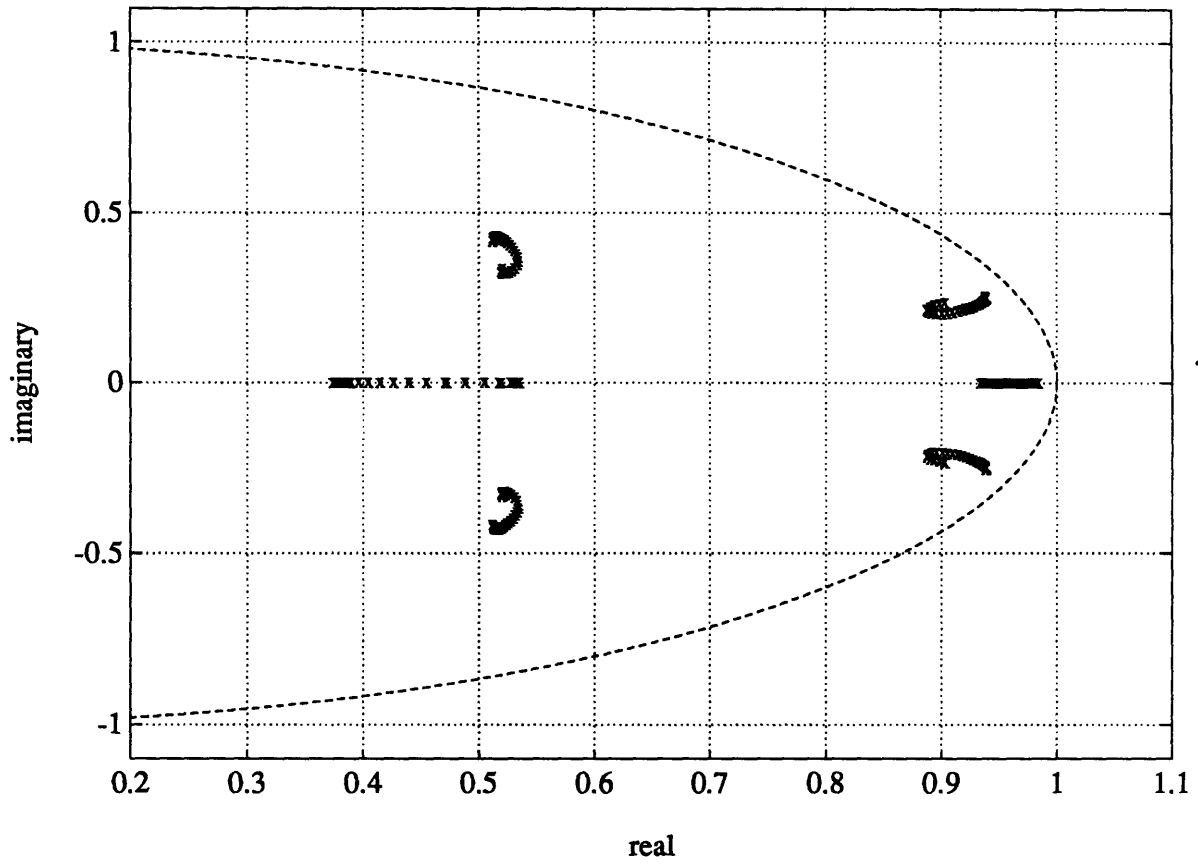
Figure 5.28: day spin-plane controller closed loop root locus vs. θ, plant perturbation case #6
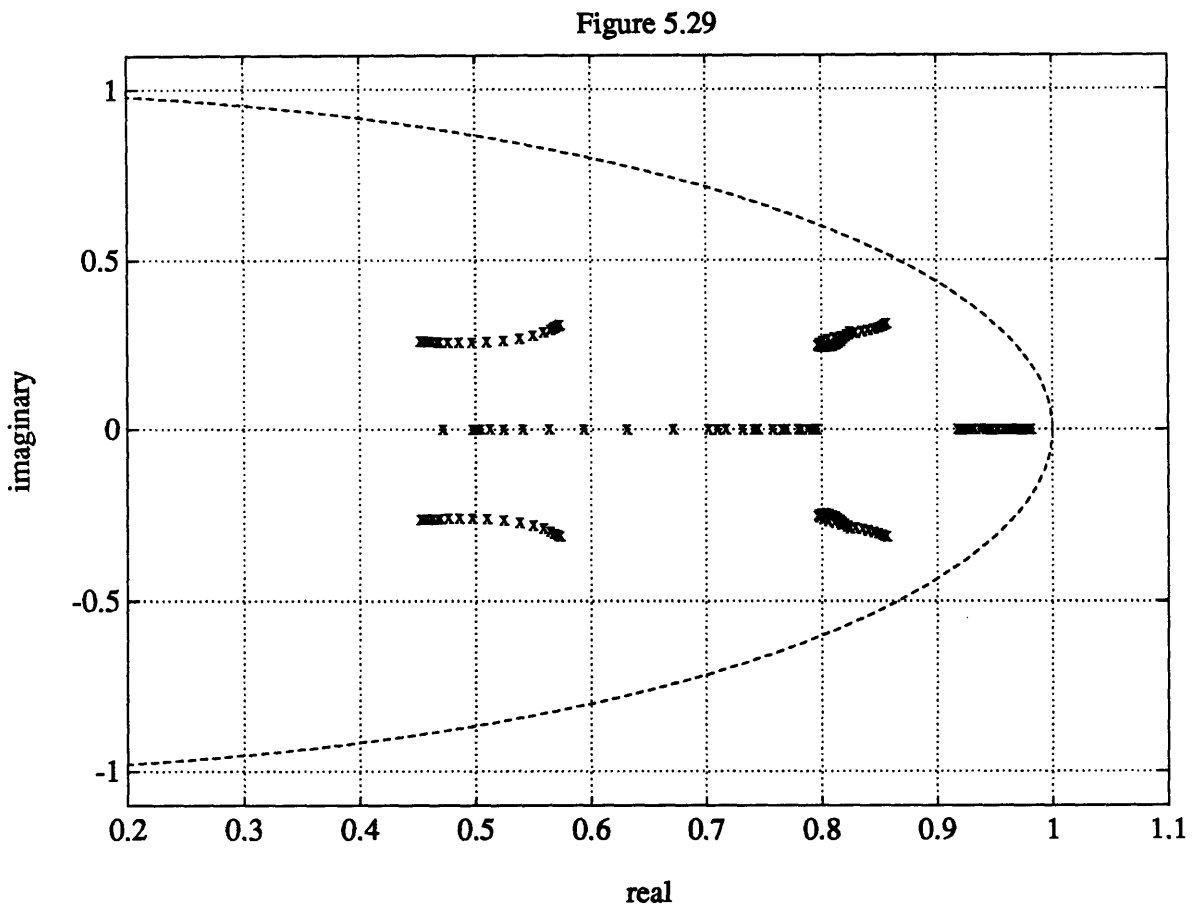
Figure 5.29

Figure 5.29: day spin-plane controller closed loop root locus vs. θ, plant perturbation case #7.

It is again observed that the controller is less tolerant to the actual plant nutation poles being at a higher frequency than lower. Therefore it is better to underestimate than overestimate the spacecraft inertias when finalizing the compensator design for implementation. Overall, the day controller is less robust than its counterpart for orbit night, and the results of this section can be seen as approximate specifications on the accuracy of inertia measurements and magnetometer calibration. It is also possible to

gain-schedule according to wheel angular momentum as it changes, but this should not be necessary on a regular operational basis because the plan is to desaturate during every orbit day to maintain h_w close to its nominal value. 10% of h_w = 1.98 N•m•s divided over an orbit (approximately 5700 sec.) is 3.5 N•m, which should be several time the average torque demand on the wheel. Furthermore, the worst case of 20% offset in the measurement of I_1 and I_3 simultaneously is very conservative, given that 10% or better is achievable. [Pers91] Thus the preliminary conclusion is that the control strategy and design presented here is acceptable from a stability robustness point of view. Code for the calculations of this section is given in Appendix A-5f.

5.3.4 Actuator and Sun-Sensor Quantization Effects on Performance:

The same analysis technique used in Section 5.2.5 is applied here to find the RMS response of body rates to quantization effects, with the additional corruption of the feedback path by sun-sensor quantization considered. Figure 5.30 show a generic block diagram which represents either the spin-axis control loop or spin-plane control loop. Note the loop is SISO in either case. As before, the actuator quantization is abstracted to a disturbance torque. The sun-sensor discretization is represented as an output disturbance, with the quantization step discussed in Chapter 2.
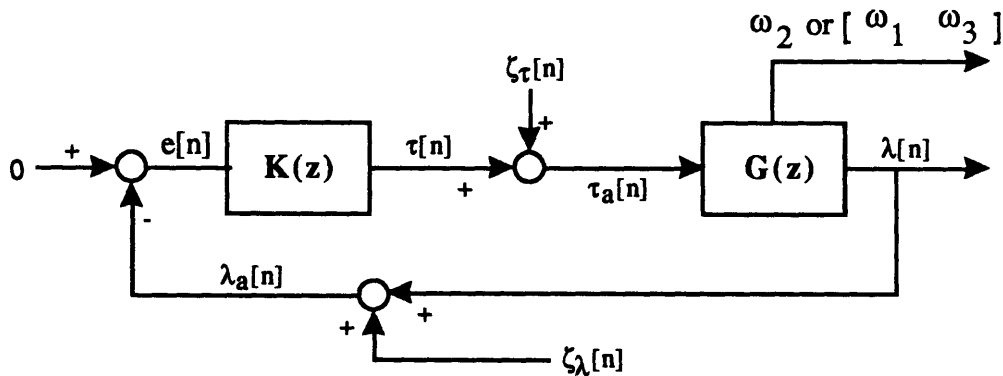
Figure 5.30: generic block diagram defining actuator and sun sensor quantization noises. The loop represents either the spin-axis or the spin-plane control loop, with K(z) being the appropriate compensator and G(z) being the appropriate plant model. We are interested in the RMS output of the ω's as functions of the ζ's.

The quantization noises $\zeta_\tau[n]$ and $\zeta_\lambda[n]$ are such that $\tau_a[n]$ and $\lambda_a[n]$ are the actual quantized signals. The assumption is therefore once again made that $\zeta_\tau[n]$ and $\zeta_\lambda[n]$ are at

each time step n governed by uniform PDF's and are zero-mean white noises. Since they are zero-mean scalars, their variances at n, the intensities of the white noises, are written:

$$\text{var}(\zeta_\tau[n]) = E\left(\zeta_\tau^2[n]\right) = \frac{q_\tau^2}{12} \tag{5.65a}$$

$$\text{var}(\zeta_\lambda[n]) = E\left(\zeta_\lambda^2[n]\right) = \frac{q_\lambda^2}{12} \tag{5.65b}$$

where q_τ and q_λ are the actuator and sensor quantization steps, respectively. The closed loop matrices can then be generated with $\zeta_\tau[n]$ and $\zeta_\lambda[n]$ as inputs to find the RMS response of ω_2 or $[\omega_1 \ \omega_3]$ using steady state Lyapunov analysis as before. The intervening algebra is familiar and is therefore skipped. Results are:

Spin-Axis Loop:

- wheel torque quantization step $q_\tau = 5 \ \mu\text{N} \cdot \text{m}$ (as night analysis)

- sun sensor output λ_1 quantization step $q_\lambda = (10/1024) \cdot (\pi/180)$ radians
  (this corresponds to $\pm 5°$ view window with a 10 bit ADC)

- RMS value of $\omega_2 = 1.46 \ °/\text{hr}$

Spin-Plane Loop:

- magnetic torque quantization step $q_\tau = 0.625 \ \mu\text{N} \cdot \text{m}$ (as in night analysis)

- sun sensor output λ_2 quantization step $q_\lambda = (10/1024) \cdot (\pi/180)$ radians
  (this corresponds to $\pm 5°$ view window with a 10 bit ADC)

- RMS($[\omega_1 \ \omega_3]$) output are functions of θ and are plotted in Figure 5.31.

The results derived here again match observed simulation results reasonably well, with peak rates observed to be about twice to three times the RMS value. It is seen by comparison with the night analysis that the sun-sensor quantization has a major effect on the output. It is also interesting to note the considerably increased sensitivity of the spin-plane loop to sun sensor quantization. During design, the surprising result was observed that the sensitivity worsened as the bandwidth of the filter loop is adjusted down, contrary to intuition. The RMS rates from this analysis do not come very close to violating the

specifications, but the caveat that they are optimistic estimates based on ideal plant models is again mentioned. Nevertheless, 10 bit ADC resolution on a $\pm 5°$ cone sun-sensor seems to be acceptable. The code for this analysis is given in Appendix A-5g.
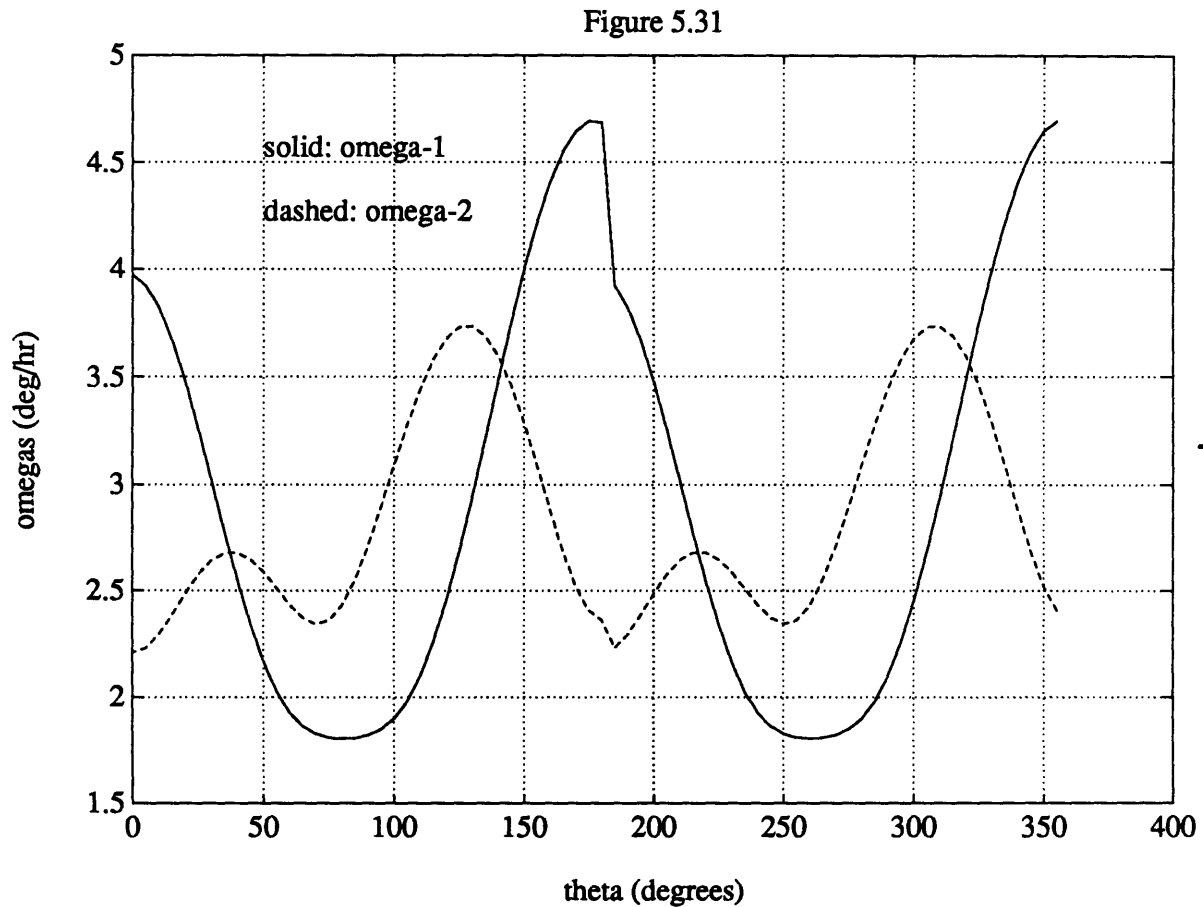
Figure 5.31

Figure 5.31: RMS([ω_1 ω_3]) from quantization noise input to spin-plane loop as functions of θ.

5.3.5 Other Implementation Issues

The anti-windup strategy discussed in Section 5.2.6 is again employed with success in the day.

A slight variation on the "initial value intercept" strategy of Section 5.2.7 is also used in both the spin-axis and spin-plane loops to reduce the large transients and actuator activity and/or saturations associated with large initial errors in sun-vector tracking.

Unlike the night, the day attitude measurements have an absolute reference- the sun vector. Therefore, the approach of redefining the null attitude is replace by essentially filtering the initial values of the sensor signals. It is important to distinguish this from filtering the sensor output itself- which would affect the closed loop dynamics unfavorably. Only the initial error is filtered, in exactly the same way that servo commands are often filtered in servo loops to remove commands with high frequency content. It is noted, of course, that no actuator mode switching is used in the day. However, discontinuities exist for both loops at the time of transition from orbit night to day, and for the spin-plane loop upon exiting from control loop shut-down during a period of unfavorable θ. Sensor initial value taring is employed in these instances. The implementation is quite simple and is illustrated in Figure 5.32.
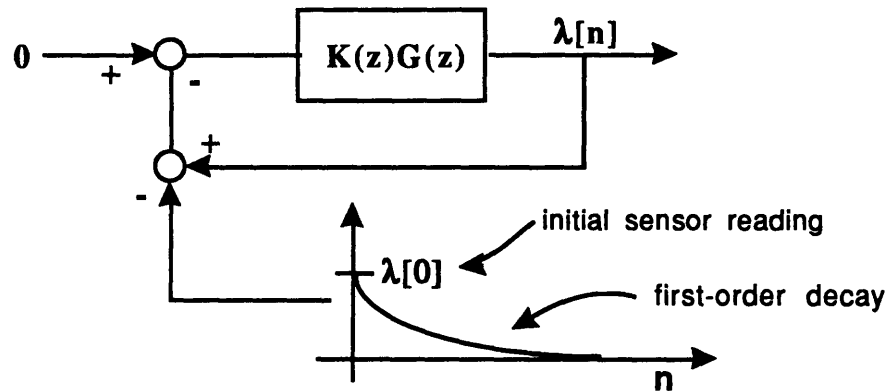
Figure 5.32: implementation of the initial value taring scheme to ease transients and actuator saturation during control loop switch-on. The initial measurement λ[0] is intercepted in software, and a decaying tare signal is set up to ease the transition. The block diagram represents either the spin axis or the spin-plane loop.

This trick is again found to be essential in preventing long and protracted coil saturations due to the physical inability to follow essentially a step attitude command of more than about 10^{-2} degrees within the closed loop time constants without saturation. The alternative of turning the bandwidth down to uselessly low frequencies to prevent saturation is unattractive.

As stated in the caption, simple first order decays for the tare signals are found to be sufficient. In the simulations, the decay constant of the tare signal for the spin-axis loop is fixed to be 95% down in 180 seconds. The spin-plane loop is considerably more prone to saturations and, of course, it is considerably more difficult to have to precess the angular momentum vector of the wheel also. Therefore, the tare signal decay constant is much

longer and is calculated based on the magnitude of $\lambda_2[0]$ itself. The formula found to work well in simulation is a 95% decay time of 1000 seconds for every 0.4 degrees of $|\lambda_2[0]|$, with a minimum of 500 seconds.

5.3.6 Momentum Wheel Desaturation

Wheel desaturation takes place during orbit day only, as platform stability is crucial during orbit night. The term "desaturation" is slightly misleading in that there is no intention to permit h_w to build up to more than a few percent off the nominal $I_w\Omega_0$. Therefore, the physical maximum wheel speed will not come close to being violated. The plan is to turn on a desaturation loop frequently. How frequently? The arguments for desaturation as often as possible (every orbit day) include:

• No expendables are used for the operation.

• The mission profile itself is divided neatly into observation during orbit night and housekeeping during orbit day. Thus, desaturation during the day does not constitute a disruption of the mission.

• Controller performance is always best when the vehicle momentum is close to the design value.

• The coils themselves have a fairly limited momentum removal rate capacity. It would be pointless to allow momentum buildup to levels comparable to or past that which can be removed in one orbit day.

The one counter-argument is:

• Recall from Chapter 3 that the disturbance torque environment spectrum contains little energy between 1 orbit rate and DC. Roughly, it is response to DC disturbance components which accumulates on the wheel. Wheel speed deviation as a function of time $\delta\Omega(t)$ will contain AC components which average to zero. The spectrum of $\delta\Omega(t)$ roughly resembles that of the disturbances. However, the bandwidth of the desaturation loop must be higher than the frequency of the very large non-linearity which turning it on and off every half orbit constitutes. Therefore, it corrects for those AC components of $\delta\Omega(t)$ with frequency at or lower than twice orbit rate, which is unnecessary. The efficiency improves as the number of orbits between desaturation increases.

However, the inefficiency of desaturating every day[1] is not drastic and, again, no expendables are involved. Therefore, the recommendation is to launch with the satellite

---

[1] unless otherwise noted, "day" refers to orbit day (95 minutes), not terrestrial day (24 hrs.)

programmed to desaturate every orbit day, then observe actual data on $\delta\Omega(t)$ to determine if the period between desaturation can be lengthened. This is sufficiently easy to do that further analysis on this particular issue is not worth the effort.

A brief review of the results of Section 4.1.4 follows. Given a desired desaturation torque, specified in $\mathcal{F}_B$:

$$\underline{\tau}_{ds} \equiv \begin{bmatrix} 0 & \tau_{ds} & 0 \end{bmatrix}^T \tag{5.66}$$

and the ambient field vector defined again in $\mathcal{F}_B$:

$$\underline{\beta} = \begin{bmatrix} \beta_1 & \beta_2 & \beta_3 \end{bmatrix}^T \tag{5.67}$$

equation (4.15) states that the spin plane coils should be actuated according to:

$$\begin{bmatrix} m_1 \\ m_3 \end{bmatrix} = \left(\frac{\tau_{ds}}{\beta_1^2 + \beta_3^2} \right) \begin{bmatrix} -\beta_3 \\ \beta_1 \end{bmatrix} \tag{4.15 repeat}$$

so as to produce the desired desaturation torque and a spin-plane disturbance torque (define as $\underline{\tau}_d$):

$$\underline{\tau}_{ds} + \underline{\tau}_d = \begin{bmatrix} 0 \\ \tau_{ds} \\ 0 \end{bmatrix} + \begin{bmatrix} -\rho\beta_2\beta_1 \\ 0 \\ -\rho\beta_2\beta_3 \end{bmatrix} \qquad \rho \equiv \frac{\tau_{ds}}{\beta_1^2 + \beta_3^2} \tag{5.68}$$

and the coil commands of (4.15) are optimal in that $||\underline{\tau}_d||$ is the minimum achievable.

The "desaturation loop" is simply the choice of $\underline{\tau}_{ds}$ based on feedback of $\delta\Omega$ and is clearly coupled to the spin-axis control loop- it can be viewed as a source of disturbance torque which, when corrected for by the spin-axis loop, results in the stored angular momentum to approach nominal. The desaturation loop is also dynamically coupled to the spin-plane control loop in the sense that $\underline{\tau}_d$ induces nutation which the spin-plane loop corrects for, but it is "decoupled" in the sense that the spin-plane loop, employing only m_2, cannot generate an opposite torque to correct for $\underline{\tau}_d$ directly. Therefore, the desaturation loop can be viewed as a low-bandwidth outer loop to the spin-axis control loop, and not in the spin-plane loop at all. This justifies the claim made earlier that it is "easy" to switch the

desaturation loop on and off- doing so requires no change in either of the orbit day attitude control loops.

The desaturation loop is chosen to be as simple as possible. The relationship between τ_{ds} and $\delta\Omega$ is $1/I_w s$, which pleasantly cannot be made closed loop unstable for any loop gain of the correct sign (simple root locus argument). Therefore, no other dynamic elements are inserted in the loop. The control law is:

- $\tau_{ds} = -K(\delta\Omega)$, $K \geq 0$. K is chosen in advance to for some default closed loop time constant. The time constant is I_w/K, and is set at 1000 sec. in simulations.

- Limit K on line so as to not violate a preset maximum for $||\tau_d||$, which is related to the computed τ_{ds} according to equation (4.20):

$$\left\|\tau_d\right\| = \frac{\left\|\tau_{ds}\right\|}{\tan(\Psi)} \qquad \text{(4.20 repeat)}$$

In simulation, $\max(||\tau_d||)$ is set to 40 μN•m.

- K = 0 when θ is bad and the spin plane control loop is shut off.

We are finally in a position to summarize the day controller in a block diagram, abstracted to the level of actuation signals being torques. Figure 5.33 shows this. Note that the only coupling shown between the spin-axis loop and the spin-plane loop is the desaturation disturbance torque; in reality, coupling due to $\mathcal{F}_B \neq \mathcal{F}_P$ also exists. Note also from the block diagram that the desaturation loop gain K establishes a $\delta\Omega$ "band" which depends on the magnitude of the spin-axis compensator command output τ_c, which in turn depends on the environmental disturbance torque magnitude. Thus the desaturation loop is seen as a compromise on the spin-axis loop disturbance rejection.
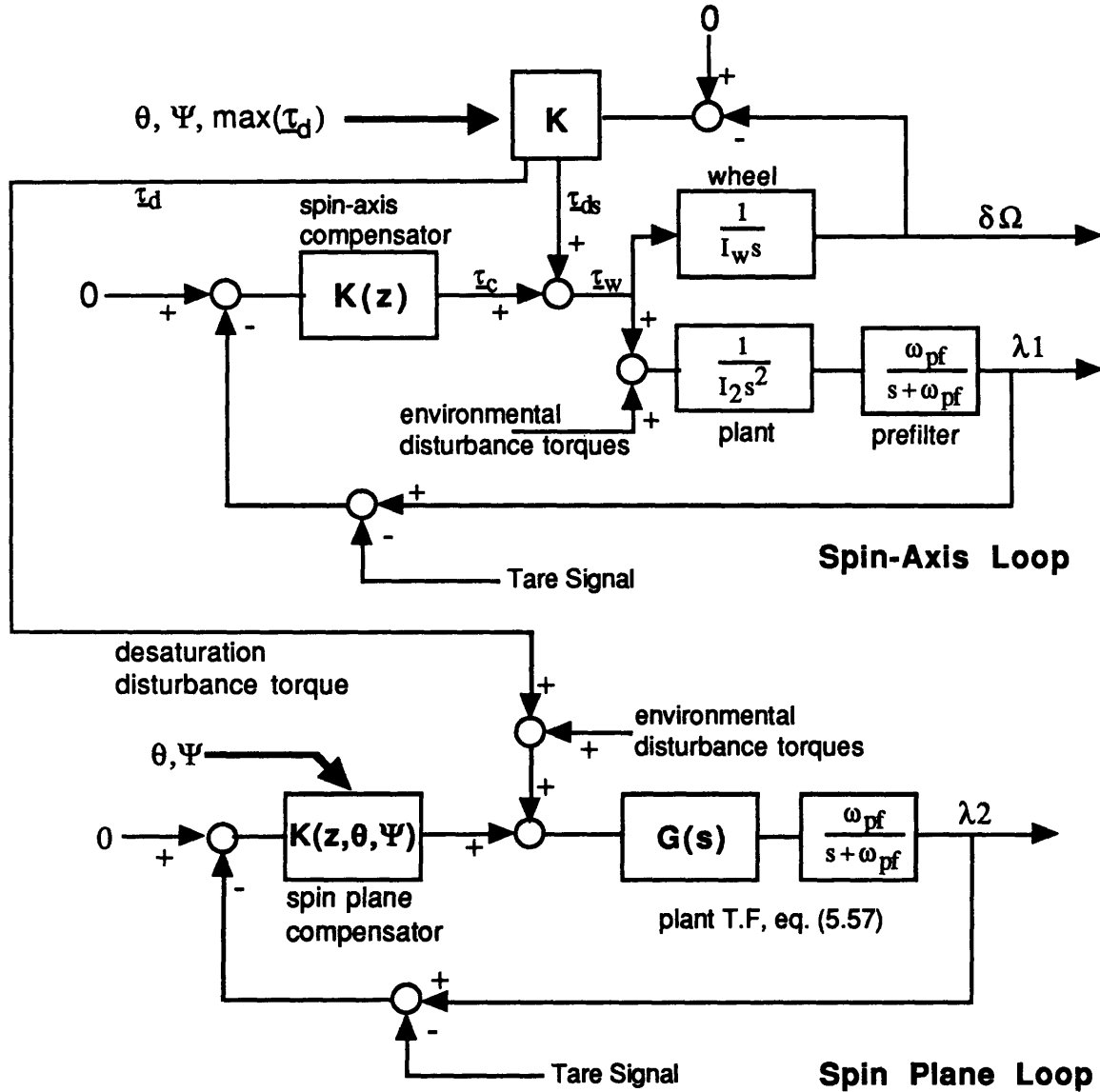
Figure 5.33: orbit day controller block diagram, mixed CT/DT representation, abstracted control signals, plant/actuator/sensor imperfections ignored. The decoupling strategy seen between the desaturation loop and the spin-plane loop is justified by the fact that the desaturation disturbance torque is orthogonal in the spin-plane to the control torque generated by the spin-axis magnet m_2. Therefore, a controllability argument would say that the spin-plane loop cannot benefit from an awareness of the desaturation loop's existence.

5.4 Slew Maneuver Control

The arrangement of the wheel axis orthogonal to the sun vector allows the possibility to add another degree of pointing freedom. By slewing about the spin axis with torque from the momentum wheel, the sun vector can be moved to an azimuth a_0 other than 0. This affords observation outside the nominal field of view. However, charging of the batteries with the solar panel during orbit day requires that the spacecraft be pointed with zero sun azimuth for the majority of the orbit day, performing the slew maneuver towards the end. The maneuver can be divided into two major tasks:

- Setting up the roll angle of the spin axis with respect to the sun vector. This determines the final direction (relative to an appropriate celestially fixed frame) of the science instrument direction $-B_3$ after a slew. The implications of this maneuver on sensor requirements and modifications to the orbit day attitude controllers are discussed in Chapter 6, but time did not permit detailed design of control for this thesis.

- Performing the slew itself and, more importantly, damping any induced nutation to permit attitude capture by the CCD camera when transitioning to orbit night. The time required for the rate damping determines how close to the end of orbit day the slew can be performed and consequently the amount of time spent at maximum solar panel efficiency. Control for the slew maneuver is demonstrated to be a small modification of the orbit day controller. Simulations are performed taking coil saturation into account to estimate realistic rate damping times, which are shown to be comfortably within 5 minutes.

Attitude sensing from sun-sensors only is assumed. Rate damping requires sun sensors of sufficient angular resolution, which implies limited field of view. The resolution of $\pm 5°$ into 10 bits, used as the design value in the previous section, is found to be sufficient. This implies an array of 18 such sun-sensors to get full 180° coverage, which is not necessary because the science instruments themselves are wide (2.5 to 2π steradians, [RICK87]) field of view. Instead, a discrete subset of desired sun vector azimuth "notches" along the $+B_3$ half of the spin plane can be defined, and high resolution sun sensors be placed with their boresights aligned with these azimuth notches. A wide field of view, low resolution sun sensor which provides coverage approaching 2π sr. is also assumed to exist, it being necessary for initial attitude acquisition and recovery from tumble in the event of an unforeseen long duration ACS shutdown. The proposed configuration is reviewed in Figure 5.34.
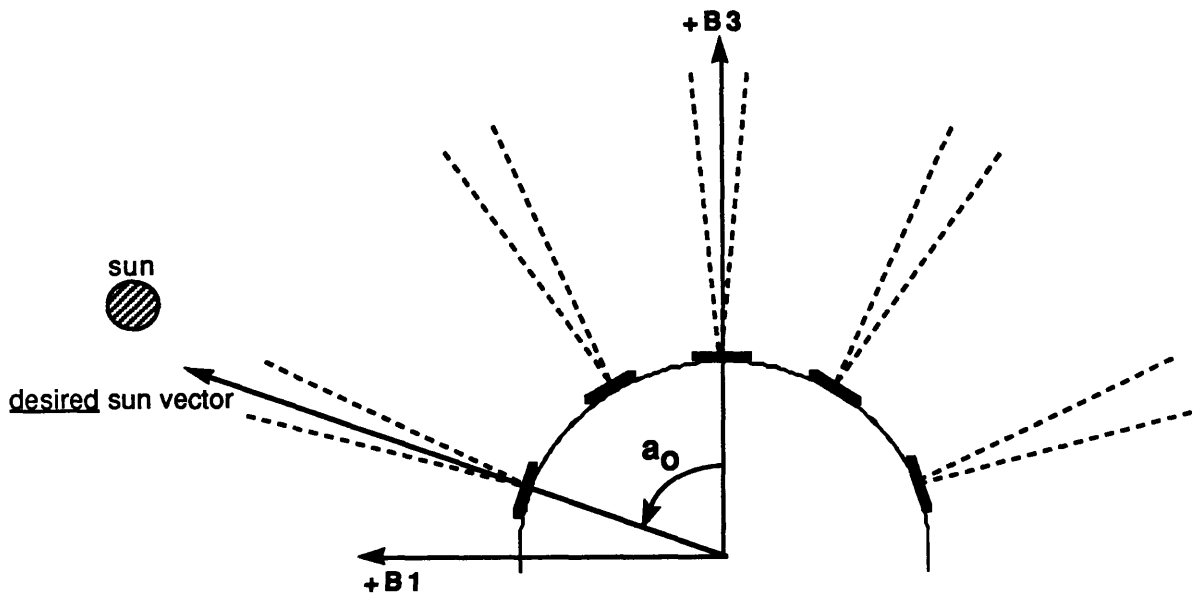
Figure 5.34: proposed sun-sensor array configuration in schematic form. The boresight azimuth notches shown are 0, ± 30°, ± 70°, but these are arbitrary and chosen merely as design examples. The actual number and azimuths of the sun sensors have not been determined. Note the emphasis on the definition of a_0 as the desired sun azimuth- an approximately -3° azimuth error λ_1 of the sun from the +70° sun sensor is shown. Also, note that by always choosing a_0 to lie along a boresight, the $a_0 = a_q + \delta a_q$ condition of Section 4.2.2 is satisfied, enabling simplifications in the kinematic formulae (4.29) and (4.30).

5.4.1 Strategy

We propose to divide the slew maneuver itself into two phases: a large angle open loop rotation about the spin axis, and a terminal azimuth nulling / nutation damping closed loop phase. The choice of performing the large angle rotation open loop is based largely on power consumption considerations. While it would be trivial to close a feedback loop around the azimuth error signal from the low-resolution sun-sensor, such an arrangement inevitably results in overshoots and undershoots in issued wheel torque commands if a trajectory of azimuth angles is commanded to be followed, This is unnecessary power consumption. It would be possible to improve on this situation with feedforward of azimuth rate trajectory commands into the loop, but such an implementation then exceeds the complexity of an open loop arrangement without any real benefits. It is simply unnecessary to track an azimuth trajectory precisely, which is what a loop closed around the sun sensor buys us. A more sensible and power efficient approach is

a bang-bang maneuver with a continuously estimated coast time between the bangs to bring the spacecraft almost to rest and with the azimuth error in the field of view of the terminal fine-resolution sun sensor. A closed loop regulator can then complete the maneuver gently while damping induced nutation. The propose scheme is summarized in figure 5.35.
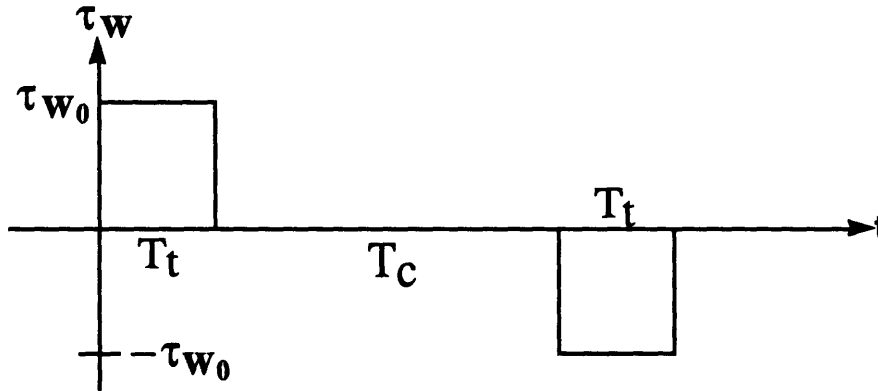
Figure 5.35: wheel torque command sequence for large angle rotation. The torque magnitude τ_{wo} and torque duration T_t are pre-set based on the desired power consumption and coasting speed. The necessary coast duration T_c is known to within the knowledge of the relevant inertias I_2 and I_w, but its estimate may also be continuously improved based on measurements from the low-resolution sun sensor driving an estimator. Note the open-loop guidance scheme ensures that no inefficient undershoots or saturating overshoots occur in the the torque history.

The terminal closed loop regulator, then, has two tasks:

- null out remaining azimuth error and azimuth rate (ω_2)
- null out induced spin plane (nutation) rates

It is decided not to null out the post slew elevation error with the terminal regulator. Such a correction involves perturbing the spin plane. If a substantial error exists, then correcting for it requires a long time to prevent continuous coil saturations and the subsequent inability to damp rates, as discussed earlier in Section 5.3.5. However, since the slew maneuver is to be performed as near to the end of the orbit day as possible, the required time doesn't exist. On the other hand, if no substantial elevation error is present, correcting for the small error is unnecessary and hinders nutation damping. It should be noted that this argument applies to the slew maneuver away from the nominal zero azimuth sun-vector at the end of the day- the terminal regulator corresponding to the slew back to the nominal

azimuth at the beginning of the orbit day is simply the day controller and does attempt to correct for elevation errors.

How much nutation is induced by the large angle rotation? This depends on T_t, τ_{wo}, the actual offset between principal and body frames, and the initial conditions at the beginning of the torque command. A proof from the equations of motion has not been done, but it is observed in simulation that by choosing to torque duration T_t to be an integral multiple of the nutation period $sqrt(I_1 I_3)/h_w$, the final induced nutation is minimized. The linearized natural motion of the body, as discussed in Sections 4.3.2 and [Hugh86], is composed of the spin-axis tracing out an ellipse around the angular momentum vector in inertial space. Thus, in the absence of a large axial rate ω_2, the transverse rates ω_1 and ω_3 are also elliptically related- that is, they are sinusoids of different magnitudes but 90° phase shifted with respect to each other and un-biased about 0. The magnitude of the nutation, $||\underline{\omega}||$, thus remains constant at its initial value. When a torque constant in the principal frame is applied (specifically the torque due to $\mathcal{F}_B$-$\mathcal{F}_P$ offset), the analysis of Section 5.3.1 show that the natural evolution of $\omega_1(t)$ and $\omega_2(t)$ is modified by constant offset terms (equations (5.49)). The offset terms now cause $||\underline{\omega}||$ to vary with time, since $\omega_1(t)$ and $\omega_2(t)$ are no longer un-biased about 0. Apparently, $||\underline{\omega}||$ varies such that it is minimum every nutation period, so that cessation of torquing after an integral number of nutation periods minimizes the initial conditions for the natural motion which then takes over.

In any case, simulation results are examined with τ_{wo} = 5 mN•m (25% max capacity of Ithaco Scanwheel), T_t = $sqrt(I_1 I_3)/h_w$ ~ 14.5 sec, and a 10° separation between B_2 and P_2. The achieved ω_2 is about 0.8 °/s, which should be sufficient for any slew. Resultant transverse rate peaks are not observed past 50°/hr, so the following worst case initial conditions should be sufficiently conservative to test the terminal regulator designs in simulation:

$$|\omega_1| = |\omega_3| = 100 \text{ °/hr, signs arbitrary}$$
$$|\omega_2| = 120 \text{ °/hr   (leftover azimuth rate)}$$
$$|\lambda_1| = 5° \text{   (leftover azimuth error)}$$

5.4.2 Terminal Regulator Designs

The spin-axis loop for the day controller is found to work sufficiently well, the initial value taring technique being especially effective in reducing violent wheel torque demands and eliminating any overshoots in the azimuth response.

The spin plane loops used are small modifications of those used in both orbit day and night. An important point to note is that, unlike the nominal day controller, the terminal regulator cannot afford to turn off when the local field is bad. Nutation damping must take place immediately to prepare for transitioning to orbit night. On the other hand, it is seen below that, when the elevation error λ_2 is not controlled, the stability sensitivity to θ being close to a_0 disappears. In fact, the gain matrix is not a function of a_0! The strategy is therefore to:

- Before the slew maneuver, predict the angle Ψ in the post slew attitude with a transformation of the magnetometer readings.

- When the predicted Ψ will be near $0/180°$ (recall figure 4.3), use a variant of the full-authority actuator mode. Otherwise, use a day controller (partial authority mode).

- For intermediate values of Ψ, use the day controller preferentially. It is found to be as effective in nutation damping as the full-authority controller, and requires less wheel activity.

The design of the modified day-controller is essentially the same as in Section 5.3, with the following change:

- modify the LQR cost function in the gain matrix calculation to not weight λ_2:

$$J = \int_0^\infty \omega_1^2 + \omega_3^2 + \tau_{tn}^2 \, dt \tag{5.69}$$

The striking thing about this change is that it leads to gains which, while they are functions of θ, do not change as varying values of the desired azimuth a_0 are used in the design plant model to calculate the gains. This can be explained physically following the same reasoning as in Section 5.3. An important consequence of this is that stability robustness due to θ measurement offset is not especially compromised as θ approaches a_0.

The filter half of the LQG configuration is the same as for the orbit day controller. The LQG gains are non-zero for only the estimated rates from the filter. Figure 5.36a shows these gains as functions of θ for $a_0 = \pm 30°$, $\pm 50°$, and $\pm 70°$. Figure 5.36b shows the closed loop poles achieved over the span of θ for $a_0 = 70°$, mapped to the s-plane via $s = \text{Log}(z)/1$. There is an unshown pole/zero cancellation at $s = 0$, of course.

Figure 5.36a

theta (degrees)
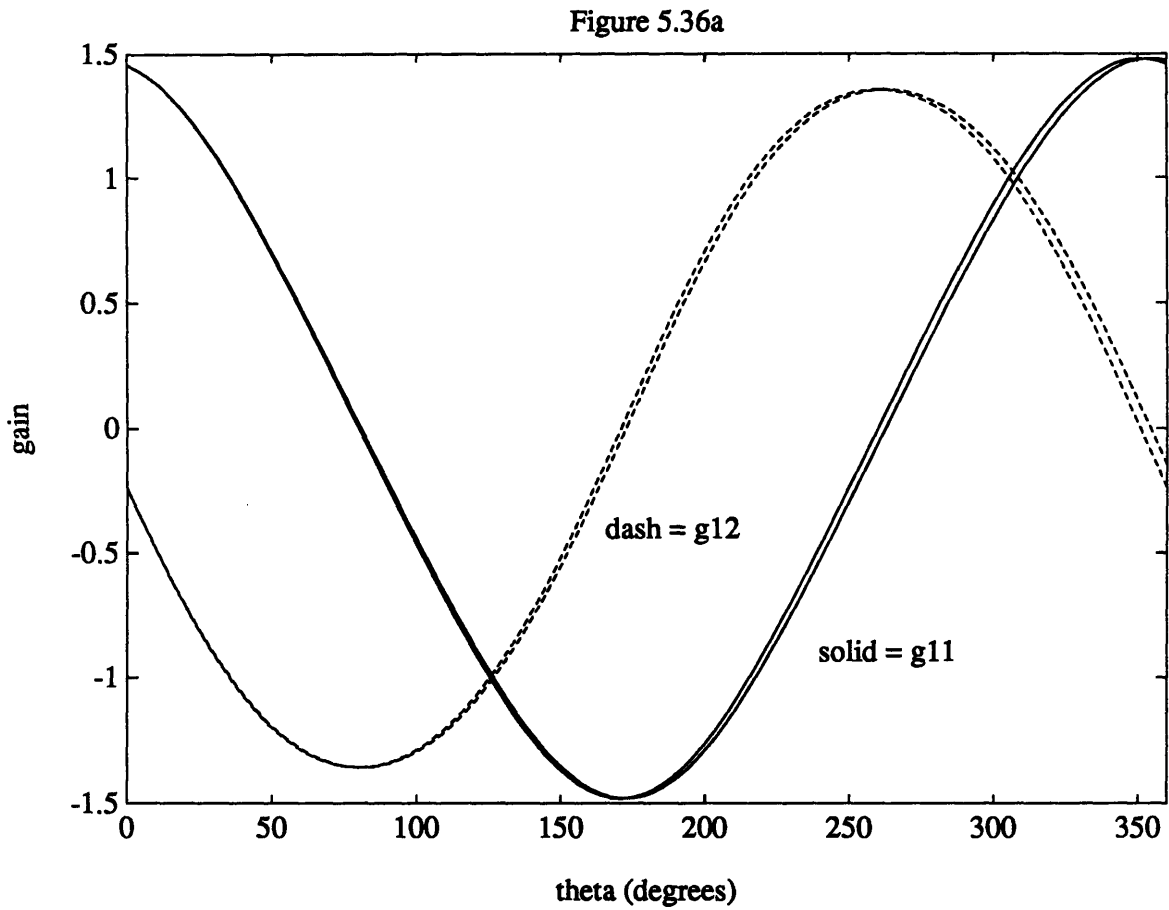
Figure 5.36a: LQG gains for $a_0 = \pm30°$, $\pm50°$, $\pm70°$; the control law is $\tau_{tn} = g11\omega_1 + g12\omega_3$, the ω's being outputs of the estimator half of the compensator. The gains are seen to be functions of θ and not functions of a_0. They are again easily fitted with either a sinusoidal fit or a polynomial fit for implementation.
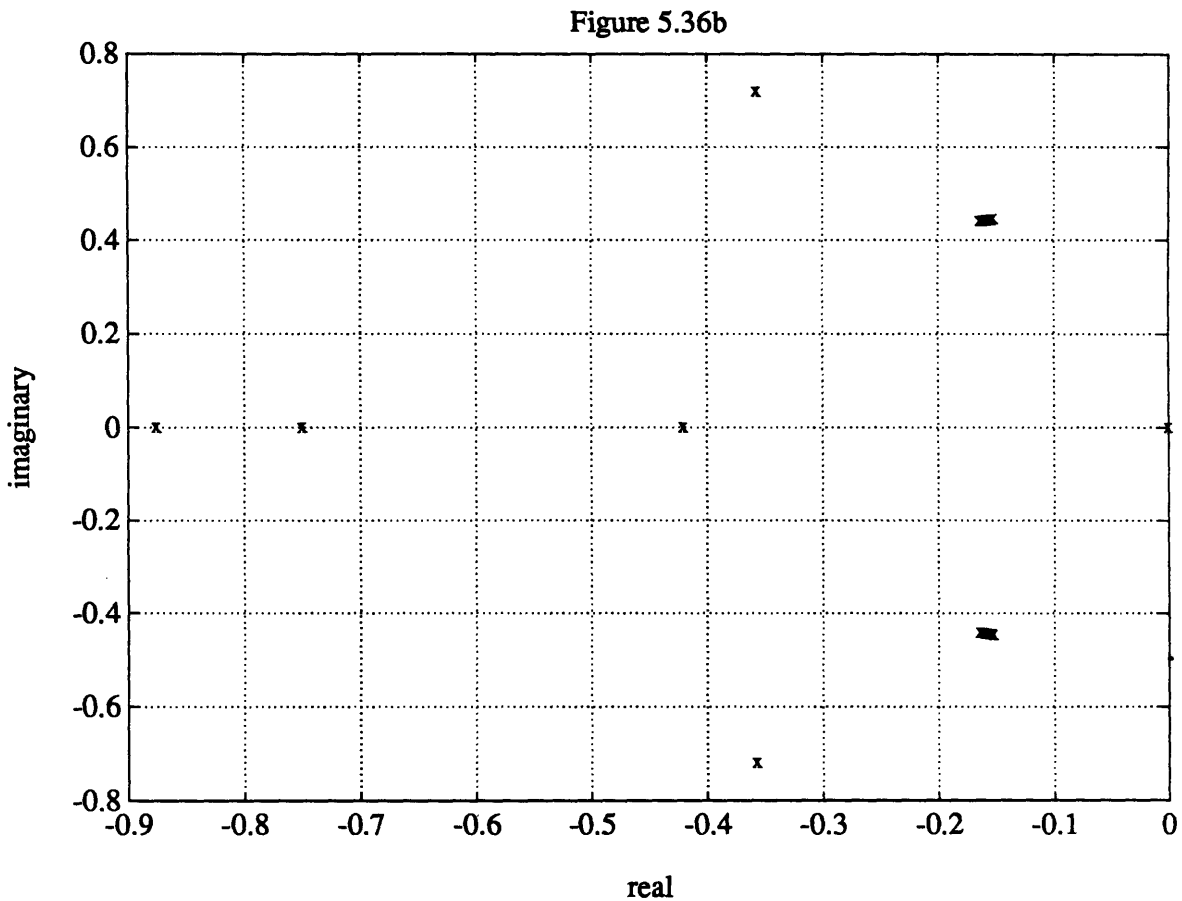
Figure 5.36b

Figure 5.36b: s-plane locus of closed loop poles as functions of θ, for the example case of a_0 = 70°. For other values of a_0, it is observed that the filter poles move slightly, but the nutation poles are in essentially the same place.

For the full-authority actuator mode, the only essential difference is the control distribution. Recall the decision to use the same spin-axis controller, which outputs a request for τ_2, the control torque about the B_2 axis. In the partial authority actuator mode, this is the same as τ_w, the wheel torque. In the full authority mode, a correction needs to be added for the spin-axis torque component which the spin-plane coils generate. A spin-plane controller is designed assuming that it can generate an arbitrary torque in the spin plane. Thus the same design plant model of (5.54) and (5.56) is used, but the state distribution matrix is simply B_{Ro}, and the control vector is $[\tau_1 \ \tau_3]^T$:

$$\begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_3 \\ \dot{\lambda}_2 \\ \dot{f} \end{bmatrix} = \begin{bmatrix} 0 & \dfrac{h_w}{I_1} & 0 & 0 \\ \dfrac{-h_w}{I_3} & 0 & 0 & 0 \\ \cos(a_o) & -\sin(a_o) & 0 & 0 \\ 0 & 0 & \omega_{pf} & -\omega_{pf} \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_3 \\ \lambda_2 \\ f \end{bmatrix} + \begin{bmatrix} \dfrac{1}{I_1} & 0 \\ 0 & \dfrac{1}{I_3} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \tau_1 \\ \tau_3 \end{bmatrix} \qquad f = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_3 \\ \lambda_2 \\ f \end{bmatrix} \qquad (5.70)$$

LQG design for this system is trivial. To minimize labor, again assume the filter portion of the LQG loop is assumed to be the same. The LQR portion of the loop is designed with the following cost function, again not weighting the elevation error λ_2:

$$J = \int_0^\infty \omega_1^2 + \omega_3^2 + \tau_1^2 + \tau_3^2 \, dt \qquad (5.71)$$

which yields the extremely simple control law, again independent of the value of a_0 in the design plant model:

$$\tau_1 = \omega_1 \qquad \tau_3 = \omega_3 \qquad (5.72)$$

where the ω's are the outputs of the filter, and a gain unit of (N/m)/(rad/s) is implicit. The requests from this loop and the spin-axis loop for control torques are assembled into the vector $[\tau_1 \ \tau_2 \ \tau_3]^T$ and sent to equation (4.4), which is inverted to calculate the necessary coil and wheel commands, just as for the orbit night actuator mode 1 controller:

$$\begin{bmatrix} 0 & \beta_3 & -\beta_2 & 0 \\ -\beta_3 & 0 & \beta_1 & 1 \\ \beta_2 & -\beta_1 & 0 & 0 \\ \beta_1 & \beta_2 & \beta_3 & 0 \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ \tau_w \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \\ 0 \end{bmatrix} \qquad (4.4 \text{ repeat})$$

Both controllers are found to perform quite satisfactorily in simulations, as will be shown in the next section.

It should be noted that throughout this chapter, little effort has been put into tweaking the compensator design gains/weights to "optimize" them. Since the mass properties are not finalized, proof of concept controllers which work are more important. Having

established that the control strategies and the given compensator structures can meet specifications with reasonably comfortable margins, the fine fiddling to tune for best performance is left until the detailed design phase and perhaps even until the satellite is in orbit.

5.5 Simulation Results

Simulation of controller performance is performed with a 4th order Runge/Kutta integrator using the full equations of motion (4.46). The angular velocity is integrated in the principal frame, but since it is a vector quantity, it can be transformed via C_{BP} easily. The attitude quaternions are integrated directly in the body frame using (4.23) and (4.24) as applied to the angular velocity in the body frame. The simulation codes are listed in Appendix A-5f. Plant imperfections modelled are summarized below:

Mass Properties:

- $\mathcal{F}_P$ is offset from $\mathcal{F}_B$ by a 10° rotation about an Euler axis in the spin plane. As discussed in section 4.3.1, a spin-plane Euler axis maximizes the coupling of h_w to the spin-plane, which is the dynamical worst case.

- The diagonal elements of the inertia matrix are all perturbed +15%, while the nominal wheel speed is up -5%. This gives a nutation frequency drop of 17% from the design plant model.

It may be questioned why the simulation condition of a nutation frequency drop is considered, when in fact the point has been made that the controllers are less tolerant to a nutation frequency rise, from stability robustness arguments. The reason is that simply stable systems don't necessarily perform well, and it will be shown below that, while the controller can remain stable with a tested case of +23% nutation frequency perturbation, the settling times and amount of sustained coil saturation are quite unacceptable. A more useful design study, therefore, is to observe the effect of a nutation frequency drop from the DPM, since it is always an option when faced with uncertainties in inertia measurements to use the lower bound for control design. It is shown in the simulations below that this is a good strategy, as settling performance is considerably better when using underestimated inertias (overestimated nutation frequency) than underestimated inertias in the DPM.

Sensors:

- ### Magnetometer (3 axis):

    Resolution: $120\mu T/1024$ in each axis

    Bias: $\pm 3\mu T$ each axis, sign random

    Misalignment with respect to $\mathcal{F}_B$: 5° about a random Euler axis

- ### CCD Camera: 1 second delay between measurement of attitude and issue of new actuator command. No other imperfections modelled.

- ### Sun sensor:

    Resolution in λ_1 and λ_2: 10°/1024.

    No noise modelled- assume no significant noise energy left after pre-filter at $\omega_{pf} = 0.75$ rad/s.

    No misalignment modelled.

Why no misalignment modelling? The kinematics analysis in Section 4.2.2 shows that inclusion of misalignment terms only contribute constants in the measurement expressions, without introducing any coupling between them if the desired sun elevation e_0 is 0 and the desired sun azimuth corresponds to the boresight. It is thus clear that tracking performance is simply to within the misalignment of the sensor, and that there is no point in modelling this explicitly in simulation.

Actuators:

- ### Magnetic Torquers:

    Resolution of moment command: $8/256$ A•m^2

    Offset (sign random) of each coil factor: 10%

    Hard saturation: 4 A•m^2

    Software saturation (in control law): 3.5 A•m^2

    Misalignment with respect to $\mathcal{F}_B$: 5° about a random Euler axis

• **Momentum Wheel**:

Resolution of acceleration command: 8×10^{-4} rad/s/s (corresponds to $5 \mu N \cdot m$ torque step for $I_w = 6.3 \times 10^{-3}$ kg$\cdot$m^2)

I_w offset (random sign): 5% (this is pretty conservative)

Hard saturation at 20 mN$\cdot$m torque (not really ever approached)

Misalignment with respect to $\mathcal{F}_B$: 5° about a random Euler axis

"Realistic" disturbance torques are calculated from the models in Chapter 3. The code is given in Appendix A-5f.

5.5.1 Orbit Night Controller, Actuator Mode 1 to 2

In actual implementation, the actuator mode chosen at the beginning of orbit night depends on the currently measured Ψ, with preference given to mode 1 (full authority) in neutral Ψ's. This example shows a transition from mode 1 to 2 during the course of the simulation. The following parameters are used:

• Simulation duration: 3000 sec.

• Integration time step: 1 sec.

• Compensator initial values: $\underline{0}$

• Plant initial values: $\underline{\omega} = [30\ 30\ -30]$ °/hr   principal rates

         $[\underline{\varepsilon}\ \eta] = [0\ 0\ 0\ 1]$ body attitude (quaternions)

         $\delta\Omega = 0$ wheel deviation from Ω_0

Note that the initial rates are higher than the specified maximum for orbit day and approaching the worst which the CCD camera is estimated to be able to achieve attitude lock. Note also the initial attitude parameters are zero, reflecting the strategy discussed earlier of minimizing transients by defining the current reference frame to be the null reference frame when coming out of a discontinuity. The compensator initial values are chosen to be 0 for simplicity.

x10⁻⁶   Figure 5.37a

Figure 5.37a: disturbance torques modeled, solid = body 1 component, dash = body 2 component, dotted = body 3 component, units N•m.
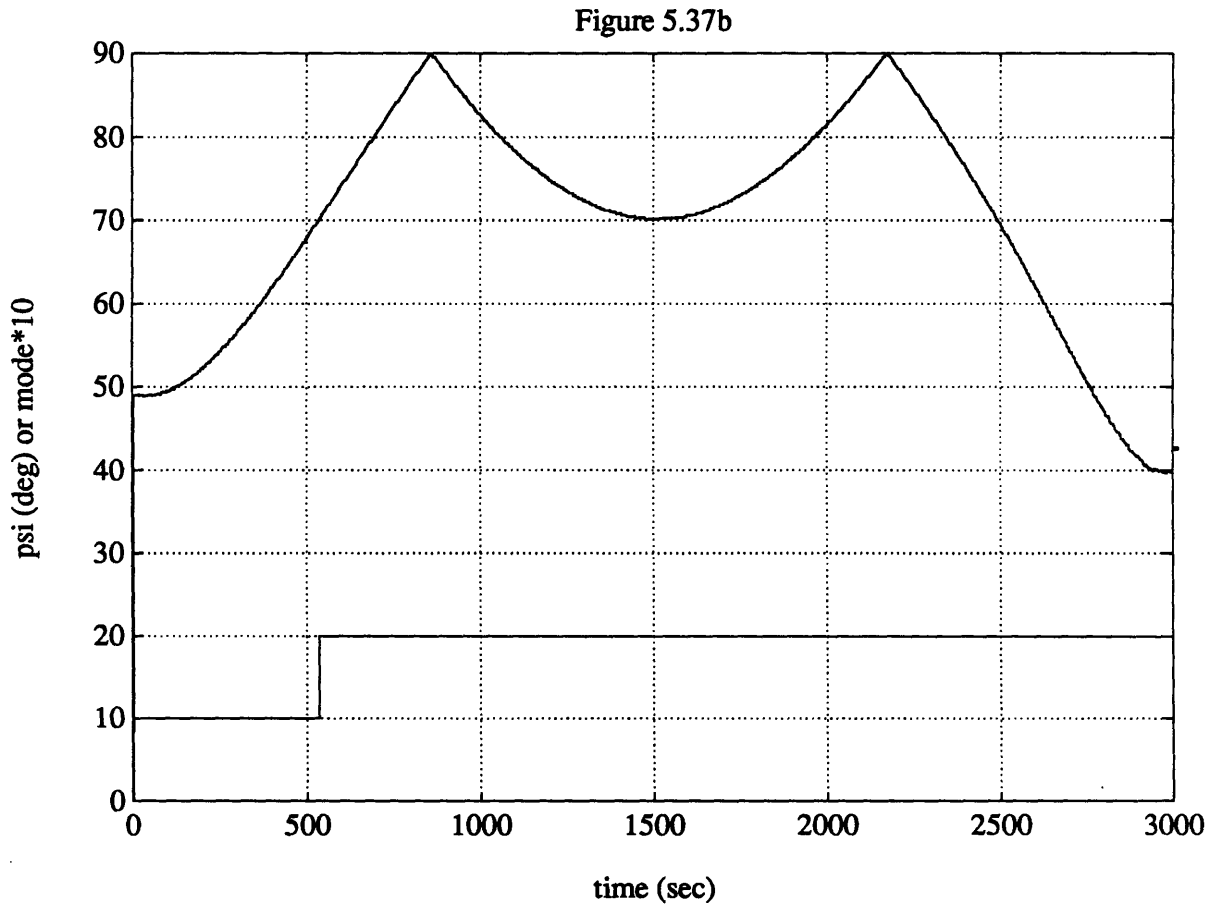
Figure 5.37b

Figure 5.37b: measured ψ history from corrupted magnetometer signal. Note the 90 ~ 180° half has been transformed to 0 ~ 90° to make actuator mode switching easier. The actuator mode history is the bottom plot, 10 = actuator mode 1, 20 = actuator mode 2.
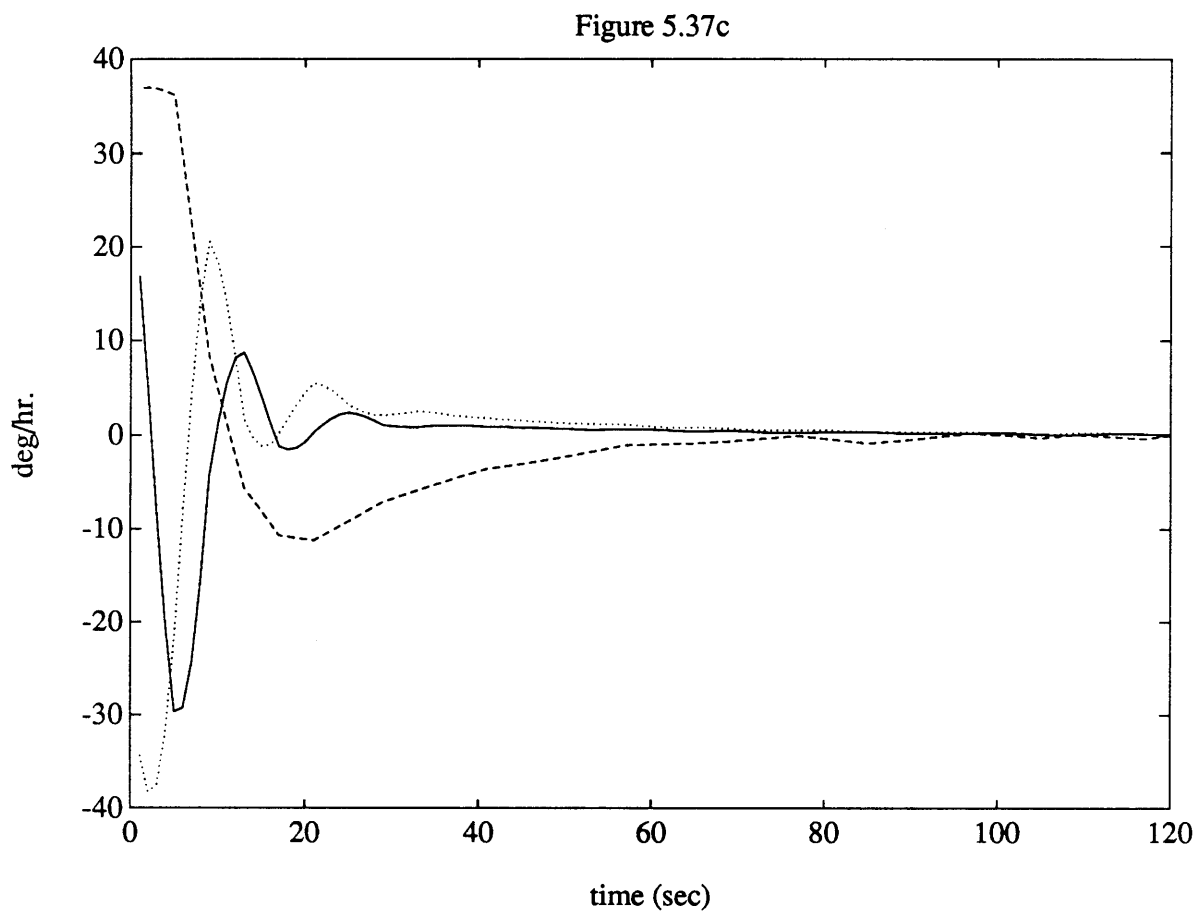
Figure 5.37c

Figure 5.37c: Rates in body frame, 0 ~ 120 sec., showing recovery from initial condition rates. Solid = body 1 component, dash = body 2 component, dotted = body 3 component, units are °/hr. The response is quite nice.
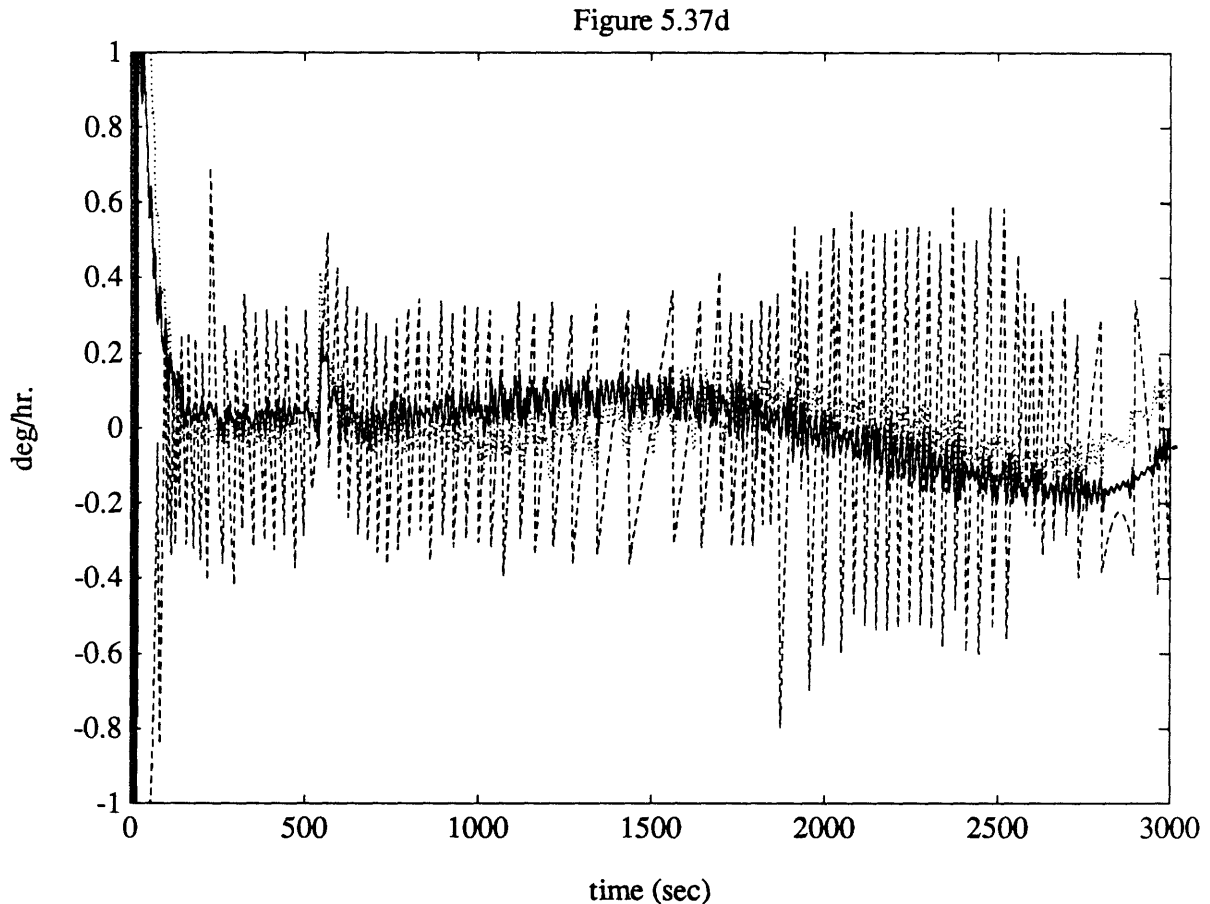
Figure 5.37d

Figure 5.37d: Rates in body frame for full 3000 sec., solid = body 1 component, dash = body 2 component, dotted = body 3 component, units are °/hr. The response is seen to meet specifications well. The limit cycles due to actuator quantization are clearly seen. RMS values for ω_2 = 0.2372 °/hr for t = 500 to 3000 sec. This is reasonably close to the value predicted analytically earlier. RMS(ω_1) for the same time period is 0.098 °/hr, while RMS(ω_3) = 0.075 °/hr. These are higher than the predicted values but are still an order of magnitude smaller than RMS(ω_2), as expected. It should be noted that the actuator mode transition at slightly past t = 500 sec. is accompanied by a visible but very small jump in rates. The attitude resetting technique works well to minimize switching transients.

Figure 5.37e
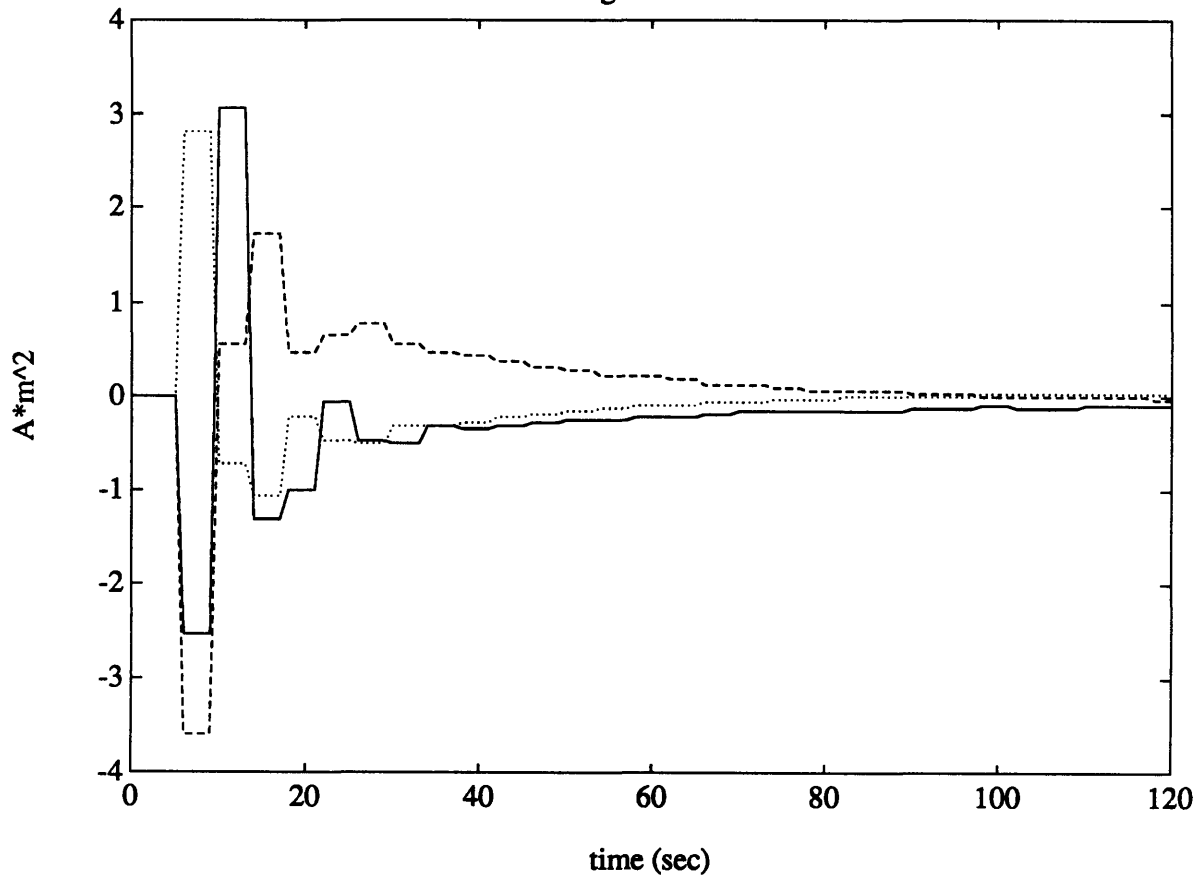
Figure 5.37e: torquer activity for t = 0 ~ 120 sec. showing recovery from initial rates. solid = m_1, dash = m_2, dotted = m_3, units = $A \cdot m^2$.
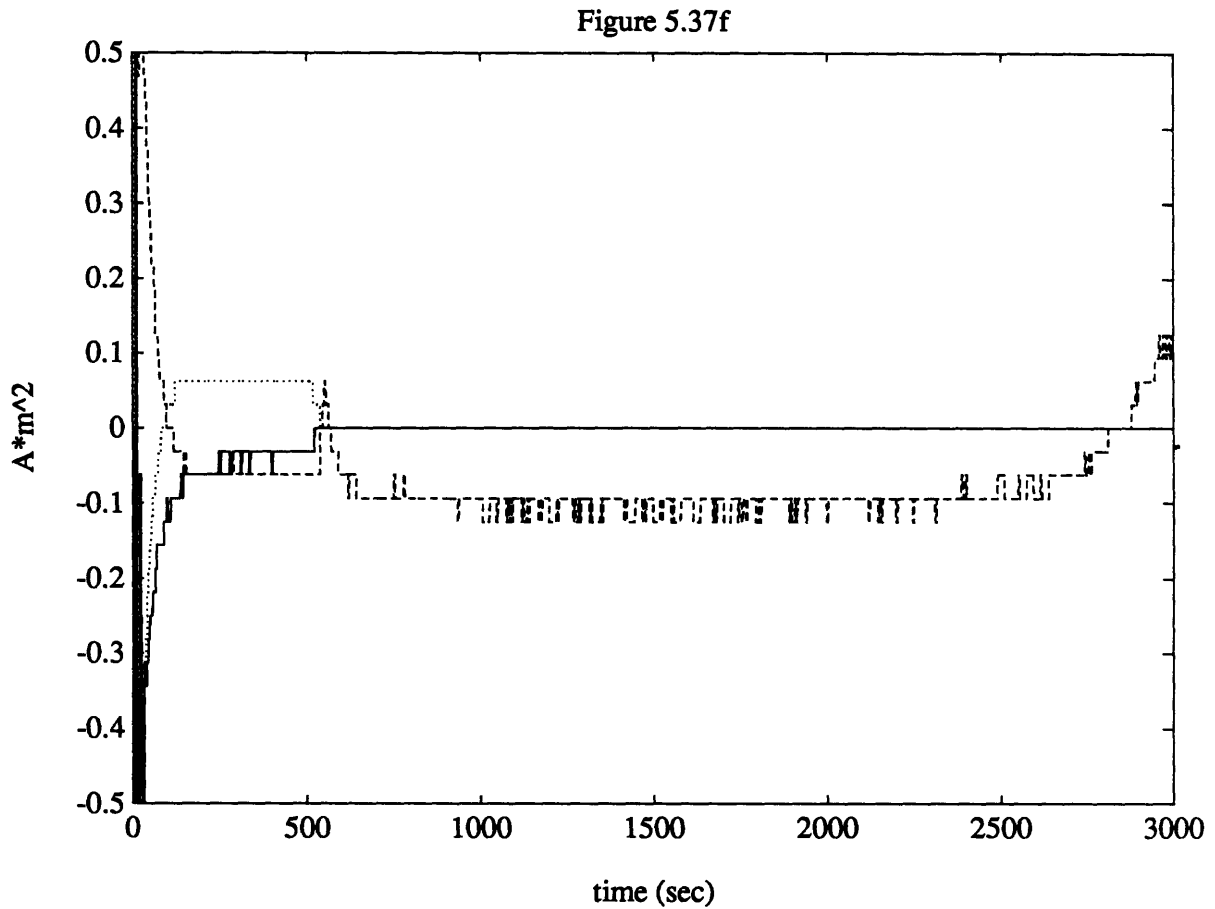
Figure 5.37f

Figure 5.37f: magnetic torquer activity for the full time scale. Solid = m_1, dash = m_2, dotted = m_3, units = A•m^2. Note that in actuator mode 2, $m_1 = m_3 = 0$. Note also the small limit cycles in m_2 caused by quantization.
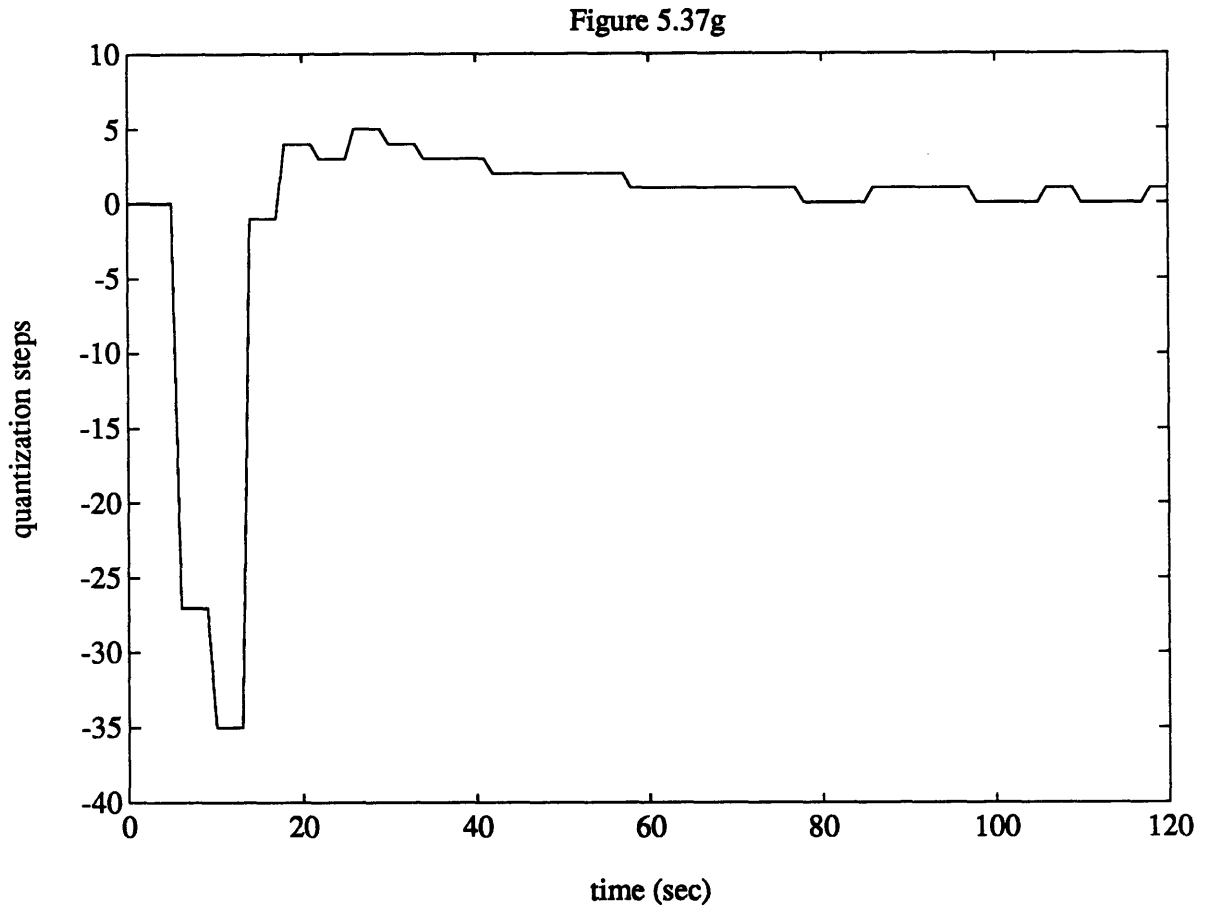
Figure 5.37g

Figure 5.37g: closeup of wheel acceleration activity for t = 0 ~ 120 sec., showing recovery from initial rates. Units are in quantization steps, each step being 8×10^{-4} rad/s/s.
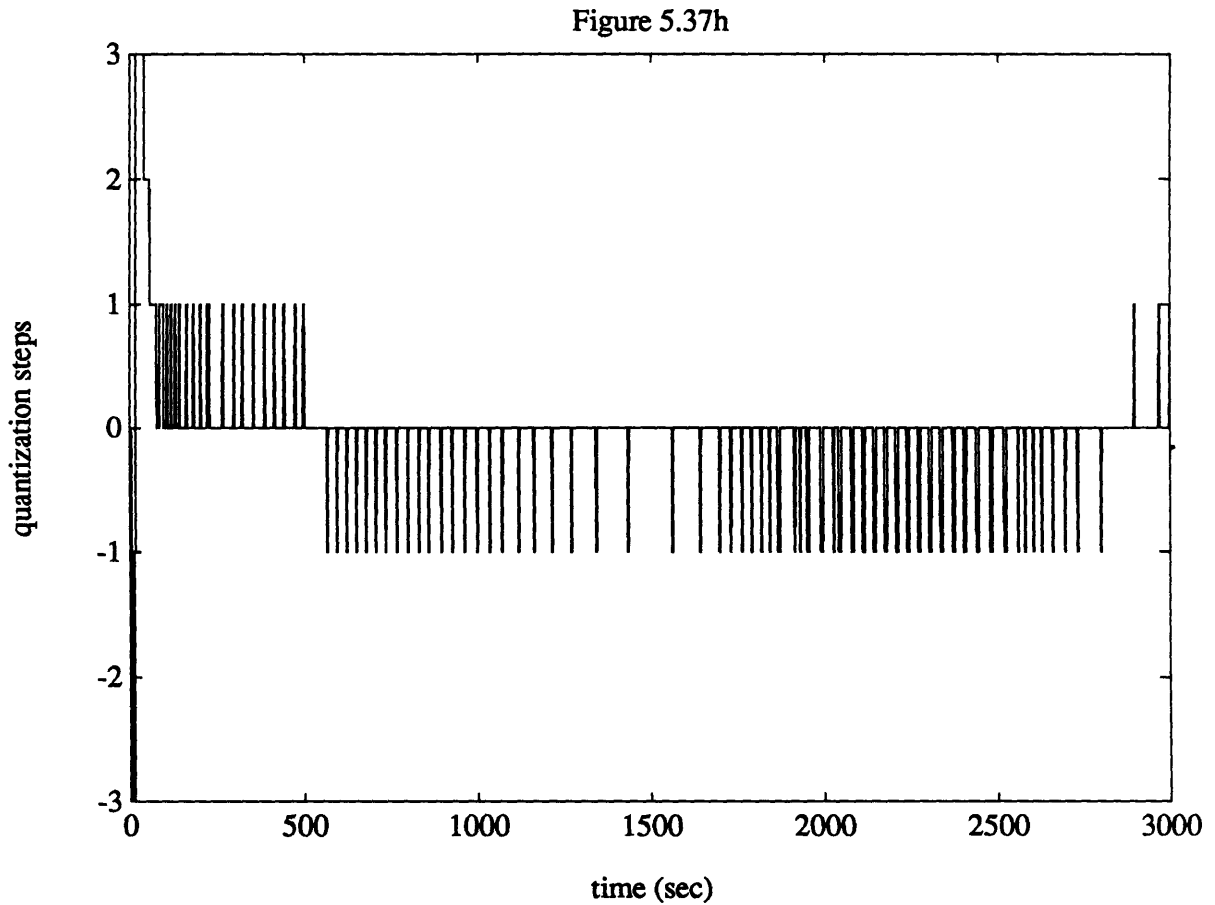
Figure 5.37h

Figure 5.37h: full time scale wheel acceleration demand, units in steps of 8×10^{-4} rad/s/s again. It is seen that, to correct for such small disturbance torques, wheel torque demand jumps between 0 and -1, causing the limit cycles seen in the rates.
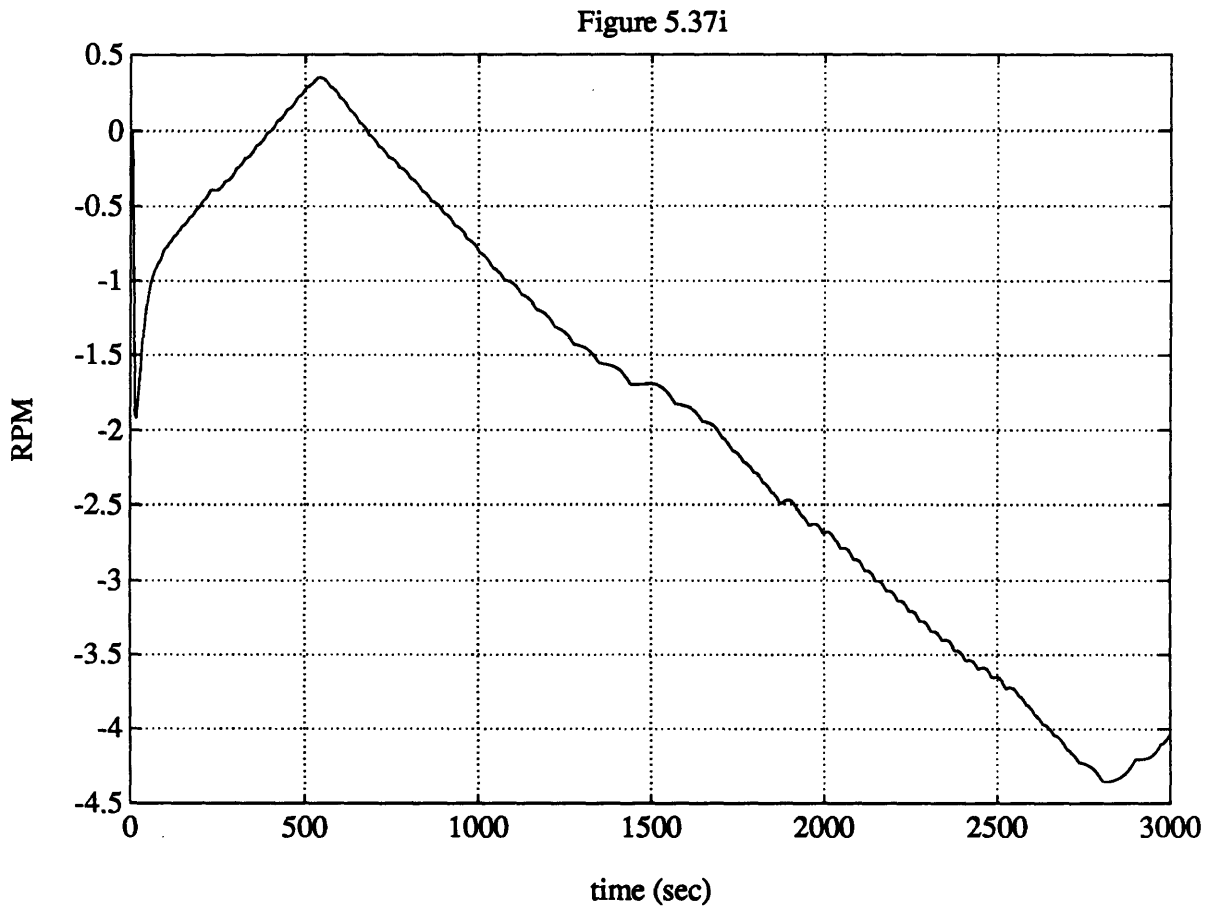
Figure 5.37i

Figure 5.37i: accumulated $\delta\Omega$ in RPM for simulation. Note the direction is as expected for counteracting the disturbance torque in Figure 5.37a. Note also the growing "wrong" direction excursion for time approaching the end of actuator mode 1 at about 500 sec., when the unfavorability of Ψ is causing more wheel demand to go towards correcting for the action of m_1 and m_3 in generating the desired control torque. Once actuator mode 2 is transitioned to, the wheel excursion is simply that necessary to counteract body disturbance component #2.

5.5.2 Orbit Night Controller, Actuator Mode 2 to 1

An example of a night simulation starting from actuator mode 2 and transitioning to mode 1 is given to demonstrate the ability of the partial authority actuator to recover from initial conditions and transient suppression during mode change. The same initial conditions are used as in the previous example:

Simulation duration: 3000 sec.
Integration time step: 1 sec.
Compensator initial values: $\underline{0}$
Plant initial values: $\underline{\omega} = [30\ 30\ -30]$ °/hr   principal rates
               $[\underline{\varepsilon}\ \eta] = [0\ 0\ 0\ 1]$ body attitude (quaternions)
               $\delta\Omega = 0$ wheel deviation from Ω_0

x10^{-6} Figure 5.38a

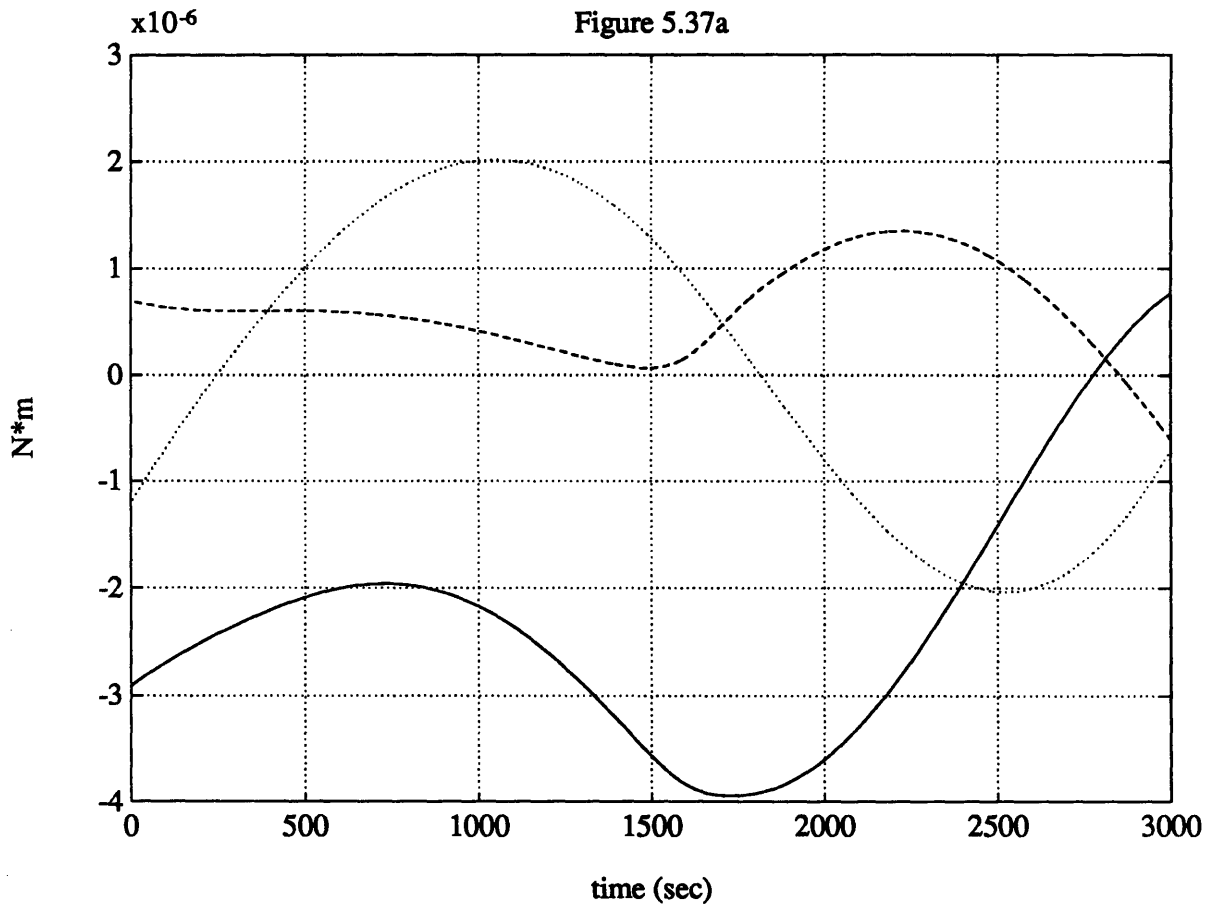Figure 5.38a: disturbance torques modeled, solid = body 1 component, dash = body 2 component, dotted = body 3 component, units N•m.
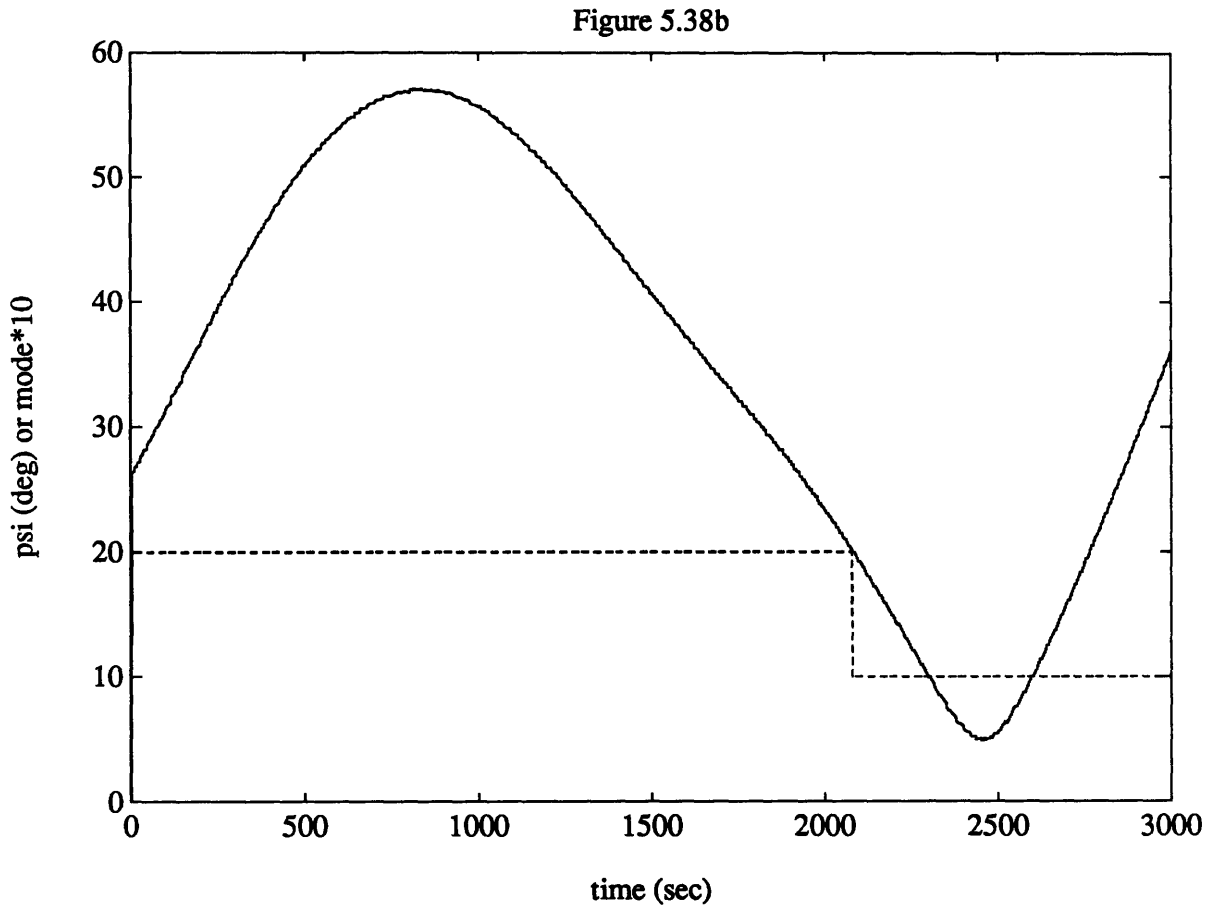
Figure 5.38b

Figure 5.38b: measured ψ history from corrupted magnetometer signal. The actuator mode history is the dashed plot, 10 = actuator mode 1, 20 = actuator mode 2.
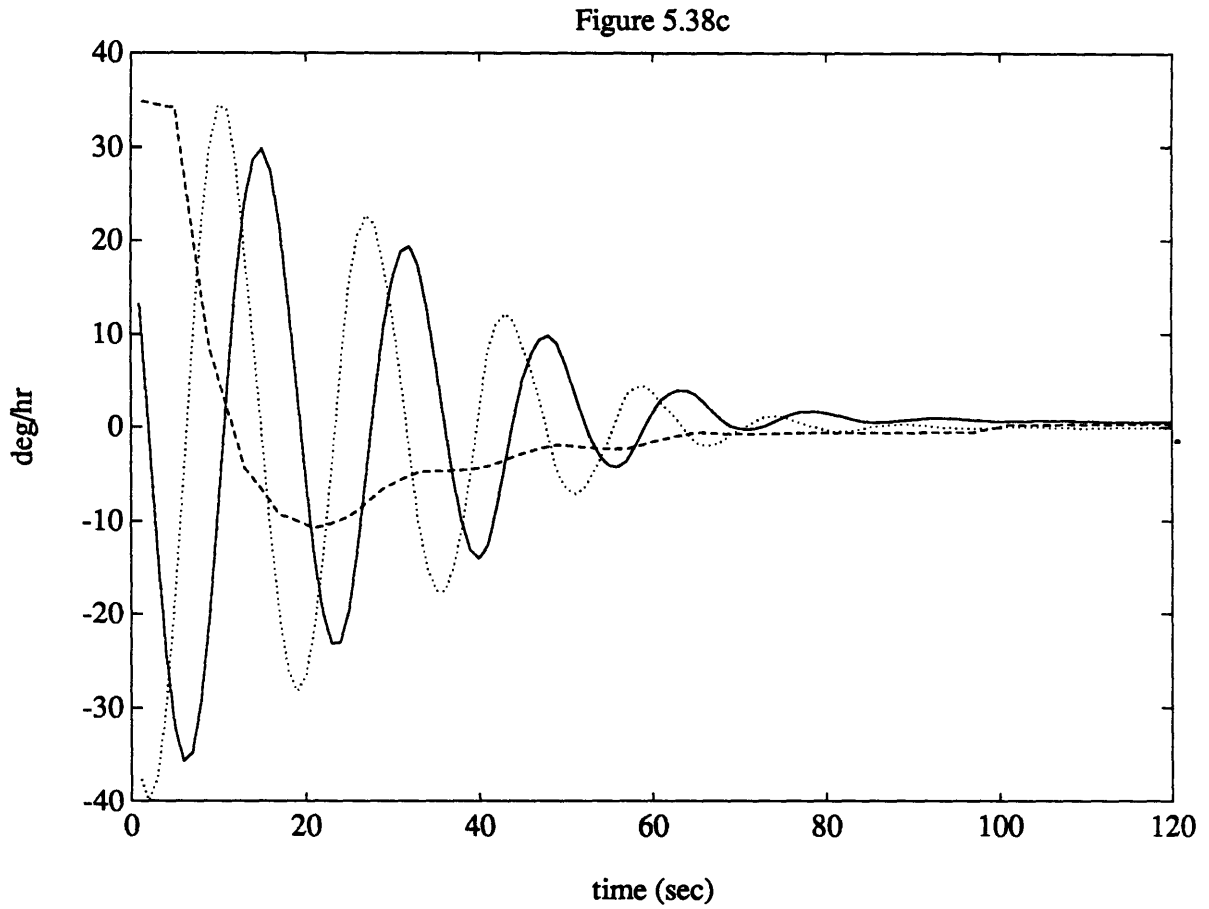
Figure 5.38c

Figure 5.38c: Rates in body frame, 0 ~ 120 sec., showing recovery from initial condition rates. Solid = body 1 component, dash = body 2 component, dotted = body 3 component, units are °/hr. Recall this is in partial actuator authority mode.

Figure 5.38d
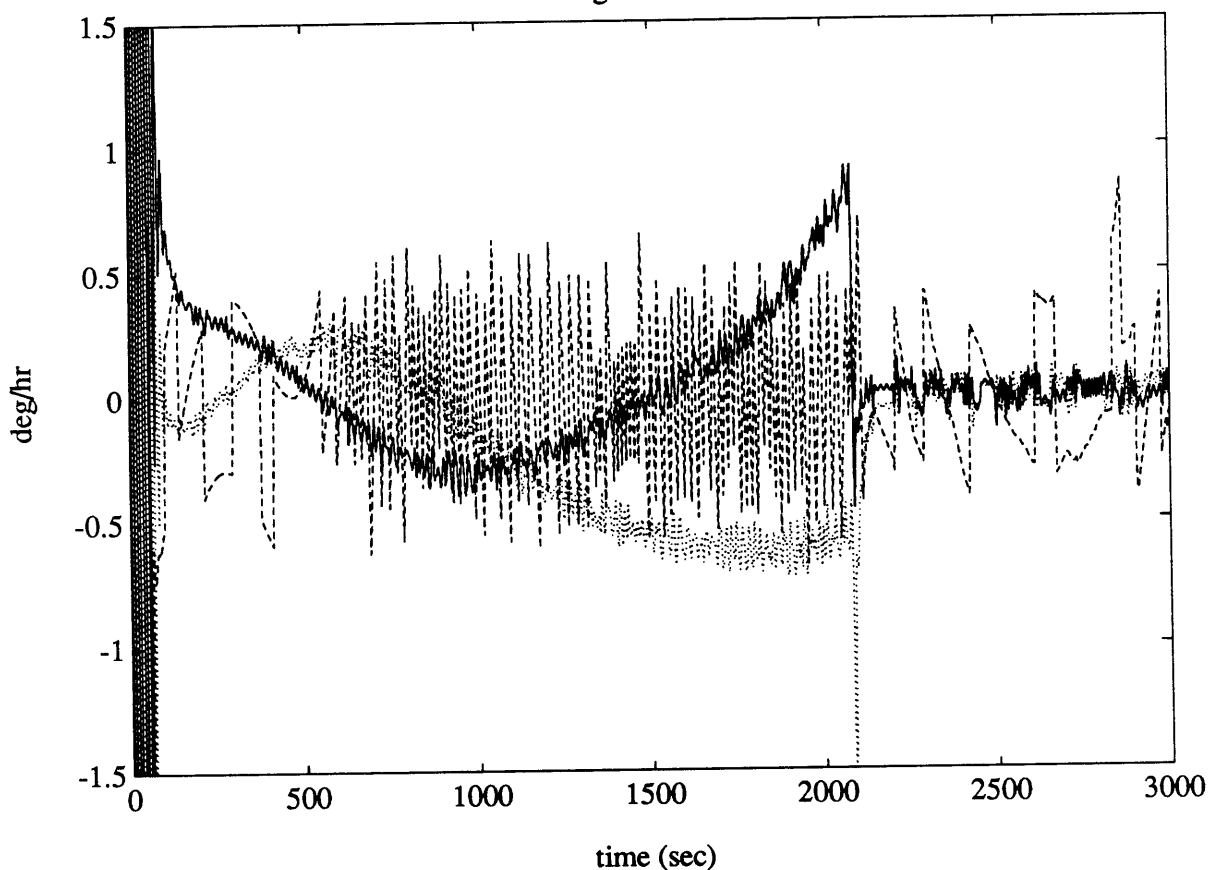
Figure 5.38d: body rates for full time scale, body 1 component, dash = body 2 component, dotted = body 3 component, units are °/hr. Note the graphic demonstration of the discussion in Section 5.2.4 about disturbance rejection. In actuator mode 2 (time to about 2100 sec.), it was argued that a component of the disturbance torque in the spin plane exists which is only suppressed in its effect on ω_1 and ω_3 by gyroscopic stiffness -6dB, while in actuator mode 1 the suppression is much greater. The high frequency fuzz is from actuator quantization, but the low-frequency swings for t<2100 sec, which disappears after the actuator mode switch, is precisely the mentioned effect. Note also the initial value intercept technique is quite effective in suppressing transients when switching actuator modes. The specification are met reasonably well.
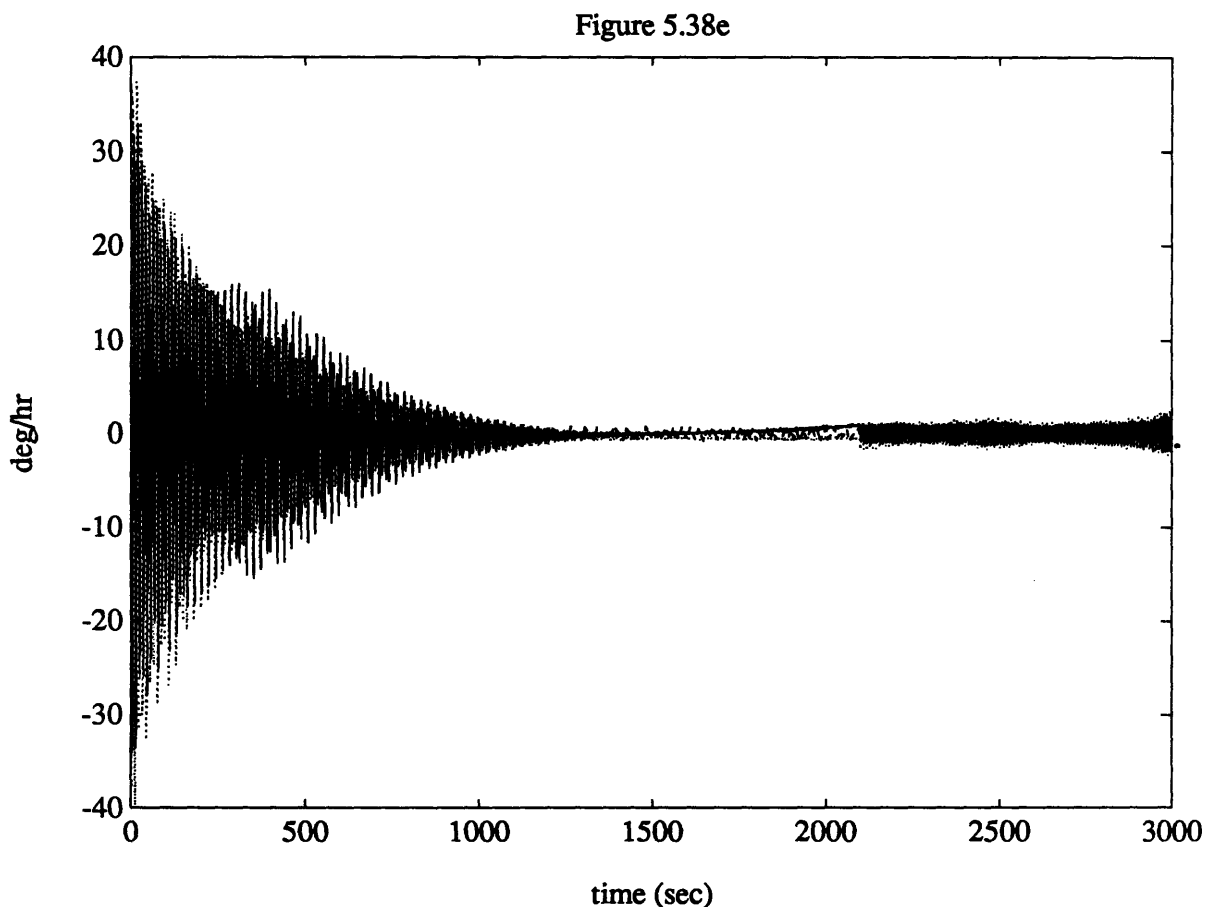
Figure 5.38e

Figure 5.38e: body frame rates for the same simulation, with all disturbances, Earth field, sensor and actuator imperfections equal, but with the diagonal elements of the plant inertial matrix dropped 15%, and h_w increased 5% (an actual plant nutation frequency rise of 23.5%). The same actuator mode switch is present at around 2100 sec. The settling time has increased to an unacceptable 1/3 of the orbit night as a result of the nutation frequency rise. Thus, while not disastrous in terms of stability, the performance is compromised when the inertias are overestimated by this amount. The increased settling times are due to both the decreased closed loop damping ratios and the increased amount of coil saturation.The lesson is to do control design with underestimated inertias.

Figure 5.38f
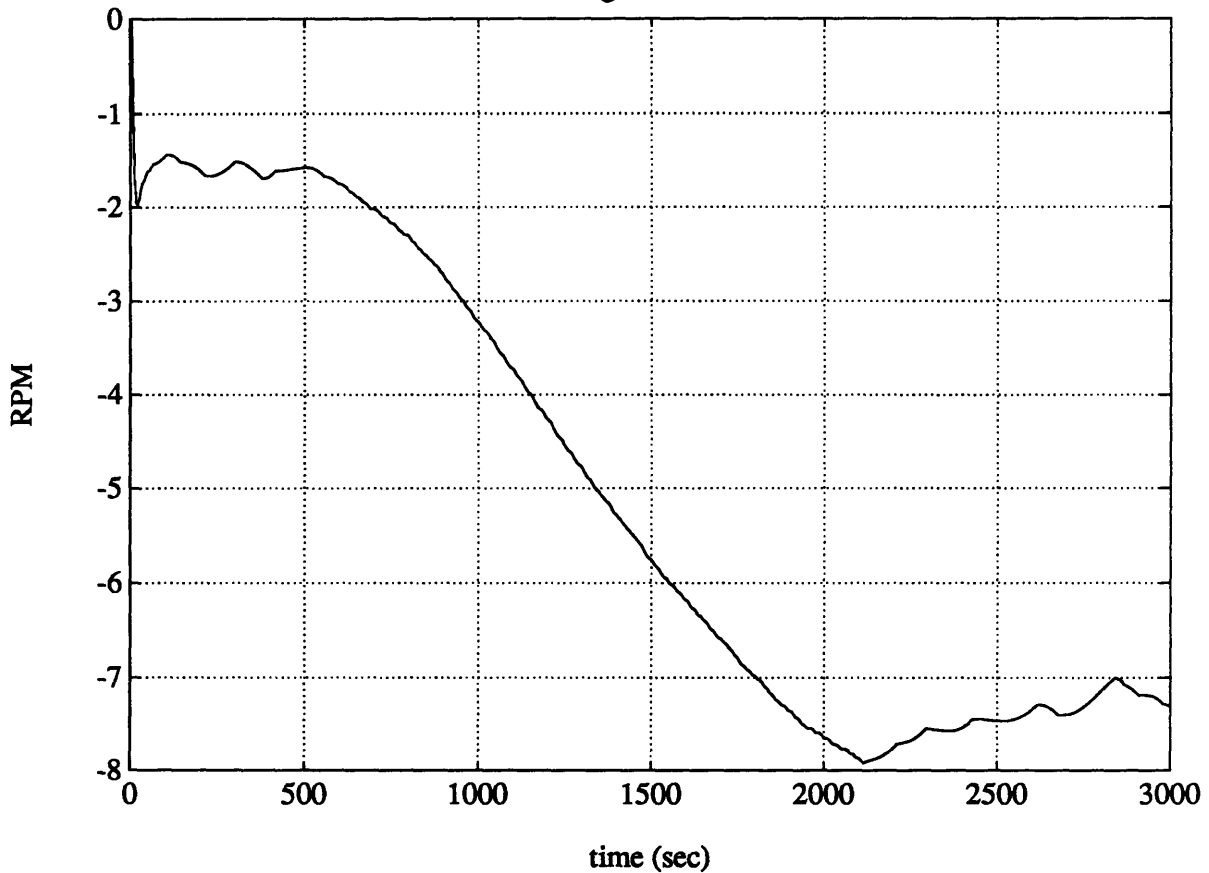
Figure 5.38f: the actuator activity (of the system shown in Figure 5.38d, not 5.38e) differs little qualitatively from the previous example (Section 5.5.1), so will not be included here expect for this plot of the accumulated wheel $\delta\Omega$ in RPM. Note that the accumulation per orbit night is not likely to exceed 1 to 2% of the nominal value. Small $\delta\Omega$ makes desaturation during the day easy ("easy" in the sense of requiring little power and disturbance generation) and improved controller performance.

5.5.3 Nominal Orbit Day Control Example

Several aspects of the nominal orbit day controller's operations are demonstrated in this example:

- Recovery from an assumed slew maneuver back to desired sun azimuth $a_0 = 0$ at the beginning of orbit day, which gives rise to these assumed worst case initial conditions:

  $\underline{\omega} = [-100\ 120\ 100]$ °/hr   principal frame rates

  $[\underline{\varepsilon}\ \eta] = [-1.5\ -2.5\ 0\ 1]$°   (infinitesimal) body quaternions

  It is legitimate to express the quaternions in degrees because of the $\Phi \sim \sin(\Phi)$ argument applied to the definition of quaternions in terms of Euler axis/angles (4.21). The given values correspond roughly to a 5° azimuth error and a 3°. elevation error from the sun sensor.

- Desaturation of the wheel, assuming a much worse than expected accumulation of:

  $\delta\Omega = 50$ RPM   wheel deviation from Ω_0

- Forced idle of the spin-plane control loop when θ gets within 20° of 0 ($a_0 = 0$).

Other values are the same as in previous examples, for the same reasons:

Simulation duration: 3000 sec.
Integration time step: 1 sec.
Compensator initial values: $\underline{0}$

x10⁻⁶
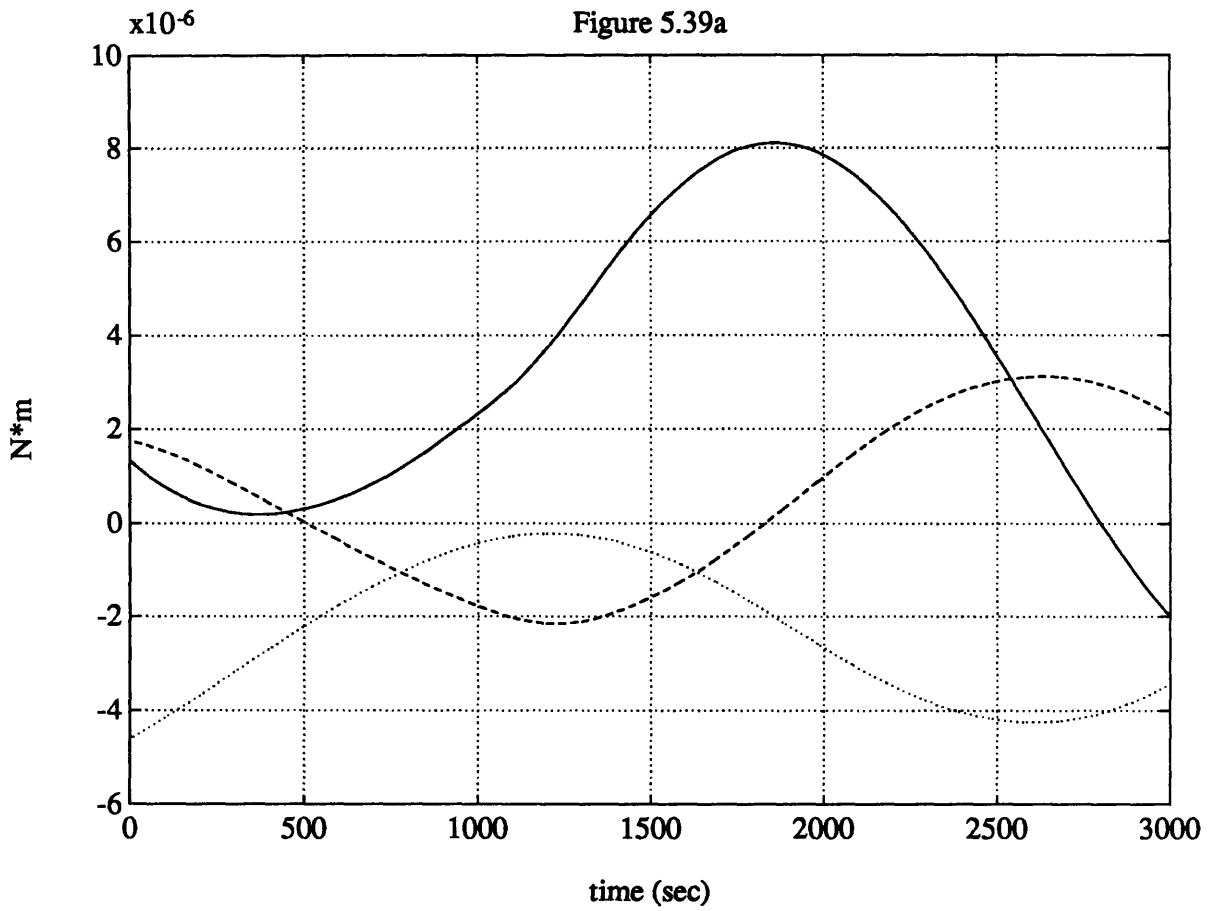
Figure 5.39a

Figure 5.39a: disturbance torques modeled, solid = body 1 component, dash = body 2 component, dotted = body 3 component, units N•m.
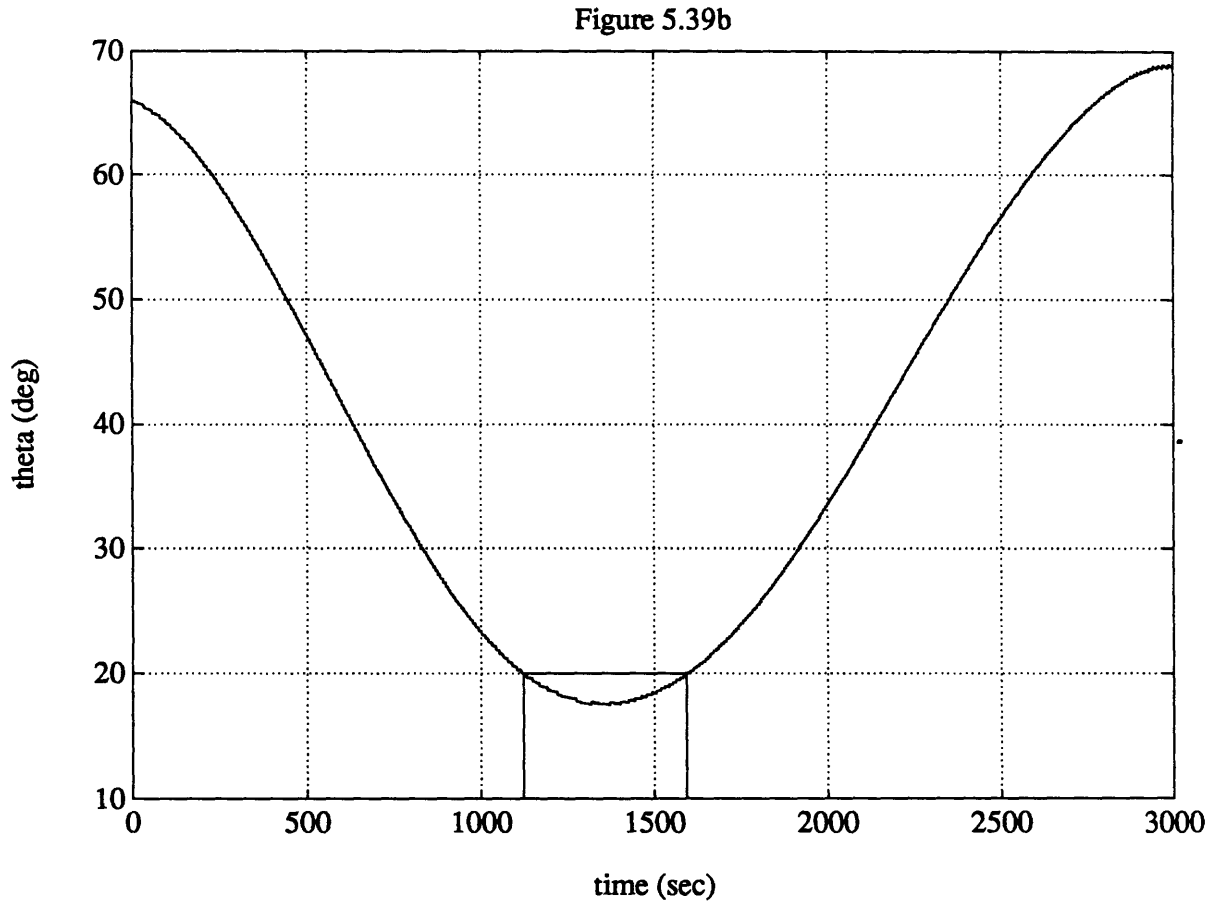
Figure 5.39b

Figure 5.39b: measured (from corrupted magnetometer model) θ history. The box represents when the spin-plane loop is idled, as discussed in the previous sections.
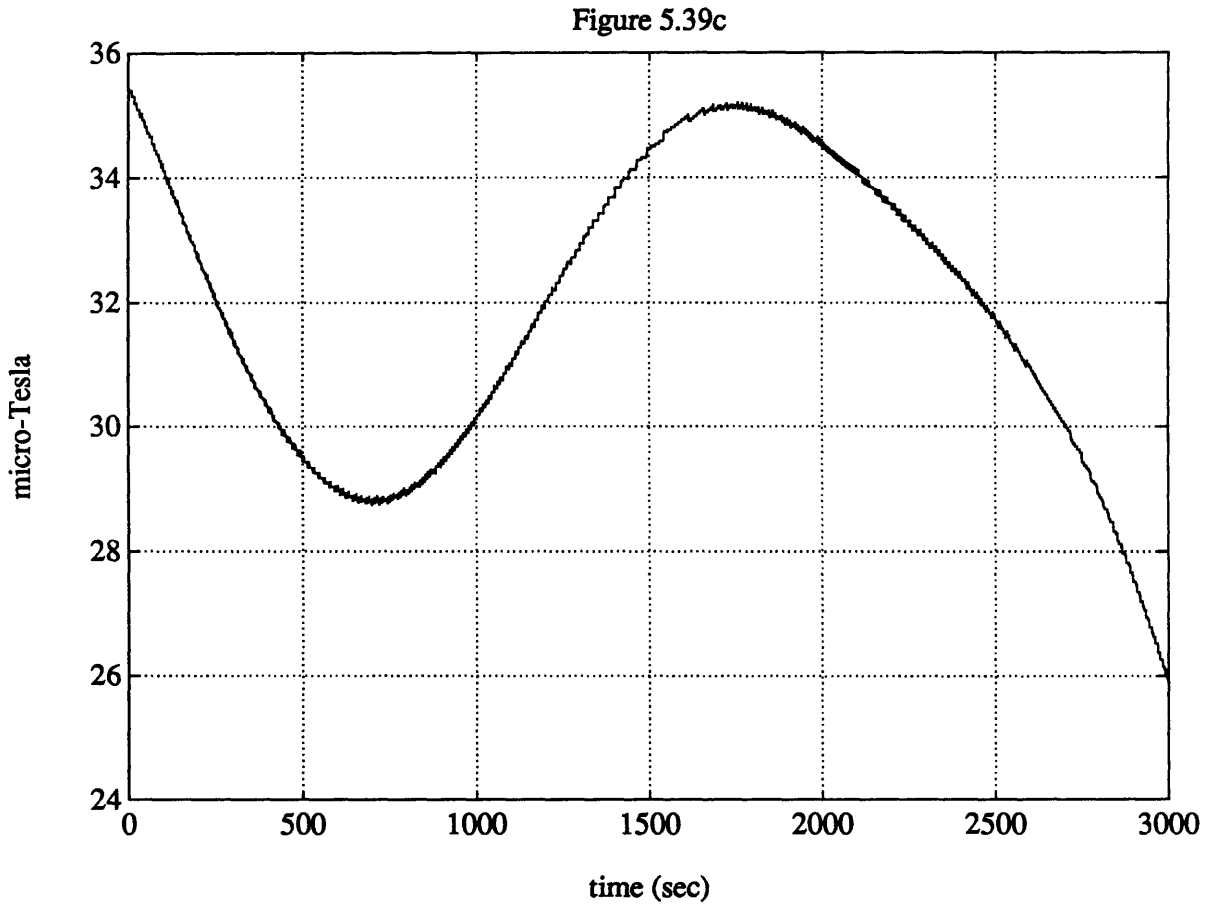
Figure 5.39c

Figure 5.39c: strength (magnitude) of the spin-plane component of the local field, again measured by the magnetometer model. Note that the values are reasonably high, making this a benign orbit on the spin-plane controller. Units are μT.
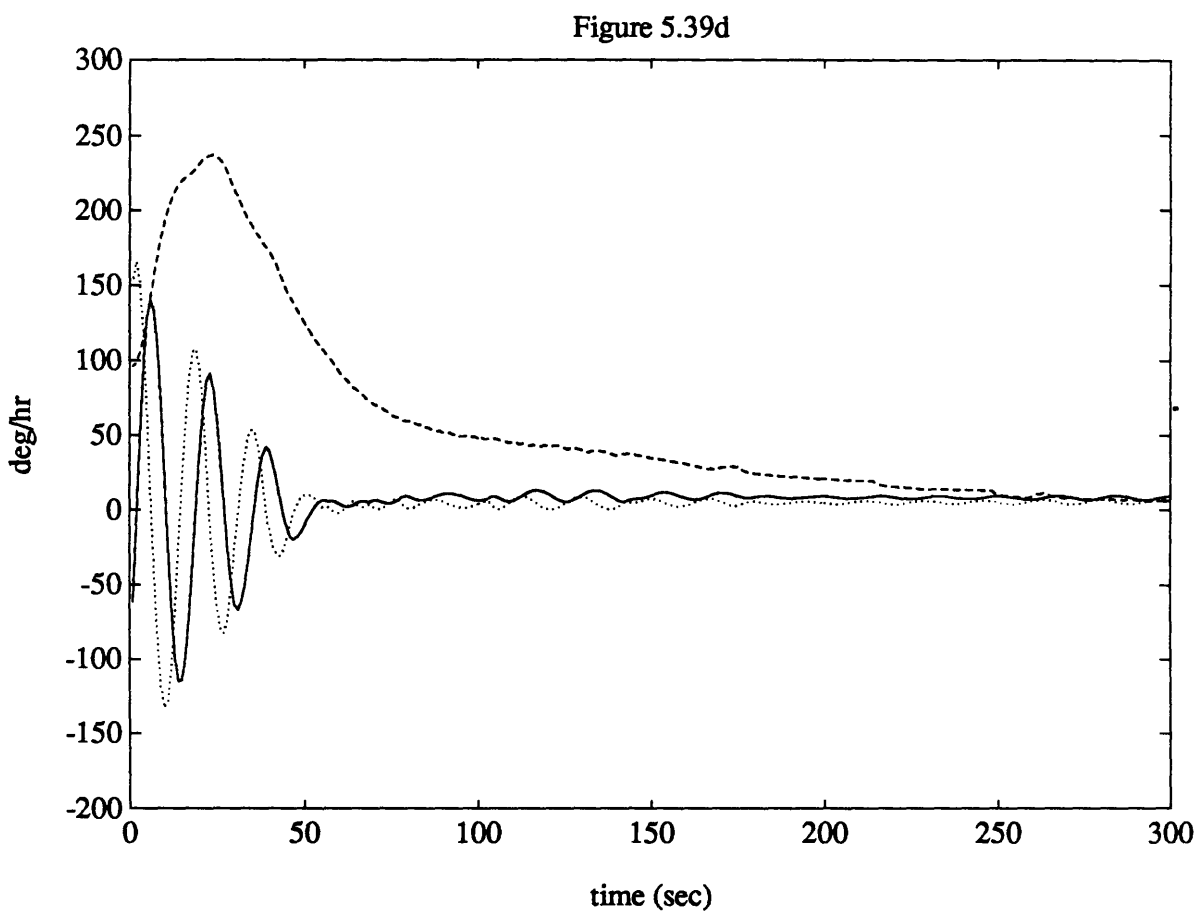
Figure 5.39d

Figure 5.39d: closeup of time 0<t<300 sec. showing recovery from large initial rates. Solid = body 1 component, dash = body 2 component, dotted = body 3 component, units are °/hr. The overshoot in ω_2 is mostly due to the controller yanking the azimuth error λ_1 to zero.

Figure 5.39e
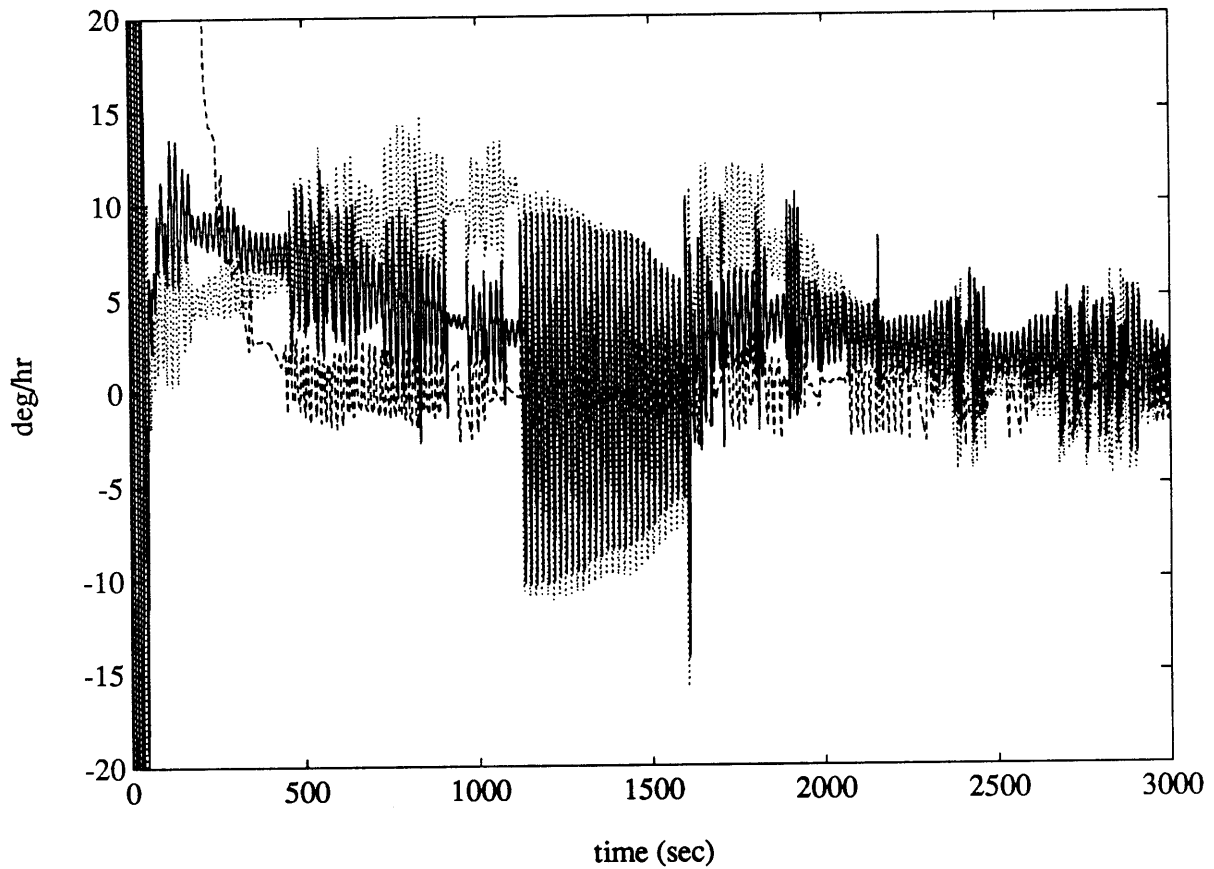
Figure 5.39e: body frame rates for full time scale. Note the period of idle from approximately 1100 < t < 1600 sec., distinguished by the mostly free nutation of ω_1 and ω_3. Transition from idling is eased by the initial value taring technique, minimizing induced transients. It is also seen, as predicted, that the sun-sensor quantization has made the limit cycles on ω_1 and ω_3 larger than those for ω_2. The RMS values are found to be in good agreement with those calculated in Section 5.3.4.
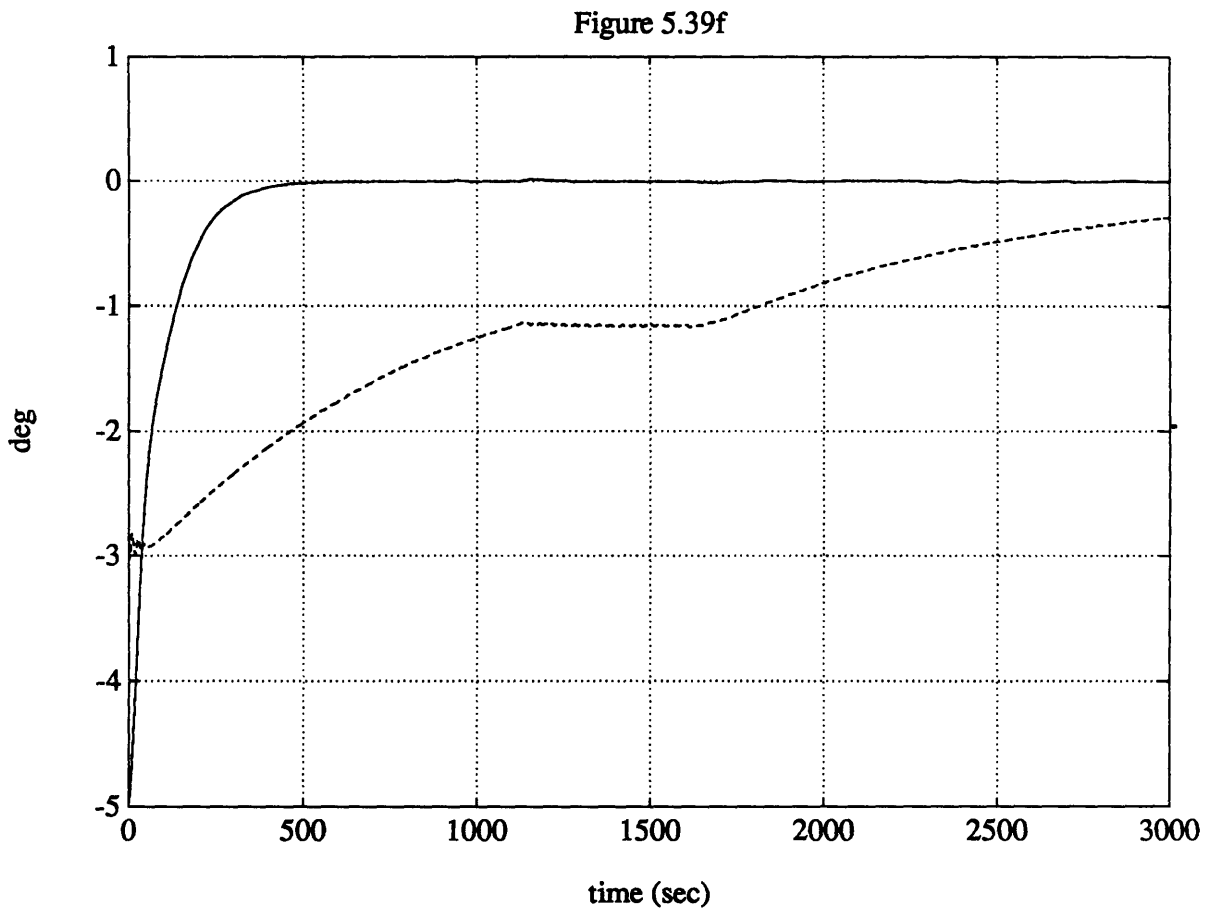
Figure 5.39f

Figure 5.39f: Processed sun sensor outputs λ_1 and λ_2, corresponding to sun vector azimuth and elevation errors respectively. Solid = λ_1, dashed = λ_2. Note the effect of the spin-plane controller idle on nulling λ_2. Sun tracking is found to be effective.

Figure 5.39g

time (sec)
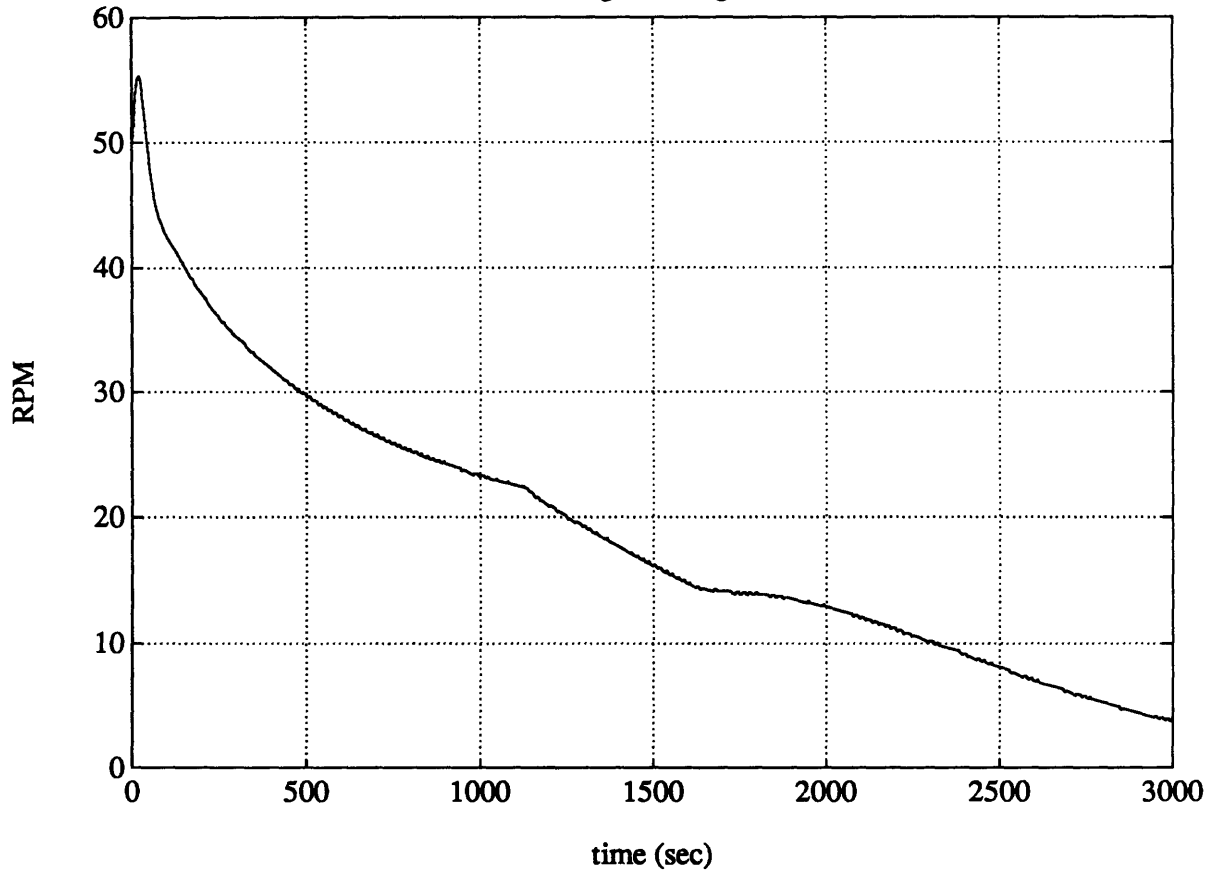
Figure 5.39g: wheel desaturation in effect- $\delta\Omega$ is brought down from 50 RPM to close to 0. Since wheel speed error is not integrated in the desaturation loop, $\delta\Omega$ will not be 0 in to presence of external disturbance torque. The desat loop gain is thus an adjustment of the $\delta\Omega$ "band" size which governs the tradeoff between attitude control and wheel speed control.
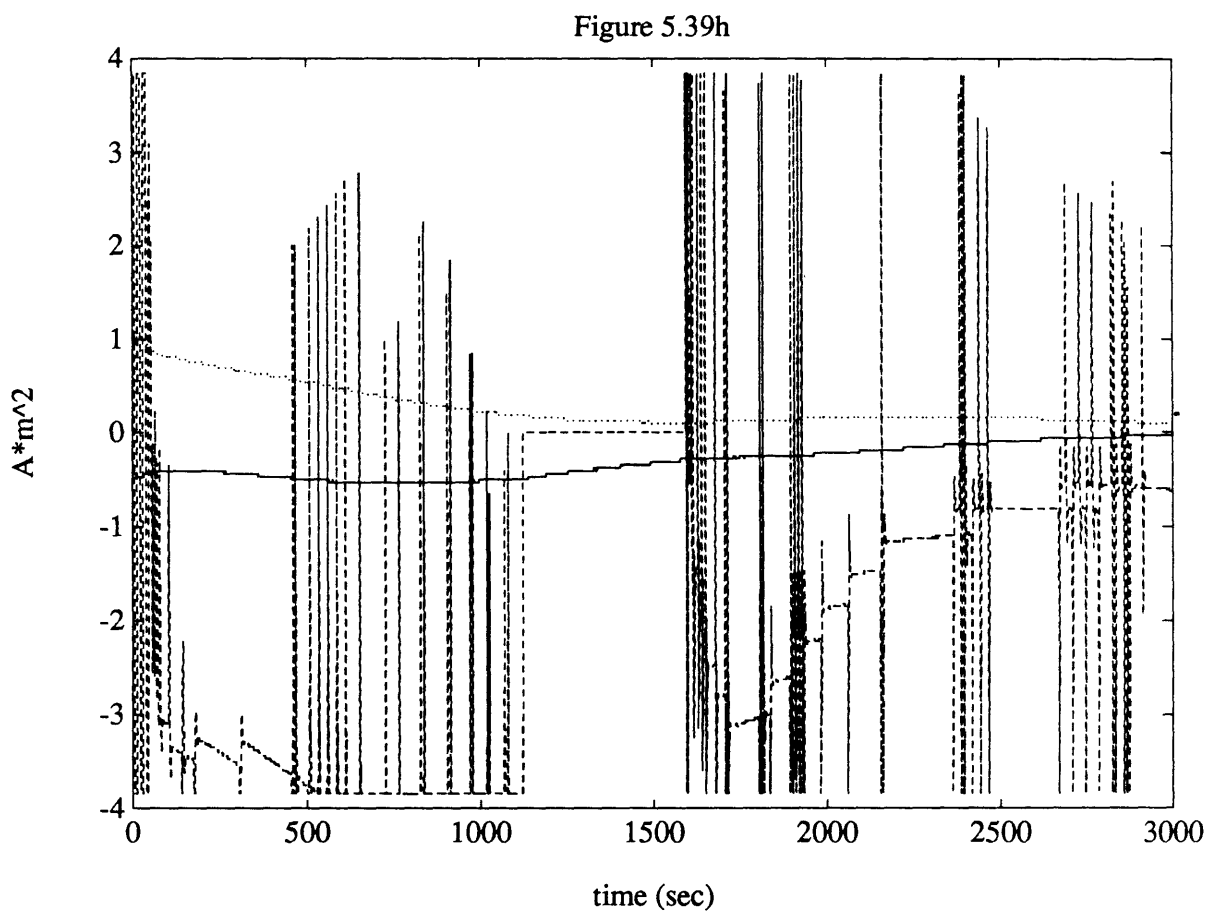
Figure 5.39h

Figure 5.39h: torquer coil activity, solid = m_1, dash = m_2, dotted = m_3, units = $A \cdot m^2$. Recall only m_2 is used for attitude control; m_1 and m_3 are desaturating. It is observed in comparison with simulations done with perfect sun sensors that the effect of sun sensor quantization is to make m_2 oscillate a lot between near 0 and short full strength spikes.
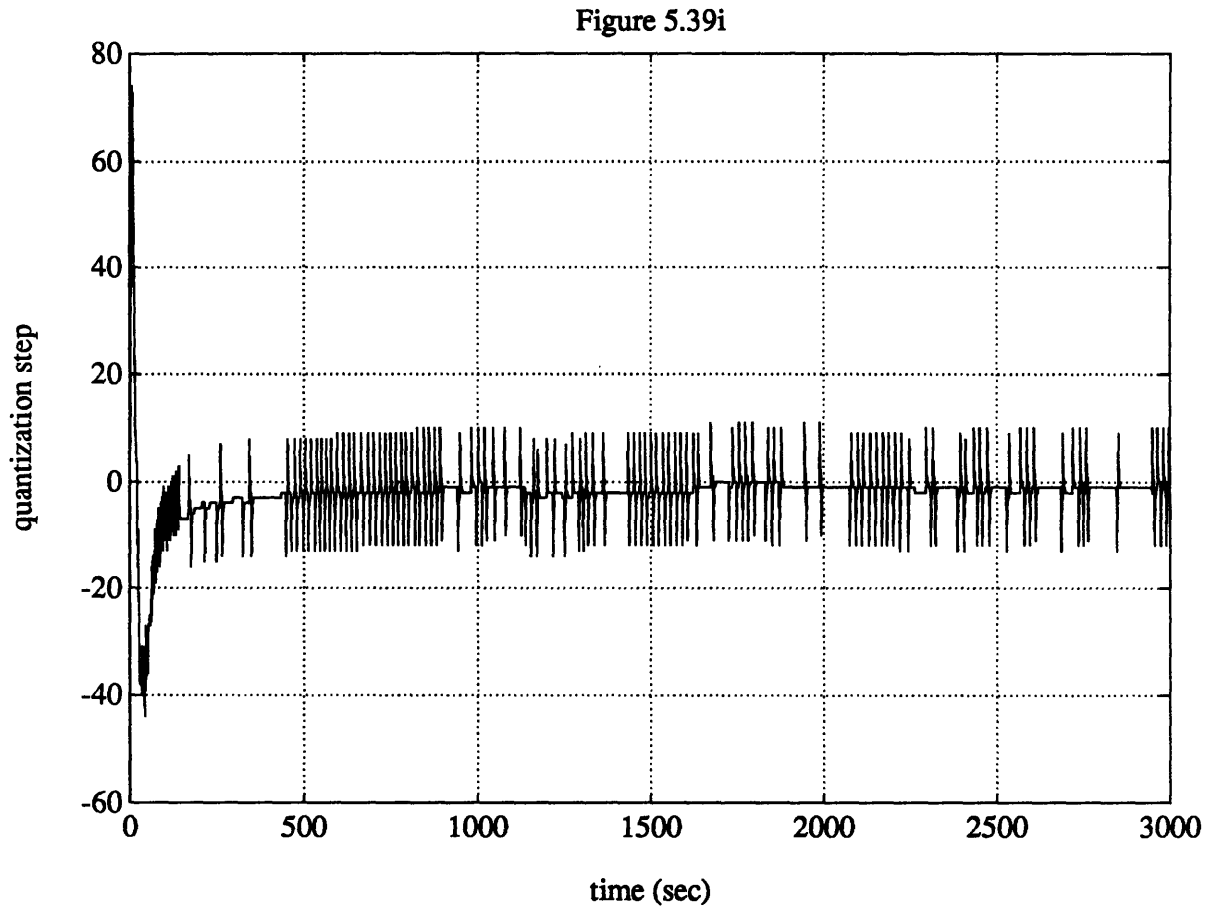
Figure 5.39i

Figure 5.39i: wheel acceleration demand in steps of 8×10^{-4} rad/s/s. Comparison with orbit night shows that spikes are larger after transients have settled because of sun-sensor quantization.

5.5.4 Examples of Post Slew Recovery for Non-Zero Azimuth

The slew terminal capture / nutation damping control schemes designed in Section 5.4 are exercised over shorter time spans. Both the partial authority actuator and full authority actuator controllers are simulated, using the same initial conditions and the same environment model. Partial authority actuator results are in Figure 40; full authority controller in Figure 41. The initial conditions used, which are again near worst case conditions following a slew, are:

$\omega = [\text{-}100\ 120\ 100]$ °/hr   principal frame rates

$[\varepsilon\ \eta] = [\text{-}0.0749\ \text{-}2.497\ 0.0783\ 1]$°   (infinitesimal) body quaternions

> (this corresponds to a rotation of 5° about an Euler axis almost aligned with B_2, but offset slightly to give some elevation error; the intent is to mimic conditions after a slew, with a rather large 5° azimuth error left over)

Other integration values are the same as before:

Simulation duration: 400 sec.
Integration time step: 1 sec.
Compensator initial values: 0

Recall that in neither controller is the effort made to correct any residual elevation error.

Figure 5.40a
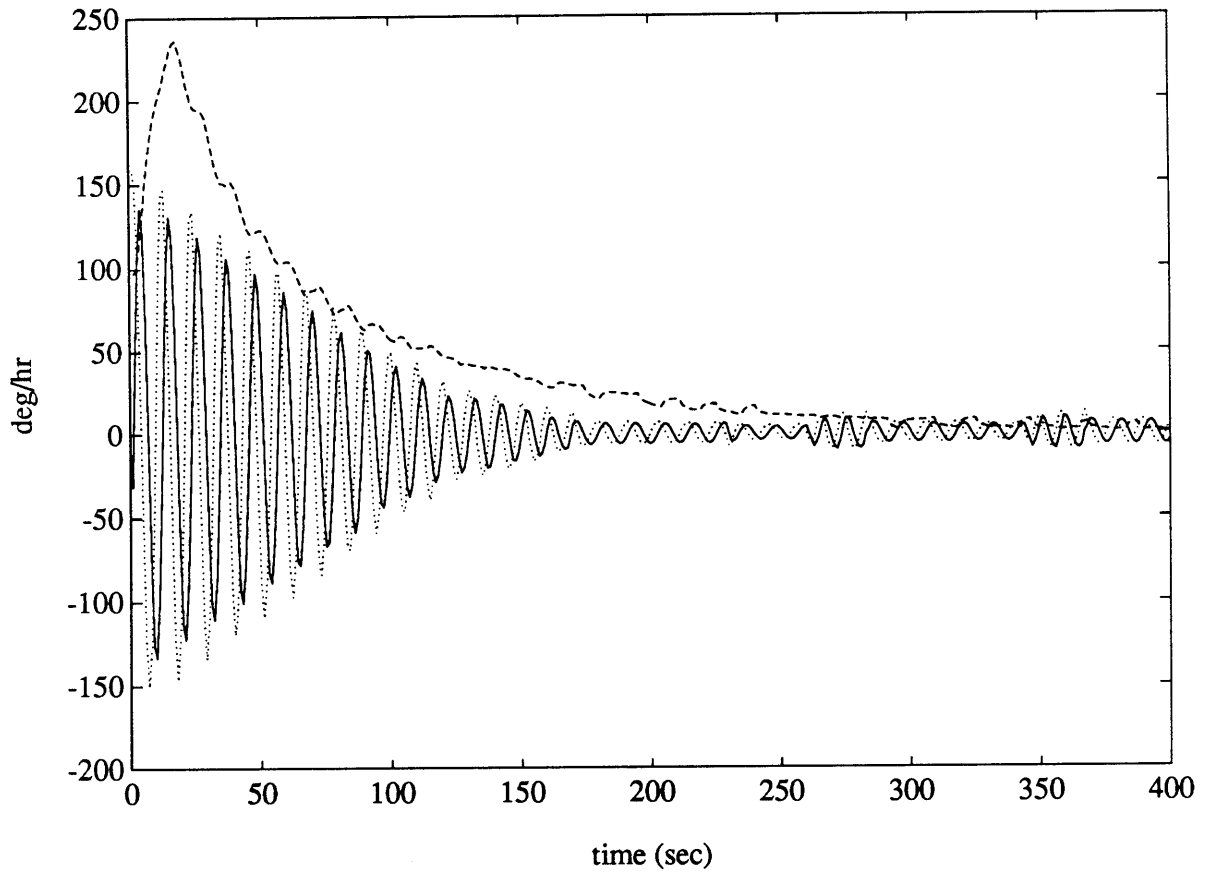
Figure 5.40a: partial authority actuator mode, rates in body frame, solid = body 1 component, dash = body 2 component, dotted = body 3 component, units are °/hr. The decay time is acceptable.
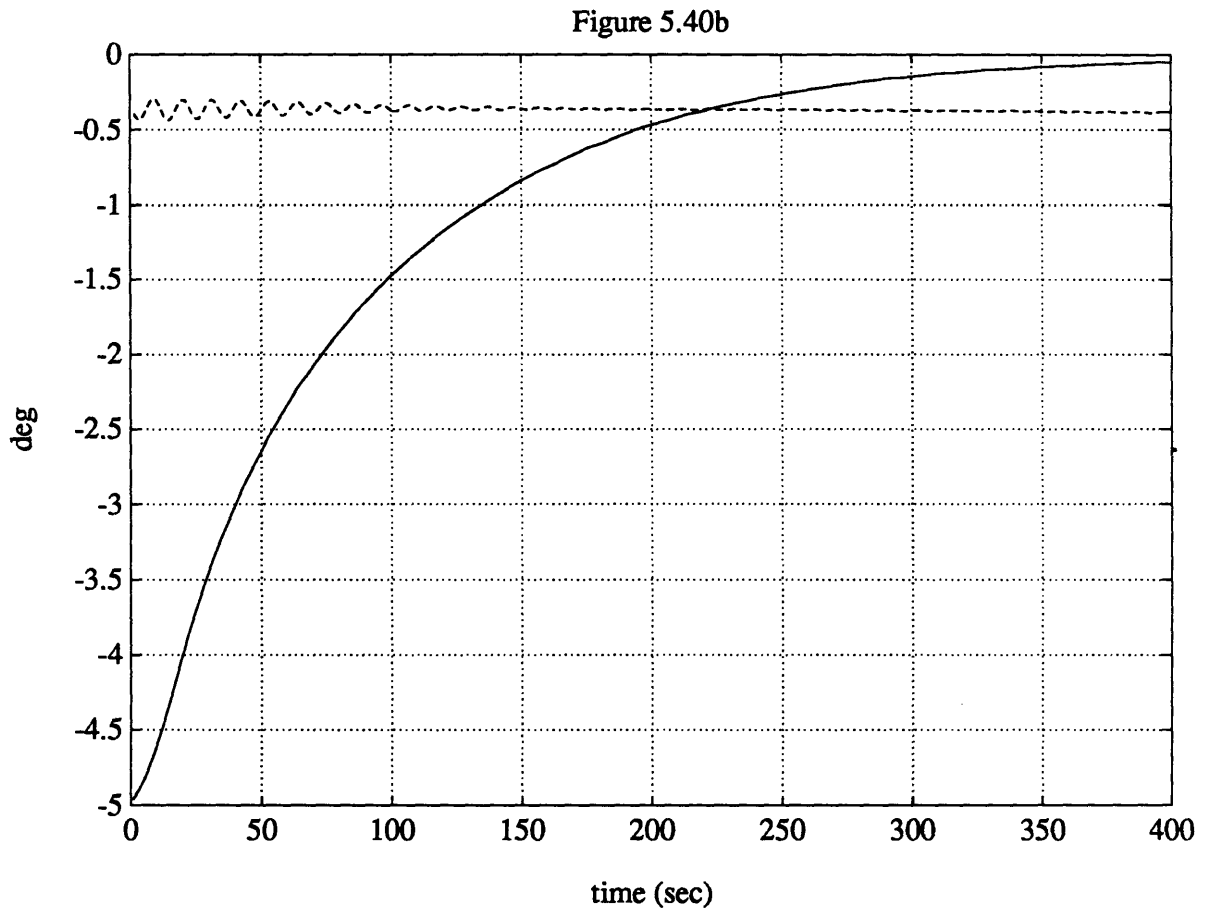
Figure 5.40b

Figure 5.40b: partial authority actuator mode, sun sensor outputs λ_1 and λ_2, corresponding to sun vector azimuth and elevation errors respectively. Solid = λ_1, dashed = λ_2. As mentioned, no attempt is made to null out λ_2. λ_1 is nulled out effectively, tracking the sensor tare function as expected.

Figure 5.40c

time (sec)
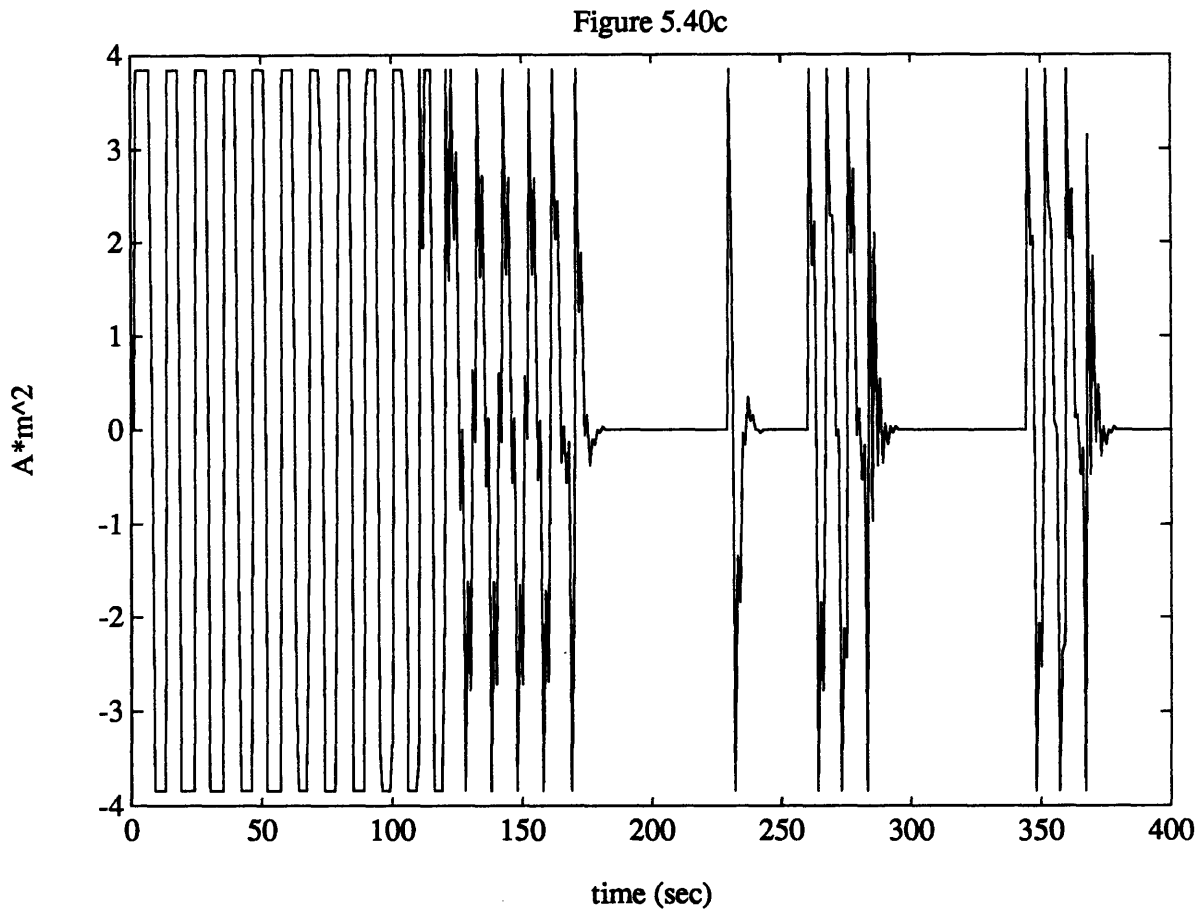
Figure 5.40c: Spin-axis coil m_2 history. Recall m_1 and m_3 are 0 when actuator mode 2 is used and the desaturation loop is turned off. The coil is seen to be mostly saturated until the initial rates are damped out. It should be pointed that in most simulations observed the coils are often saturated when damping large rates. This does cause the decay time to be longer than the linear prediction, but is still effective (many earlier satellites nutation damped using only saturated or constant current coil commands- see [Wert78]).
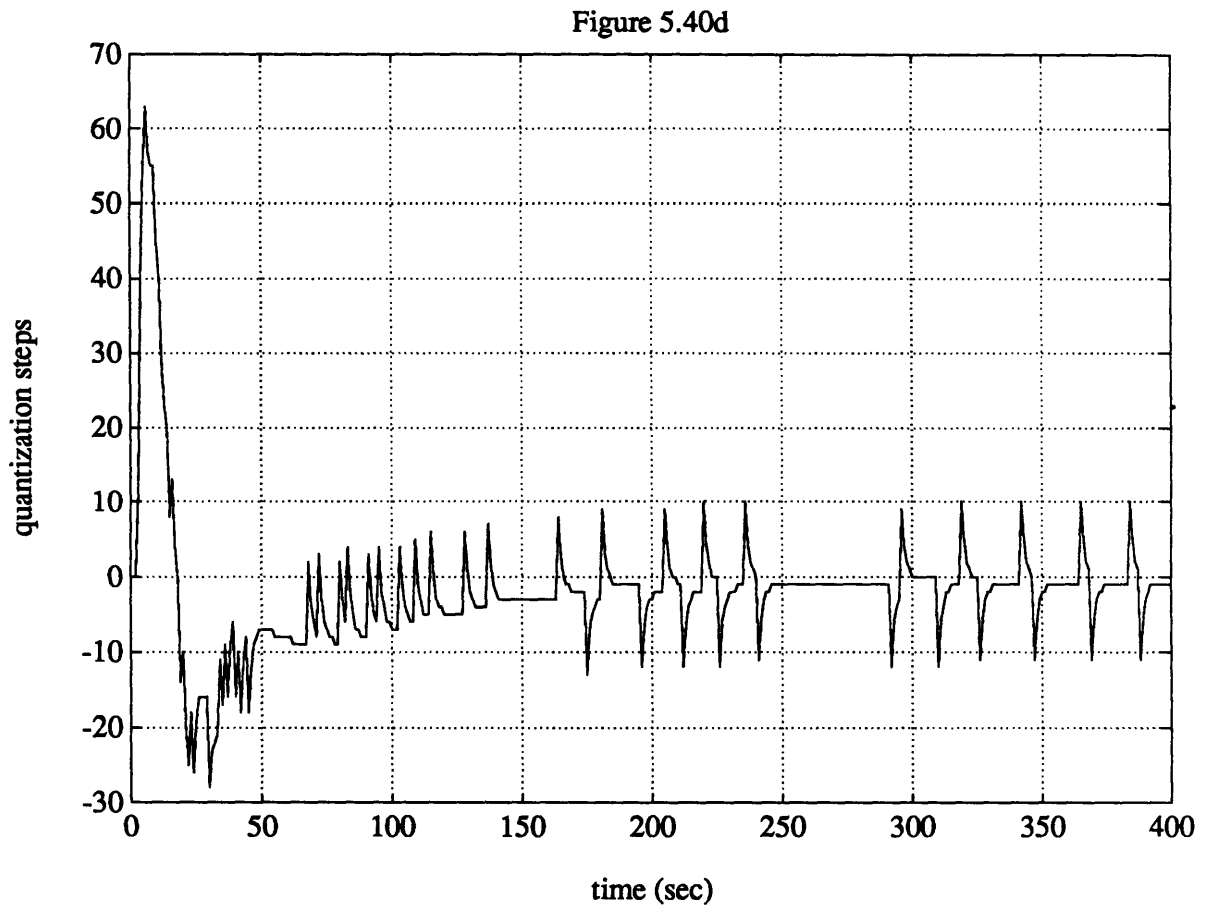
Figure 5.40d

Figure 5.40d: partial authority actuator mode, wheel acceleration demand in steps of 8×10^{-4} rad/s/s.
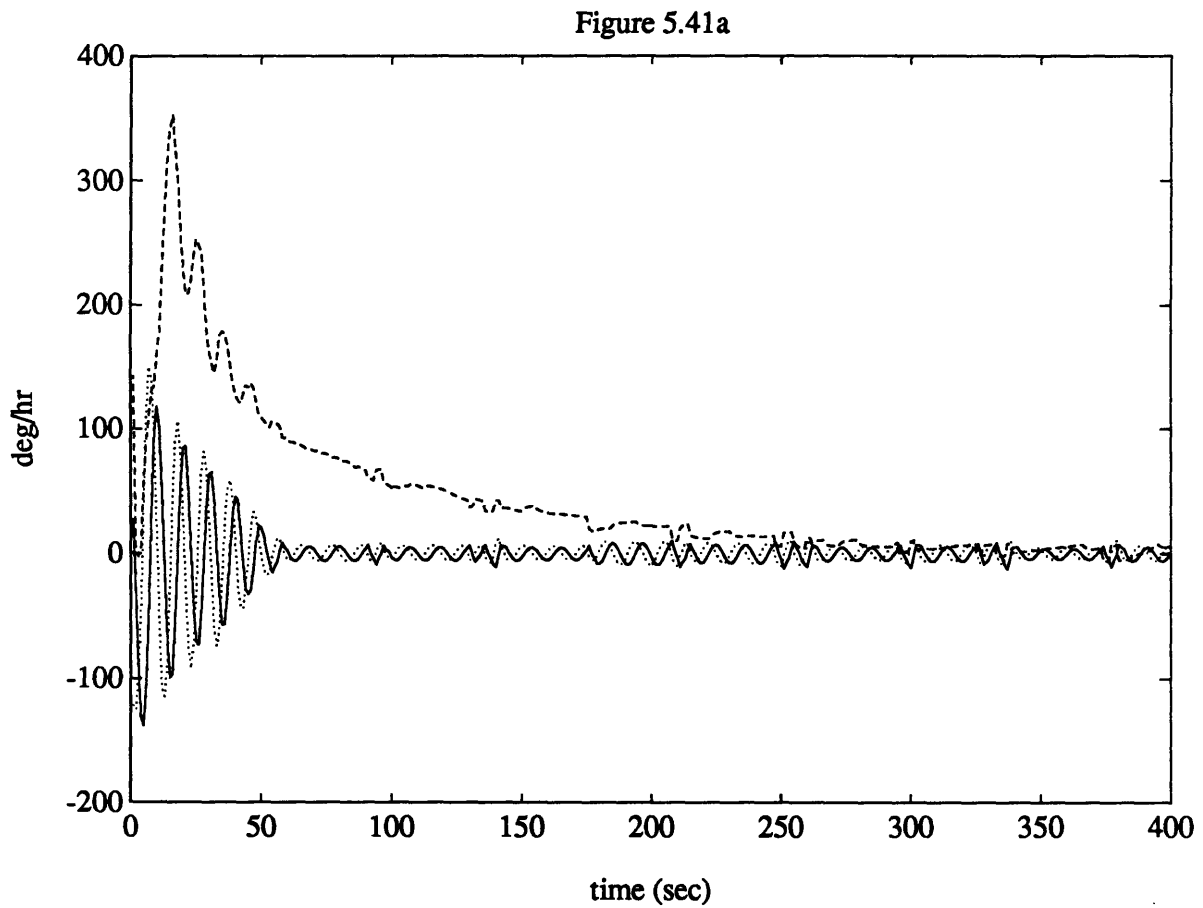
Figure 5.41a

Figure 5.41a: full authority actuator mode, rates in body frame, solid = body 1 component, dash = body 2 component, dotted = body 3 component, units are °/hr. Notice the considerably faster decay of nutation compared to the partial authority mode, but this comes at a price.

Figure 5.41b

time (sec)
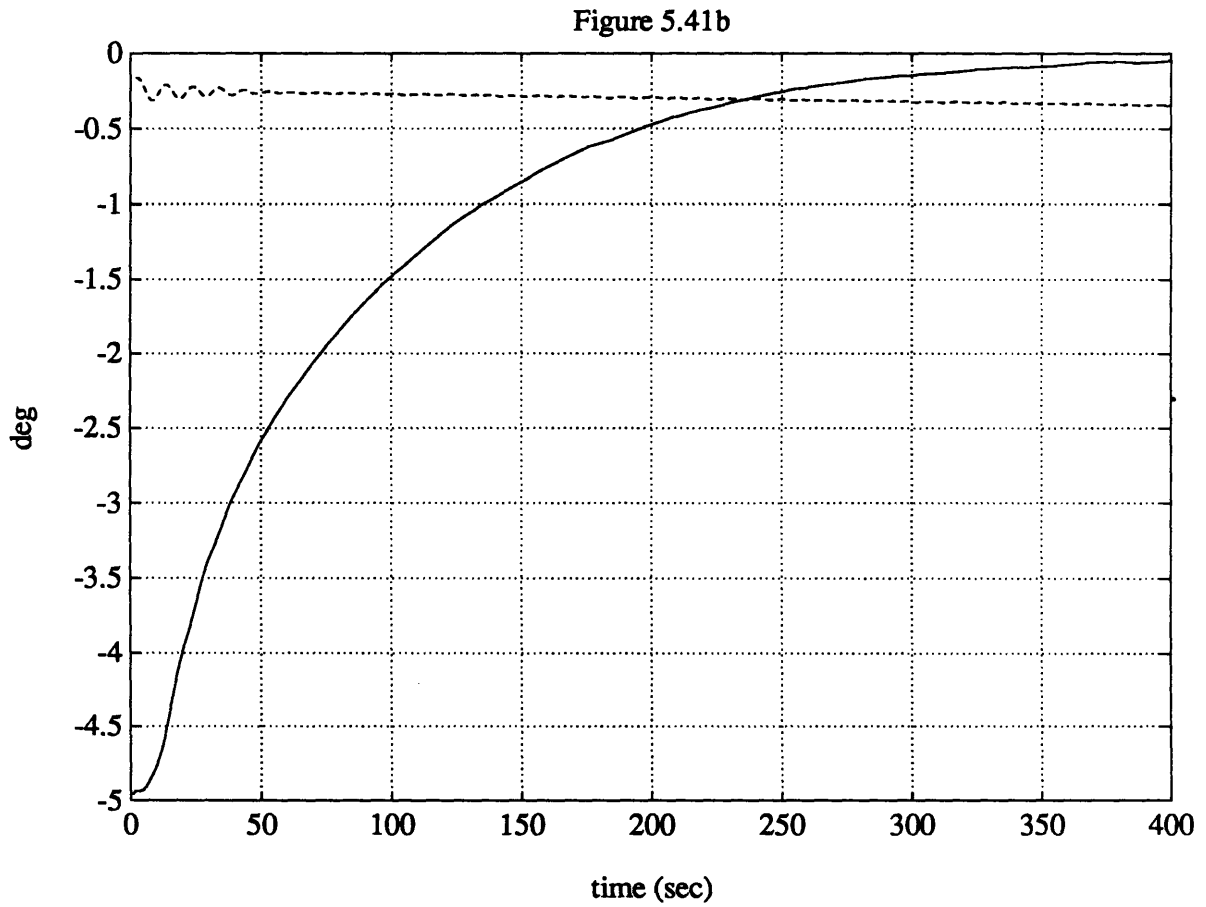
Figure 5.41b: full authority actuator mode, sun sensor outputs λ_1 and λ_2, corresponding to sun vector azimuth and elevation errors respectively. Solid = λ_1, dashed = λ_2. The same performance is observed, since the spin-axis controller is not changed.

Figure 5.41c
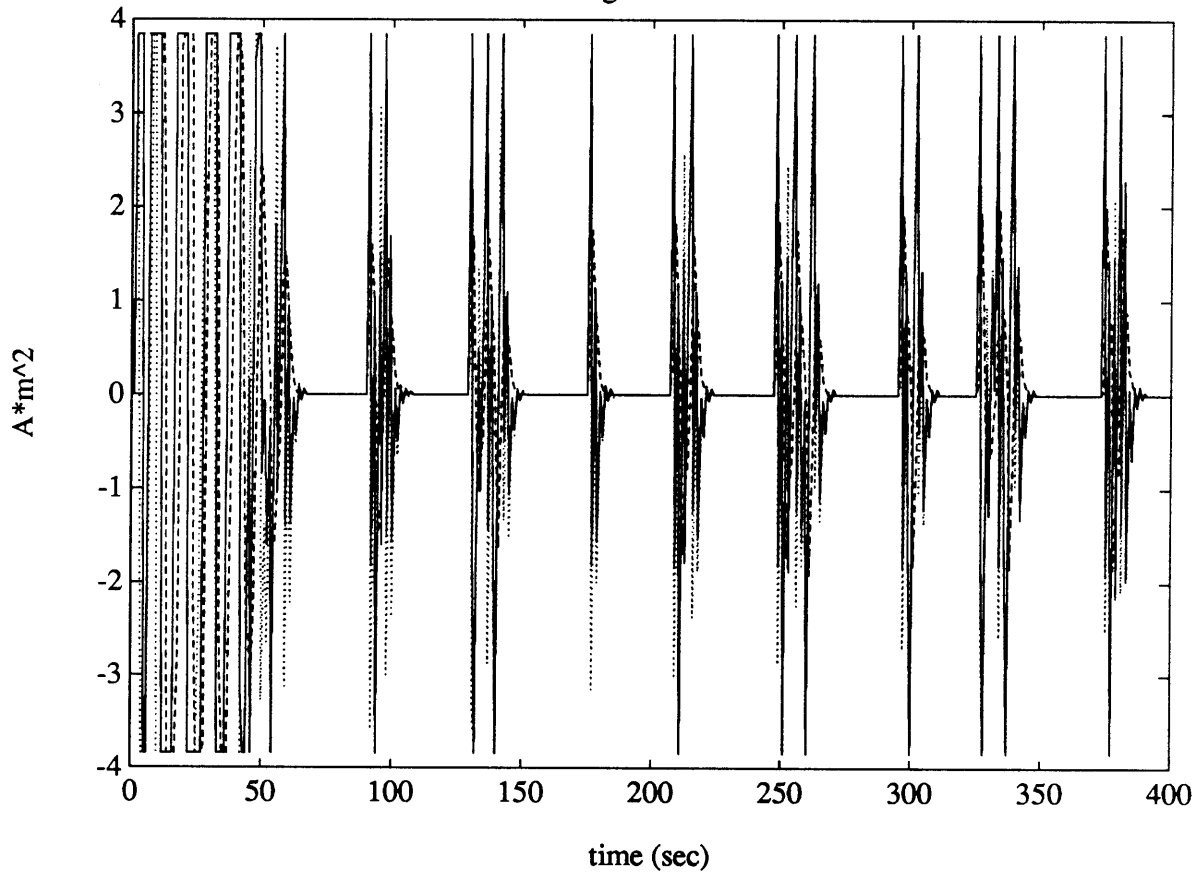
Figure 5.41c: torquer coil activity, solid = m_1, dash = m_2, dotted = m_3, units = $A \cdot m^2$. In full authority actuator mode, all three coils are used.
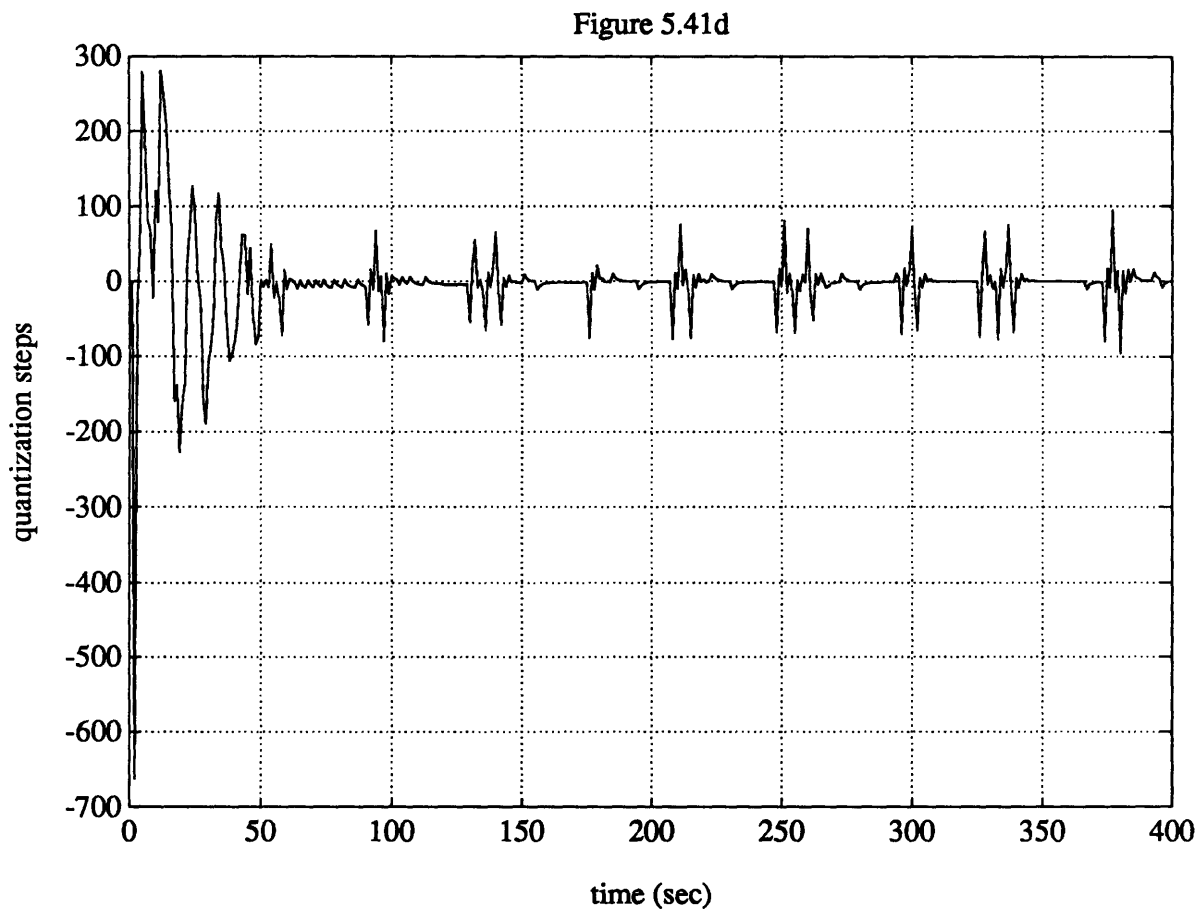
Figure 5.41d

Figure 5.41d: full authority actuator mode, wheel acceleration demand in steps of 8×10^{-4} rad/s/s. Note the considerably larger demand than that in figure 5.40d. The full authority controller comes closer to the decay time of the linear design system by wielding much larger actuator activity. In general, the full authority actuator mode should not be used unless the local Ψ prevents the use of the partial authority actuator mode.

5.5.5 Some Observations

A point mentioned several times is the observation that performance robustness is not symmetric, but rather favors greatly actual inertias being greater than the design values. Therefore, when implementing the final controller based on measured inertias, the lower bound in the uncertainty region should be used. This of course assumes that the lower bound will not cause the design plant model to have nutation poles too high to actively damp with a sample rate of 1/4 Hz, which is reasonable as it is unlikely that HETE will undergo that great a weight reduction from the current design values, and there certainly doesn't exist a need for more stored angular momentum.

It is also observed that even with conservatively high estimates of the disturbance torques, the accumulation of wheel speed in comparison to the nominal speed is small (on the order of 1 or 2%). This argues for desaturation every orbit day to keep attitude control well tuned, as the power consumption would not be great.

It is seen that the controller is fairly tolerant of principal-body frame offsets- the $10°$ offset between B_2 and P_2 seemed to affect performance very little. This is good news, as it implies rather large errors in products of inertia measurements can be tolerated.

By far the most important sensor/actuator imperfection is quantization; all other imperfections modelled are found to be insignificant compared to the effects of this. The sun-sensor resolution of $\pm 5°/10$ bits per axis used is quite achievable, as is the ± 4 A$\bullet$m^2 / 8 bits current control on the coils (most likely PWM control) [Pers91]. The 5 μN$\bullet$m per step control of wheel torque, however, will require careful design. It is seen from both analytical prediction and simulation that fine control of the wheel torque is necessary to meet the tight rate specifications. Should the achievable torque resolution be higher, the bandwidth of the controllers should be adjusted so as to limit rate response.

It is noted that the regulator gains exhibit nearly sinusoidal shapes as functions of θ in the gain-scheduled controllers. This suggests the possibility that some sinusoidal input/output transformation could have rendered the plant LTI, arriving at the gain schedules in one controller design. This has not been investigated, as the less elegant approach pursued in this thesis of solving many LQG problems is trivial with available software.

The simulations included in this thesis represents a small portion of the total which have been run. In most, the performance has been consistent with the good results presented. However, on rare occasion the orbit day controller as designed will suffer from a protracted bout of coil saturation at one rail, despite the anti-windup strategy employed. Figures 5.42a & b show an example. Though its occurrence is rare, it should be noted and corrected for in the full implementation. The cause of the extended saturation is consistently observed to be a divergence between the tared elevation error command and the achieved elevation error, usually caused by a period of insufficient field strength in the spin plane near the beginning of the controller's operation. This is shown in Figure 5.43. The design philosophy take has not been to specify large (i.e. heavy and power hungry) magnets to brute force through the specifications, but to design aggressively for performance using small magnets which would be inadequate for arbitrary conditions, then tackle the saturations by taking advantage of the specific requirements on the controller. It is always observed that the controller performs well at damping rates and nulling the elevation error as long as the elevation error is small enough. The current implementation of a first-order decaying tare signal on the elevation error is the simplest possible, and is effective for most situations. A slightly more elaborate scheme where the tare signal is sensitive to both long-duration coil saturations and growing divergence between commanded and achieved elevation errors should take care of the observed problems. Unfortunately time did not permit testing this idea.
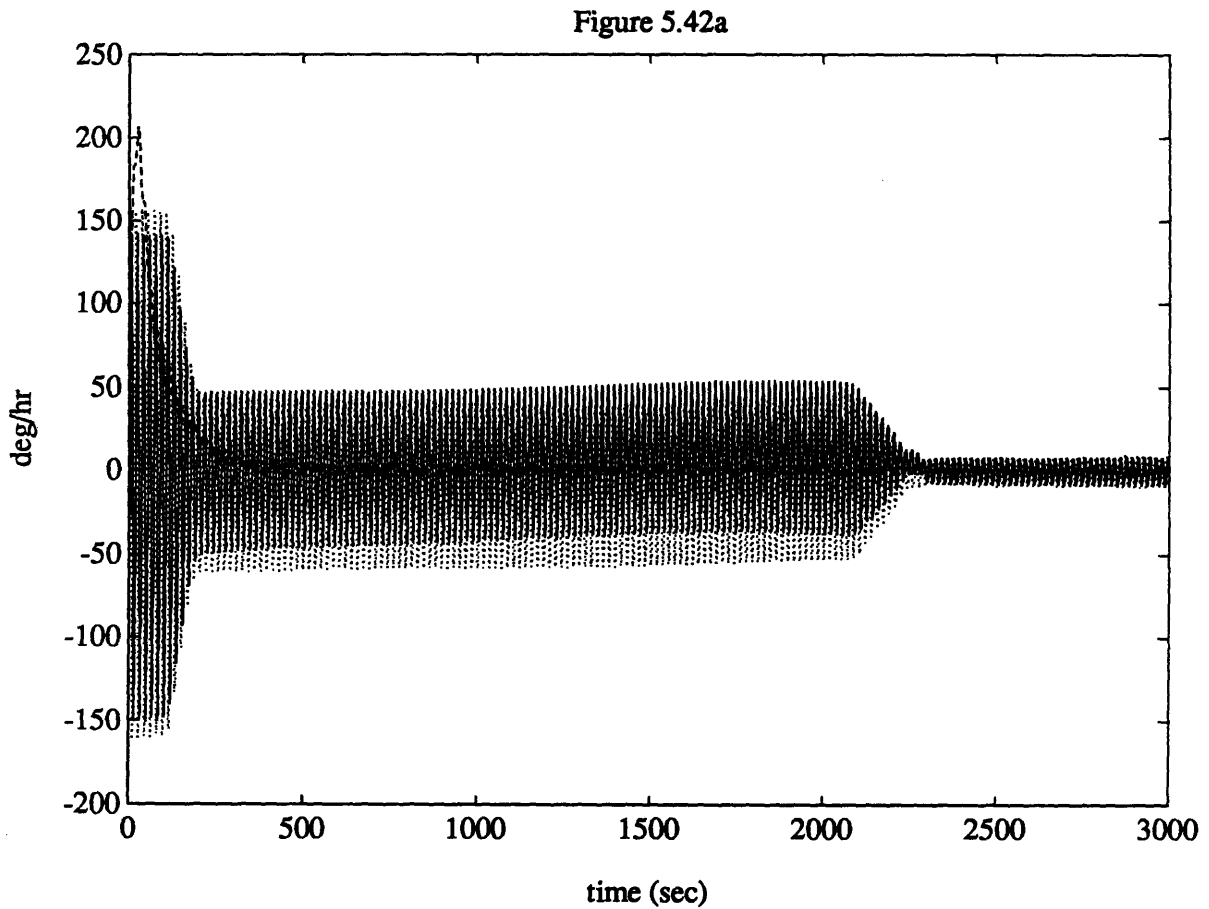
Figure 5.42a

Figure 5.42a: a case of orbit day rate damping from large initial rates stalled by persistent coil saturation. Note recovery does not occur until about 2100 sec., which corresponds to the elevation error λ_2 being reduced- see Figure 5.43.
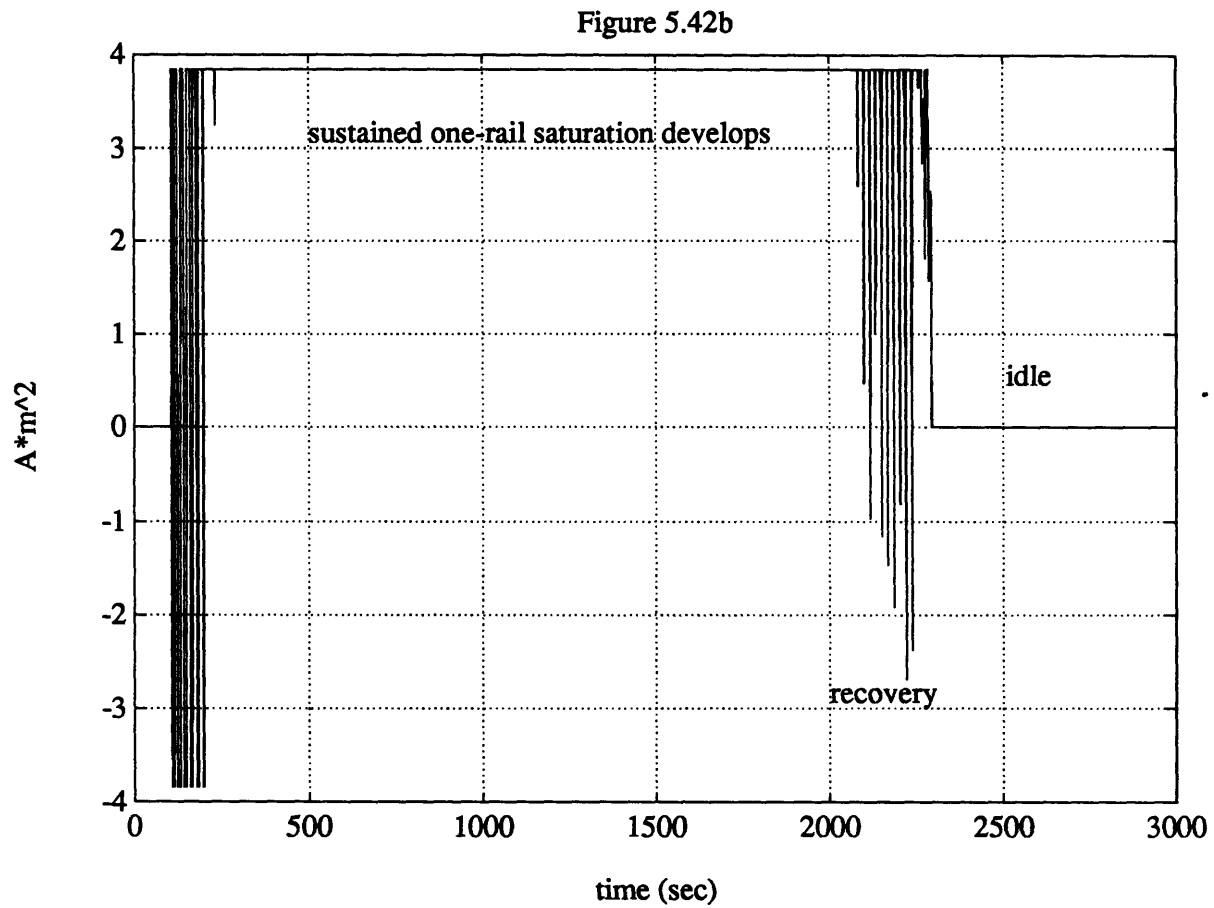
Figure 5.42b

Figure 5.42b: associated spin axis torque coil commands, with the observed large one-rail saturation. Note that for this particular example, just as recovery begins, the spin-plane controller is commanded into idle because the spin-plane field component becomes too small. This constitutes a good example of worst case, and was in fact a reason for selecting this example. Note also that recovery begins when the λ_2 error is small, as seen in comparison with Figure 5.43 below.
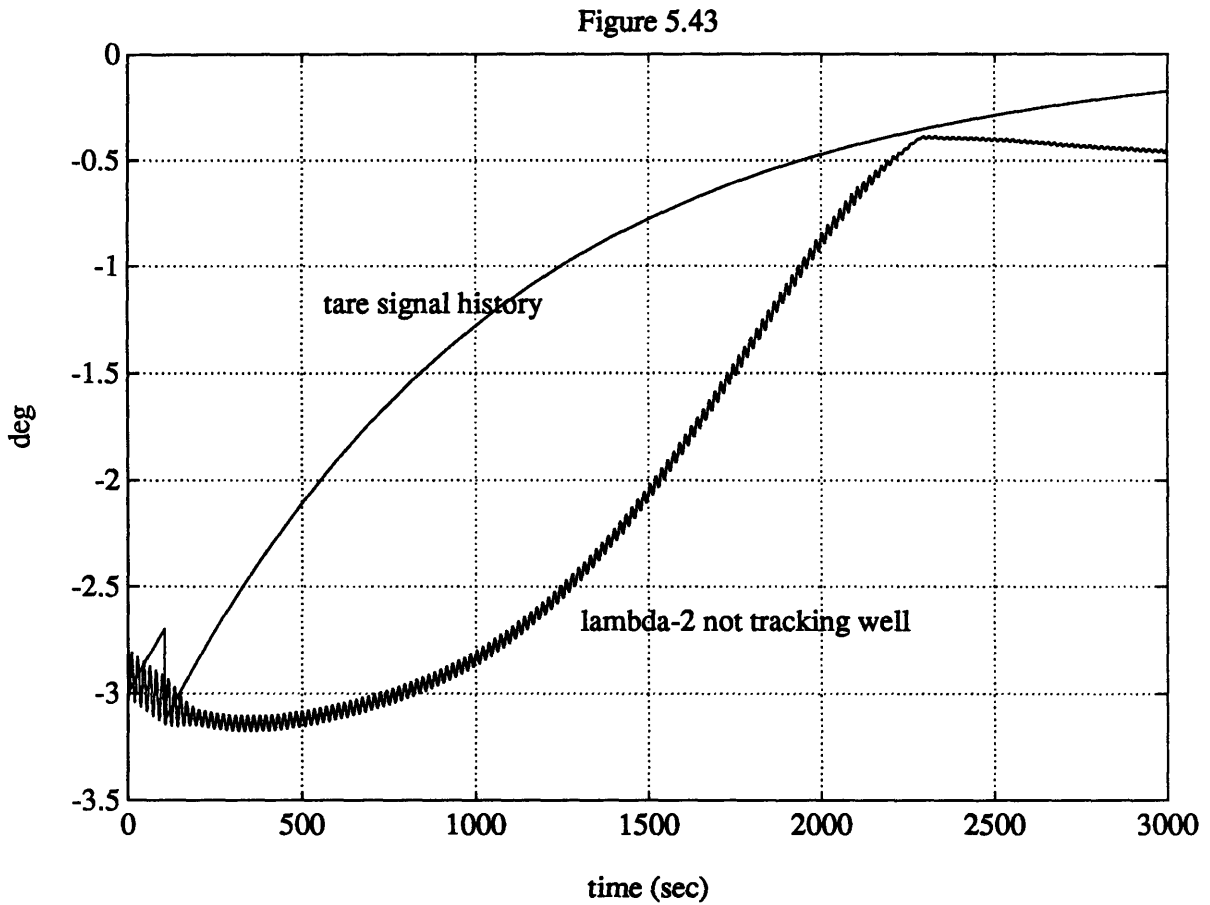
Figure 5.43

Figure 5.43: the λ_2 tare signal and λ_2 response, showing the divergence developing early. The fed-back elevation error is the difference of the two signals. The spin-plane controller is acting as it is designed to- closing the gap of its elevation error. It is always observed that rate recovery begins when the elevation error is closed sufficiently small. Therefore, a more intelligently implemented tare generator sensitive to the situation here would solve this problem.

VI. Suggestions for Further Work

6.1 Regulating the Spin-Axis

It has been noted that the controllers designed in Chapter 5 ignore the first part of the slew maneuver- the need to achieve and maintain a desired inertial orientation of the spin-axis (the sun-fixed frame $\mathcal{F}_S$ would be a convenient reference) so as to slew the instrument axis ($-\mathbf{B}_3$) to a desired direction. This section suggests strategies for implementing this control and argues that no new hardware need be added (unless higher performance is desired). Detailed analysis has not been performed on the suggested strategies; hence the chapter title.

To facilitate discussion, we define the angle ρ as the angular error of the sun axis about $\mathbf{B}_3$. It should be made clear that the current definition for the orbit day quaternions $[\varepsilon_1 \; \varepsilon_2 \; \varepsilon_3 \; \eta]^T$ does not account for ρ. Recall Section 4.2.2, where the sun sensor / vehicle kinematic relations were derived. Only 2 degrees of freedom were specified in defining a desired sun-vector azimuth and elevation, the mathematical manipulation being the choice of a 2-1-3 Euler angle sequence to connect the sun-sensor-measured reference frame and the desired reference frame. Recall that the third Euler angle was assumed to be zero; that ignored third rotation is, of course, ρ. Clearly, signals from a single sun-sensor do not measure ρ.

The first thing to note is that ρ need not be nulled to 0 very tightly. Since the instruments are all wide field of view, errors in ρ, say $\pm 5°$, will not make a difference unless the object to be observed would have lied at the edges of the slewed FOV, in which case the slew maneuver itself should be re-specified. The second is that it would be desirable to actively control ρ during orbit day only, again stressing the importance of maintaining low body rates during orbit night for high-quality observations. Having said so, the two issues to examine are, as usual for sketching control design, sensor and actuator strategies.

6.1.1 Sensor Strategy

Do we need another sensor? Clearly, ρ is not observable during orbit day with sun-sensors. It is observable during orbit night, where the star-ID algorithm can calculate the transformation between $\mathcal{F}_B$ and $\mathcal{F}_S$ easily. Therefore, at the beginning of every orbit day, ρ

is known. We can estimate the worst amount of "damage" a regulator on ρ can do operating blind for an entire orbit day. The physical relationship is:

$$\left|\frac{d\rho}{dt}\right| = \frac{|\tau_1|}{h_w} \qquad (6.1)$$

where τ_1 is the control torque component along $\mathbf{B_1}$, which through gyroscopic coupling generates rotation about $\mathbf{B_3}$. Wheel momentum bias is $h_w = 2\ N\bullet m\bullet s$, as used in Chapter 5. If we make the reasonable assumption that the local field direction permits an average of 25% of the net control torque magnitude to be pointed along $\mathbf{B_1}$, and assume the following still conservative numbers:

average mag. moment:	$4\ A\bullet m^2$
average field strength:	$30\ \mu T$

we get $|d\rho/dt| = 8.5 \times 10^{-4}$ °/s, which over 40 minutes is $\Delta\rho = 2°$, which, even in the very unlikely case of being a factor of 2 too small, would still be acceptable. Therefore, the potential limit cycles arising from controlling ρ blind every orbit day is acceptable, arguing no additional sensors are needed. An appropriate sensor strategy is to observe ρ during orbit night with the UV instrument and correct for it essentially open-loop during the day.

It is noted that the argument rests on assuming weak magnets, which leads to the more sensible interpretation of the argument as this: operating blind to ρ during orbit day is fine as long as one is willing to correct ρ slowly compared to twice orbital rate.

6.1.2 Actuator/Control Strategies

We begin with a review of the various decouplings taken advantage of in Chapter 5 to see what needs to change. The decoupling between the spin axis and the spin plane attitude controllers depends only on mass properties and can clearly be retained. The decoupling between the spin-plane controller and the desaturation loop is more subtle, and rests on the assumption that the desaturation loop commands the spin plane magnets m_1 and m_3 such that the disturbance torque they produce in the spin-plane is always perpendicular to the direction on the spin plane along which the spin-axis coil m_2 can generate a control torque (see Section 4.1.4, Section 5.3.6, and Figure 4.4). The spin-plane

control loop commands only the spin-axis coil m_2. Therefore, the desaturation loop and spin-plane loop need not be aware of each other's existence.

The current design has neither loop actuating with any concern for ρ. Two options are suggested for introducing ρ error knowledge:

"ad-hoc" option

The idea is to maintain the explicit decoupling between the spin-plane loop and the desaturation loop by using the same $[m_1 \ m_3]$ coil commands as those given in equation (4.15). The spin-plane loop also uses the same design, with gains scheduled on θ, λ_2 errors tared to prevent hard saturations. The presence of a ρ error, calculated at the end of orbit night, establishes a preferred half-plane on the spin plane for actuation torques. Figure 6.1 illustrates a generic case:
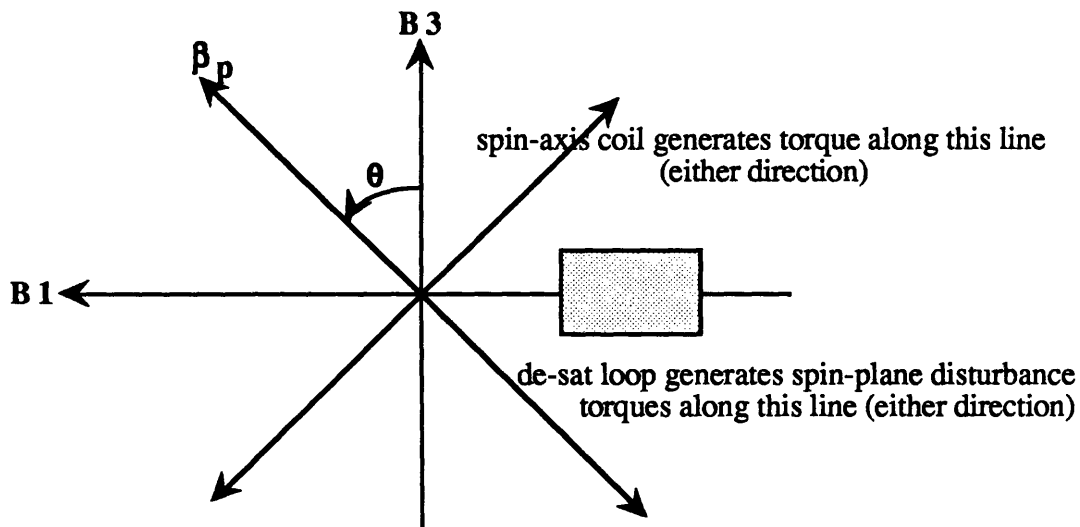
Torques generated in this half plane is favorable (moves ρ in the right direction).

β_p = projection of local field onto spin plane

Figure 6.1: view of the body fixed spin-plane B_1-B_3 showing the generic case where torques generated in the shaded half-plane results in reducing ρ. The closer a torque is to the B_1 axis, the larger $|d\rho/dt|$. The achievable spin-plane torque directions are also shown, given a generic β_p vector.

Currently the spin-plane loop and the desaturation loop command torques without regard to the half-plane which they would lie. We could augment the loops by weighting their authority according to the measured θ and the desired half-plane. The issue of stability can be skirted if the "weighting" is accomplished outside the feedback loop by intelligent taring of the λ_2 feedback and the wheel-speed error feedback, since this is simply changing regulator set points.

The ρ correction effectiveness of this scheme requires further analysis, but it seems likely at the outset that the effectiveness would be small, since ρ only establishes a desired direction, and the "actuating" signals are really wheel speed and λ_2 errors. This may be a suitable strategy for maintaining a desired spin-axis orientation, once it has been achieved.

"expanded LQG" option

Another possible approach is a variation on the full-authority actuator scheme used during orbit night, with the additional twist of including the wheel speed error as an observable state variable to be nulled as well. Furthermore, since it is still true that λ_2 contains no ρ information, an estimator for ρ, initialized at the end of orbit night and running open-loop (no innovation signal) for the duration of orbit day, provides a ρ "feedback" signal. The idea is to abandon the decoupling of the spin-plane and desaturation loops to coordinate control torques from the wheel and all three magnets. The entire system, with λ_2 and wheel error $\delta\Omega$ as feedback measurements and θ as a scheduling measurement, is lumped into an LQG regulator which attempts to null all the state variables. We still maintain the decoupling of the spin-axis attitude loop- it simply takes in sun-sensor measurement λ_1 and commands the wheel.

The physics of the problem hasn't changed, so the increase in system order must be accompanied by an increase in the number of pole/transmission zero cancellations (uncontrollable modes). This is a statement that given an arbitrary initial attitude, wheel momentum state, and local field direction, all the state variables in the expanded system can't be controlled independently. Therefore, this formulation of the problem is simply a more formalized design method to implement a controller which is aware of all the variables we wish to null.

The effectiveness of these suggestions must be evaluated further to see if their performance is acceptable. "Acceptable performance" on precessing the spin-axis should

also be better defined in consultation with the science team. It would simplify things greatly, for example, if only a single desired spin-axis orientation need be maintained.

6.2 Attitude Acquisition

Following insertion into the proper orbit, HETE must de-tumble, acquire the sun-vector, and spin up the momentum wheel. It is noted that the fine body-rate control specifications for the observation phase of the mission do not apply here- we will assume that attitude information from the UV instrument is not available at this point anyway. Spin-up is fairly straightforward- reverse the desaturation discussion in Chapter 4. De-tumble and sun acquisition can be accomplished with a standard magnetic control maneuver which is not necessarily optimal but is fairly robust and simple to implement. [Stic74] suggests the following control law from magnetometer measurements to torque coil moments:

$$m_i = -k\dot{\beta}_i \qquad k \text{ const} > 0, \quad i = 1,2,3$$

$\dot{\beta}_i = $ magnetometer derivative along body axis i $\qquad\qquad$ **(6.2)**

$m_i = $ coil moment along body i

The control law is seen to oppose change in the measured field vector, and a proof based on kinetic energy arguments shows that the control law is globally stable in bringing spacecraft rotation to correspondence with field rotation, which is approximately orbit rate. This is true regardless of whether the gain k causes the coils to saturate.

If the spacecraft has a momentum bias, [Coll73] and [Stic74] show that the same control law (6.2) de-tumbles the spacecraft and also places the spin-plane on the orbital plane. [Stic74] also references a stability analysis which indicates that for the maneuver to be stable, the condition:

$$h_w + 2I_2\omega_0 > 2I_T\omega_0 \qquad\qquad \textbf{(6.3)}$$

must be satisfied. I_T is the larger of I_1 or I_3, the transverse inertias, and ω_0 is orbital rate. Condition (6.3) is quite easily satisfied, since $2I_T\omega_0$ represents only 0.6% of the nominal momentum bias h_w for the observation mission phase.

Since the spacecraft spin-plane is now on the orbital plane, at the intersection of the orbit plane with the ecliptic plane the elevation error is zero. The spacecraft orbit intersects

the ecliptic plane at two points, one in orbit night, the other in orbit day. At the latter point, a rotation about the wheel axis- which can be done in a direction that also brings wheel spin-up closer to completion- can null the azimuth error. The spin-axis loop can then be activated to lock on to the sun azimuth, while the spin-plane loop- using an LQR law designed without λ_2 weighting like that used for post-slew recovery, can begin the gross nutation damping. When rates have been settled and the design h_w has been achieved, the regular spin-plane loop can be turned on to gently null out the elevation error. Completion of the final ACS milestone- transitioning to the orbit night controller- is then governed by the battery state and the success of instrument power-up and checkout.

References

[Coll73]   Collins, D.H. and Bonello, D.P., "An Attitude Control System for Earth Observation Spacecraft." AIAA paper 73-854.

[Conc69]   Concus, P., Crane, G. E., Satterlee, H. M.: *Small Amplitude Lateral Sloshing in Spheroidal Containers under Low Gravity Conditions*. NASA CR-72500 / Lockheed LMSC-A944673, February 1969.

[Fran80]   Franklin, Gene F. and Powell, J. David: *Digital Control of Dynamic Systems*. Addison-Wesley Publishing Company, Menlo Park, CA, 1980.

[Hugh86]   Hughes, Peter C.: *Spacecraft Attitude Dynamics*. John Wiley and Sons Inc., New York, 1986.

[IAGA86]   Barraclough, D. R. and Kerridge, D. J.: *IGRF 1985 Grid Point Values and Charts*. IAGA Bulletin #52, International Union of Geodesy and Geophysics, Paris 1986.

[Kapa88]   Kapasouris, P., Athans, M., and Stein, G.: *Design of Feedback Control Systems for Stable Plants with Saturating Actuators*. MIT LIDS report number LIDS-P-1756, March 1988.

[Kwak72]   Kwakernaak, Huibert and Sivan, Raphael: *Linear Optimal Control Systems*. Wiley-Interscience, New York 1972.

[Lern79]   Lerner, G.M. and Shuster, M.D.: "Magnetometer Bias Determination and Attitude Determination for Near-Earth Spacecraft." AIAA paper 79-1695.

[Mali81]   Malin, S.R.C. and Barraclough, D.R.: "An Algorithm for Synthesizing the Geomagnetic Field." *Computers and Geosciences*, Vol. 7 , No. 4, pp. 401-405, (1981).

[NASA69-1]   NASA: *Magnetic Fields, Earth and Extraterrestrial*. NASA SP-8017, March 1969.

[NASA69-2]   NASA: *Spacecraft Magnetic Torques*. NASA SP-8018, March 1969.

[NASA69-3] NASA: *Models of the Earth's Atmosphere, 120 to 1000 km.* NASA SP-8021, May 1969.

[Pers91] Personal communication with AeroAstro Inc. and MIT CSR staff.

[Rick87] Ricker, George R. et. al., *Proposal to NASA for a High Energy Transient Experiment.* Submitted by the Center for Space Research, MIT, Cambridge, MA.

[Stic74] Stickler, A. C. and Alfriend, K. T., "An Elementary Magnetic Attitude Control System." AIAA paper 74-923.

[Tasc88] Tascione, Thomas F: *Introduction to the space environment.* Orbit Book Co., Malabar, FL, 1988.

[Wert78] Wertz, James R. (editor): *Spacecraft Attitude Determination and Control.* D. Reidel Publishing Co., Holland/Boston, 1978.

[Wert91] Wertz, James R. and Larson, Wiley J. (editors): *Space Mission Analysis and Design.* Kluwer Academic Publishers, The Netherlands/Boston, 1991.

Appendix A MATLAB Code Referenced in Text

A-3a Coordinate Frame Transformations

```
function [ces, cse] = suneq(tropic)

% [ces cse] = suneq(tropic)
% tropic: inclination of equator WRT ecliptic **IN DEGREES**
% last updated 3/10/91 DHC-
% note screwup in vernal equinox direction corrected

tropic = tropic*(pi/180);
cse = [1     0              0;
       0     cos(tropic)  sin(tropic);
       0    -sin(tropic)  cos(tropic)];
ces = cse';
```

```
function [cge, ceg] = greenequ(greenwich)

% [cge, ceg] = greenequ(greenwich)
%
% specifies difference between earth greenwich frame and earth
% equatoral frame. angle greenwich ** in degrees ** goes from vernal
% equinox (axis e1) to axis g1, which points to intersection between
% greenwich meridian and equator.
% 9/10/90 DHC

greenwich = greenwich*(pi/180);
cg = cos(greenwich);
sg = sin(greenwich);
cge = [ cg sg 0;
       -sg cg 0;
         0  0  1];
ceg = cge';
```

```
function [coe, ceo] = eqorb(nodeang, incline)
% [coe ceo] = eqorb(nodeang, incline)
% nodeang: angle from vernal equinox to orbit ascending node
% incline: inclination of orbit about line of nodes, 0 -> orbit in e1-e2
plane
% ** BOTH IN DEGREES!! **
% last updated 9/7/90 DHC

d2r = pi/180;
nodeang = nodeang*d2r;
incline = incline*d2r;
cn = cos(nodeang); sn = sin(nodeang);
ci = cos(incline); si = sin(incline);
```

```
temp1 = [ cn    sn    0;
          -sn    cn    0;
           0     0    1];    % rotate by nodeang about e3
temp2 = [1  0  0;
         0  ci si;
         0 -si ci];          % rotate by incline about temp1_1
coe = temp2*temp1;
ceo = coe';
```

```
function [clo, col] = orbloc(eta)
% [clo col] = orbloc(eta)
% eta = orbit true anamoly, 0 -> local3 = -orbit1
% ** eta IN DEGREES **
% last updated 9/7/90 DHC


eta = eta*(pi/180);
se = sin(eta);
ce = cos(eta);

clo = [-se    ce    0;
        0     0    -1;
       -ce   -se    0];
col = clo';
```

```
function [cbl, clb, cbs, csb] = bodyloc(sun, roll, cse, ceo, col)

% [cbl clb cbs csb] = bodyloc(sun, roll, cse, ceo, col)
%
% HETE's nominal trajectory is assumed to be sun-pointing, with an
% arbitrary roll angle about the sun vector
%
% sun: angle in ecliptic plane which points to sun, 0 -> 1st day of fall
% roll: body roll angle wrt ecliptic plane, CW looking towards body3
axis
% ** BOTH IN DEGREES **
% cse, ceo, col: other rotation matrices
% last updated 9/7/90 DHC

d2r = pi/180;
sun = sun*d2r; roll = roll*d2r;
cs = cos(sun); ss = sin(sun); cr = cos(roll); sr = sin(roll);
temp1 = [cs   ss   0;
         -ss   cs   0;
          0    0   1];
temp2 = [0    cr    sr;
         0   -sr    cr;
         1    0     0];
cbs = temp2*temp1; csb = cbs';
cbl = cbs*cse*ceo*col; clb = cbl';
```

A-3b Ground Track Calculation

```
function [longi, colat] = lvlhcoord(ceo, cge, col)

% [longi colat] = LVLHCOORD(ceo, cge, col)
% calculates the east longitude and colatitude for a given orbit point
% *** -180 <= longi <= 180, 0 <= colat <= 180 ***
% (note IGRF routine is happy with these coordinate domain limits)
%
% orbit elements and local anomoly information exist in Col and Ceo
% time of day (ie relation between prime meridian and vernal equinox)
% exists in Cge matrix.
%
% DHC 9/27/90

r2d = 180/pi;
cgl = cge*ceo*col;
cel = ceo*col;

ze = cel*[0 0 -1]';         % local zenith in frame e
zep = [ze(1) ze(2) 0]';     % project to equatorial plane
zep = zep ./ norm(zep);     % normalize Zep                           .
zgp = cge*zep;              % zenith equator projection in greenwich
frame

c = cross([1 0 0]', zgp);
longi = acos([1 0 0]*zgp)*r2d*sign(c(3));
colat = acos([0 0 1]*cgl*[0 0 -1]')*r2d;

---------------------------------------------------------------------

% code to generate groundtrack in Figure 3.6

node = 120;                 % right ascension, assumed to not change
incline = 28;               % inclination
period = 95;                % orbital period in minutes
dgdeta = (period/(24*60));  % earth spin per degree true anamoly,
[deg/deg]
g0 = -70;                   % initial prime meridian-vernal equinox
separation

clear longi,clear colat

for jj = 1:length(incline)
  green = g0;
  kk = 0;
  for eta = [0:5:360*3],
    kk = kk + 1;
    green = green + 5*dgdeta;
    [cge ceg] = greenequ(green);
    [coe ceo] = eqorb(node,incline(jj));
    [clo col] = orbloc(eta);
    [longi(jj,kk) colat(jj,kk)] = lvlhcoord(ceo,cge,col);
    if longi(jj,kk) < 0, longi(jj,kk) = 360+longi(jj,kk); end
  end
end
```

A-3c Orbital Eclipse Fraction Calculation

```
% [Wer91] eq(5-3) says
%     phi = 2*acos(cos(rho)/cos(beta))
% where: phi = orbit angular travel spent in eclipse
%        rho = angular radius of earth at altitude
%        beta = angle of sun vector from orbital plane
% this is simulated for a year below

d2r = pi/180; r2d = 180/pi;
nm2km = 1.852; km2nm = 1/nm2km;
dayconv = (360/365)*d2r;

tropic = 23.439;
[ces cse] = suneq(tropic);

node = 360*rand;
regress = -6.7;                       % node regression in deg/day
incline = 28;
alt = 300;                            % orbit altitude, nautical miles
re = 6371;                            % earth mean radius in km
rho = asin(re/(re+nm2km*alt));        % earth angular radius
crho = cos(rho);

for day = 1:465,
    theta = dayconv*day;
    sunvec = [cos(theta) sin(theta) 0]';
    node = node + regress;
    [coe ceo] = eqorb(node,incline);
    so = coe*ces*sunvec;              % sun vector in orbit frame
    proj = [so(1) so(2) 0];
    so = so ./ norm(so);
    proj = proj ./ norm(proj);       % now proj*so = cos(beta)
    phi(day) = 2*acos(crho/(proj*so));
    frac(day) = phi(day)/(2*pi);
end

frac = 1-frac;                        % get fraction illuminated
phi = r2d*phi;
```

A-3d Function 'vecfftmag'

```
function [out] = vecfftmag(invec, nff)

x = fft(invec,nff);
for j = 1:length(x),
  out(j,1) = norm(x(j,:));
end
out = out*(2/nff); % scale for plotting half window
out(1) = out(1)/2; % DC element correction
```

A-3e Disturbance Torque Model Spectra

```
% Exercises disturbance torque models in orbit. Express in body
% frame. Take FFTs.
%
% Everything is indexed to match ETAVEC, an evenly spaced vector of true
% anomolies used for simulation. The parameters defining the orbit will
% relate ETAVEC to inertial space, local time etc..
%
% last updated: 3/22/91 DHC


clear
load igrfcoef85              % load current IGRF coeffs
d2r = pi/180; r2d = 180/pi;
nm2km = 1.852; km2nm = 1/nm2km;
randomize;


% orbit characteristic constants
etavec = [0:15:30721];       % 85.3 orbits, Nyquist = 12x orbit rate
[faxis nff] = freqaxis(etavec);
[ces cse] = suneq(23.439);
incline = 28;
sun = 360*rand;              % time of year info
greenwich = 360*rand;        % time of day start
nodeang = 360*rand;          % orbit ascension
roll = 360*rand;             % roll about sun vector
alt = 300;                   % altitude in nautical miles
cln = [0 1 0; -1 0 0; 0 0 1]; % local north/east/down to LVLH transform
re = 6371;                   % earth mean radius in km
altkm = alt*nm2km;           % altitude in km
mu = 3.986e14;               % earth grav const (in Nm^2/kg)
rc = re+altkm;               % orbit radius from earth center in km
gravk = 3*(mu/((1000*rc)^3)); % for grav gradient calcs
torbit = (2*pi)/(sqrt(mu/((1000*rc)^3)));   % circular orb. period (sec)
dgdeta = torbit/(24*60*60);             % deg earth spin per deg true
analomy
nodeprec = -2.06474e14*(rc^(-7/2))*cos(incline*d2r);  % (Wertz p68)
dadeta = nodeprec*dgdeta*(1/360);
deta = etavec(2) - etavec(1);               % spacing of etavec
da = dadeta*deta; dg = dgdeta*deta;
soltk = [5.5e-8 5.5e-8 5.8e-8]';        % solar torque magnitudes
vel = 7.6e3;                            % orbital vel. m/s
rho = 4e-13;                            % day atmo. density kg/m^3
[arms apad] = aerodim(1);               % aero tork model consts


% s/c mass properties
[ib ip cpb] = inert('dry', 'depl');


% choose solar torque area imbalance and propellor tilt direction
solsgn = diag(sign(rand(3,1)-0.5));
suntork = solsgn*soltk;


% choose residual s/c mag moment]
mag = (2/10);                % units of A*m^2
direc = [rand rand rand]';   % s/c dipole direction in body frame
```

```
direc = direc ./ norm(direc);
moment = mag*direc;

%** now plow thru orbits as defined by etavec

 for index = 1:length(etavec),

    % calculate all rotation matrices for this point in orbit
      eta = etavec(index);
      nodeang = nodeang + da; greenwich = greenwich + dg;
      [cge ceg] = greenequ(greenwich);
      [coe ceo] = eqorb(nodeang, incline);
      [clo col] = orbloc(eta);
      [cbl clb cbs csb] = bodyloc(sun, roll, cse, ceo, col);
      cls = clb*cbs;

    % magnetic contribution
      [longi colat] = lvlhcoord(ceo, cge, col);
      [bx by bz] = igrf(4,colat,longi,alt,gigrf,higrf);
      bb = cbl*cln*[bx by bz]';        % local field in body frame
      diptork(index,:) = cross(moment, bb)';

    % gravity gradient contribution
      cpl = cpb*cbl;
      c3 = cpl(:,3);
      ggtork(index,:) = (gravk*cross(c3,ip*c3))';

    % aero torque contribution
      fv = cbl*[-1 0 0]';              % flow vector = -LH in body frame
      airtork(index,:) = aerotork(rho,vel,fv,apad,arms)';

    % add 'em up
      sumtork(index,:) = airtork(index,:)+diptork(index,:)+...
                         ggtork(index,:)+suntork';

 end       % orbit loop terminator

% take FFT's

dipspmat = vecfftmag(diptork,nff);
airspmat = vecfftmag(airtork,nff);
ggspmat = vecfftmag(ggtork,nff);
specmat = vecfftmag(sumtork,nff);
elapse = etime(clock, tstart)
```

A-3f 2-D Aero Torque Models Comparison

```
% Compares effect of aero drag model involving shadowing
% and not for the specific case of the orbit plane being in
% the body-fixed x-z plane

% ** vehicle dimensions:
inch2meter = 2.54e-2;
d2r = (pi/180);
cm = [0 0 -18.4]'*inch2meter;        % origin to c.m. in body frame
pdl = 27.1*inch2meter;               % paddle length
pdw = 10.3*inch2meter;               % paddle projected width (ie flattened
arc)
scl = 35*inch2meter;                 % overall s/c length
scw = 18*inch2meter;                 % s/c cylinder width (where paddles
attach)
hinge = -8*inch2meter;               % hinge coord along z axis
afull = pdl*pdw;                     % full paddle area
slfr = scl-abs(hinge);               % shadowing length for front
slbk = scl-slfr;                     % for back
cpf = (scw+pdl)/2;                   % unshaded panel c.p. along x axis    .

% orbit and simulation parameters
vel = 7.6e3;                         % orbital vel. m/s
rho = 2e-13;                         % day atmo. density kg/m^3
fconst = 1.2*rho*vel*vel;            % note implicit Cd = 2.4
etavec = [0:15:30721];               % Nyquist rate = 12x orbit rate

% ** plow thru orbits
for jj = 1:length(etavec),
    phi = etavec(jj)*d2r;
    cosphi = cos(phi); sinphi = sin(phi);
    side = sign(-sinphi);            % defines the unshaded side
    flow = [-sinphi 0 cosphi]';      % flow vector

    if cosphi >= 0,
       bodl = slfr;                  % science side gives shielding
       pnv = [0 0 1]';               % relevant normal vector is +Z
    else
       bodl = slbk;                  % thruster side gives shielding
       pnv = [0 0 -1]';              % relevant normal vector is -Z
    end

    shl = abs(bodl*(sinphi/cosphi)); % positive shadow length
    cps = (shl/2)+cpf;               % shaded panel c.p. along axis
    ashade = (pdl-shl)*pdw;          % shaded paddle area
    if (ashade < 0), ashade = 0; end % total shading check

    fdir = (pnv'*flow)*flow;         % direction force application
    ff = fconst*afull*fdir;          % force on full panel
    fs = fconst*ashade*fdir;         % force on shaded panel

    rs = [(side*cps) 0 hinge]' - cm; % c.m. to shaded panel c.p.
    rf = [(-side*cpf) 0 hinge]' - cm; % c.m. to unshaded panel c.p.
    rf2 = ([(side*cpf) 0 hinge]')- cm; % case of shaded panel not shaded
```

```
    t1 = cross(rf,ff) + cross(rs,fs);   % torque for shading model
    t2 = cross(rf,ff) + cross(rf2,ff);  % torque for no shading model
    tkv(jj,:) = [t1(2) t2(2)];
end

% ** take FFT magnitudes:

[faxis nff] = freqaxis(etavec);
tkf = (2/nff)*abs(fft(tkv,nff));
tkf(1,:) = tkf(1,:) ./ 2;    % DC element correction
```

A-3g Aero Torque Model Used in Simulations

```
function [moments, apad] = aerodim(ack)

% ack is a dummy variable
% ** vehicle model dimensions for use by function aerotork:
inch2meter = 2.54e-2;
cm = [0 0 -18.4]'*inch2meter;     % origin to c.m. in body frame
pdl = 27.1*inch2meter;            % paddle length
pdw = 10.3*inch2meter;            % paddle projected width
scl = 35*inch2meter;              % overall s/c length
scw = 18*inch2meter;              % s/c cylinder width (paddle attach)
hinge = -8*inch2meter;            % hinge coord along z axis
cpf = (scw+pdl)/2;                % unshaded panel c.p. along axis

% calculate moment arms
rxr = [cpf 0 hinge]' - cm;     % c.m. to right x-z plane panel c.p.
rxl = [-cpf 0 hinge]' -cm;     % c.m. to left x-z plane panel c.p.
ryr = [0 -cpf hinge]' -cm;     % c.m. to right y-z plane panel c.p.
ryl = [0 cpf hinge]' - cm;     % c.m. to left y-z plane panel c.p.

% pass these along
apad = pdl*pdw;                   % full paddle area
moments = [rxr rxl ryr ryl];
```

```
function tork = aerotork(rho,vel,flow,area,m)

% simple aerodynamic torque model
% rho = density, vel = speed
% flow = 3x1 UNIT vector of flow velocity
% area = paddle area
% m = [3x4] matrix of moment arms precalculated in aerodim
% determine correct surface normal

if flow(3) >= 0,
   pnv = [0 0 1]';                % relevant normal vector is +Z
else
   pnv = [0 0 -1]';               % relevant normal vector is -Z
end

% calculate force and torque, note implicit Cd = 2.4
f = (1.2*rho*vel*vel)*area*(pnv'*flow)*flow;
tork = cross(m(:,1),f)+cross(m(:,2),f)+...
        cross(m(:,3),f)+cross(m(:,4),f);
```

A-4a Body-Principal Frame Transformation

```
function [ib, ip, cpb] = inert(fuel, config)

% [ib ip cpb] = INERT(fuel, config)
% access file to current HETE inertia matrix information
%
% inputs:   fuel = 'wet' or 'dry'
%           config = 'stowed' or 'xfer' or 'depl'
%           any other keywords will generate error message
% outputs:  ib = 3x3 inertia matrix in body frame (stored as data)
%           ip = 3x3 diagonal principal axis frame
%           cpb = dir cosine matrix between frames- such that
%               u(principal) = cpb u(body), u = arbitrary vector
%
% 29 June 1990 DHC
%
% current data on body frame inertia matrix placed here:
% data last update: 7/9/90
% code last update: 8/28/90
% wheel is now in body y axis

if strcmp(fuel,'wet') == 1,
    if strcmp(config,'stow') == 1,
        ib = [ 6.56    -0.01    -0.01
              -0.01     6.78    -0.05
              -0.01    -0.05     2.54];

    elseif strcmp(config,'xfer') == 1,
        ib = [ 6.41    -0.01     0.01
              -0.01     7.78    -0.04
               0.01    -0.04     3.68];

    elseif strcmp(config,'depl') == 1,
        ib = [7.21     -0.01     0.01
              -0.01     7.75    -0.04
               0.01    -0.04     4.70];

    else error ('INERT: unrecognized configuration keyword')
    end

elseif strcmp(fuel,'dry') == 1,

    if strcmp(config,'stow') == 1,
        ib = [ 3.83    -0.01    -0.02
              -0.01     4.06    -0.07
              -0.02    -0.07     1.92];

    elseif strcmp(config,'xfer') == 1,
        ib = [ 3.89    -0.01     0.01
              -0.01     5.25    -0.07
               0.01    -0.07     3.06];

    elseif strcmp(config,'depl') == 1,
        ib = [ 4.86    -0.01     0.01
```

```
                    -0.01    5.41      -0.06
                     0.01   -0.06       4.08];

       else error ('INERT: unrecognized configuration keyword')
       end

  else error ('INERT: unrecognized fuel state keyword')
  end

  % condition pvec and ip to counter eig routine output ordering
  % goal: order pvec such that it is almost diag, resolving all sign
  % ambiguities as well

  [pvec ip] = eig(ib); ip = diag(ip)';
  [jj index] = max(abs(pvec));
  [index pvec ip] = colsort(index, pvec, ip);  % function colsort is
                                               % printed below
  jj = sign(diag(pvec));
  pvec = [jj(1)*pvec(:,1)  jj(2)*pvec(:,2)  jj(3)*pvec(:,3)];
  ip = diag(ip);

  % assemble direction cosines (note assumption is made that eig
  % routine returns eigenvectors of unit length- true for pro-matlab v3.5 .
  % and up, or mac-matlab v1.2 and up, unsure for pc-matlab...)

  p1 = pvec(:,1);
  p2 = pvec(:,2);
  p3 = pvec(:,3);
  cpb = [p1'*eye(3); p2'*eye(3); p3'*eye(3)];

  end




  function [io, ao, bo] = colsort(ix, a, b)

  % [io, ao, bo] = colsort(ix,a,b)
  % ix = vector (row or column) to be sorted (ascending)
  % a,b will be sorted BY COLUMN corresponding to ix
  % {therefore must make #col(a,b) = length(ix)}

  l = length(ix);
  for q = 1:l,
     ileft = ix(q:l); [small, index] = min(ileft);

     temp = ix(q); ix(q) = small; ix(index+(q-1)) = temp;

     temp = a(:,q); a(:,q) = a(:,(index+q-1)); a(:,(index+q-1)) = temp;

     temp = b(:,q); b(:,q) = b(:,(index+q-1)); b(:,(index+q-1)) = temp;
  end
  io = ix; ao = a; bo = b;
```

A-4b Principal/Body Frame Offsets

```
function [ib, ip, cpb] = pertinert(fuel,config,diagoff,phi)

% [ib ip cpb] = PERTINERT(fuel,config,diagoff,phi)
% allows parametric perturbation of mass properties
%
% inputs:  fuel = 'wet' or 'dry'
%          config = 'stowed' or 'xfer' or 'depl'
%          diag: fractional principal diagonal offset
%          phi: euler angle of Fp Fb offset, axis random on
%               spin plane
% outputs: ib = 3x3 inertia matrix in body frame
%          ip = 3x3 diagonal principal axis frame
%          cpb = dir cosine matrix between frames-
% last updated: 5 April DHC

[ib ip cpb] = inert(fuel, config);

% offset diagonal elements
ip = ip + (diagoff .* ip);

% offset body frame from principal frame
a = [rand 0 rand]'; a = a ./ norm(a);    % euler axis on spin plane
cpb = euler(a,phi);
ib = cpb'*ip*cpb;
```

```
function [out1, out2] = euler(in1,in2);

% Euler's Axis/Angle Theorem
% if # inputs = 1:
%     in1 = rotation matrix c
%     out1 = euler axis (3,1), out2 = rotation angle (degrees)
%     ** no check for singular cases of trace(c) = 3 or -1 **
% if # inputs = 2:
%     in1 = euler axis (3,1), in2 = rotation angle (degrees)
%     out1 = rotation matrix c
% 3/22/91 DHC

if (nargin == 1),          % c --> a & phi
    c = in1;
    t = trace(c);
    phi = acos(0.5*(t-1));
    a = 0.5*[(c(2,3)-c(3,2));(c(3,1)-c(1,3));(c(1,2)-c(2,1))];
    a = a ./ sin(phi);
    phi = (180/pi)*phi;
    out1 = a; out2 = phi;
elseif(nargin == 2),
    a = in1; phi = in2*(pi/180); c = cos(phi);
    out1 = (c*eye(3)) + ((1-c)*a*a') - (sin(phi)*crossmat(a));
    out2 = 0;
end
```

A-5a Night Controllor Design Codes

Actuator Mode 1 (Full Authority):

```
% synthesize DT lqg controller for HETE mission night mode 1
% state vector definition:
%   1-3: omega1 ~ omega3
%   4-6: epsilon1 ~ epsilon3
% note eta is assumed 1 for small angle analysis
% note assume all in principal frame
% last updated: 2/27/91


plantdef;
[phi gam] = c2d(a,b,4);


cnite = [zeros(3,3) eye(3)];
[n m] = size(b); [p n] = size(cnite); dnite = zeros(p,m);


% ** estimator & control design


nprocwt = eye(3);
nmeaswt = 1e-1*eye(3);
nstatewt = diag([1 1 1 1e-2 1e-2 1e-2]);
nctrlwt = diag([1 1 1]);
nhd = dlqe(phi,gam,cnite,nprocwt,nmeaswt);
ngd = dlqr(phi,gam,nstatewt,nctrlwt);


%** define some matrices
phiol = [phi            gam*ngd*(nhd*cnite-eye(nhd*cnite));
        zeros(6,6)    (phi-gam*ngd)*(eye(nhd*cnite)-nhd*cnite)];
gamol = [gam*ngd*nhd; -(phi-gam*ngd)*nhd];
phicl = [phi-gam*ngd*nhd*cnite      gam*ngd*(nhd*cnite-eye(nhd*cnite));
        (phi-gam*ngd)*nhd*cnite    (phi-gam*ngd)*(eye(nhd*cnite)-
nhd*cnite)];
ccl1 = [cnite zeros(3,6)];    % extract principal frame eta's
ccl2 = [eye(3) zeros(3,9)];   % extract principal frame omega's
dd = zeros(3,3);
dlcl = [gam; zeros(6,3)];


pfc = log(eig(phi-phi*nhd*cnite))./4;
prc = log(eig(phi-gam*ngd))./4;
```

Actuator Mode 2 (Partial Authority):

```
% fits polynomial to gains for NIGHT mode 2 scheduled controller
% only regulator gains are computed- filter gains are the same as
% for mode 1
% last updated 4/5/91 DHC


% set these parameters:
nswt = diag([1 1 1 .01 .01 .01]);        % LQR state weight
ncwt = diag([1 1]);                       % LQR ctrl weight
```

```
numpt = 100;                                    % # of interpolating points
used
order = 8;
tvec = [0:1/numpt:1];
plantdef;
[phi gam] = c2d(a,b,4);

% solve a bunch of Ricatti equations to generate data to fit to
clear ngdat

for jj = 1:length(tvec),
   theta = tvec(jj)*(2*pi);
   bdir = [cos(theta)   0;
           0            1;
           -sin(theta)  0];
   g = dlqr2(phi,gam*bdir,nswt,ncwt);
   ngdat(jj,:) = [g(1,:) g(2,:)];
end

% now generate the polynomial approximation
clear ngtcf
indv = [1 3 4 6];
for jj = 1:4,
   ind = indv(jj);
   ngtcf(jj,:) = polyfit(tvec,ngdat(:,ind)',order);
end
ngwm = [0 mean(ngdat(:,8)) 0 0 mean(ngdat(:,11)) 0];

% save the results
ngfn = 'nsch.mat';
strn = ['save ' ngfn ' ngfn nswt ncwt ngdat tvec ngtcf '];
strn = [strn 'ngwm order indv phi gam'];
```

```
function [gain] = getngain(theta,ngtcf,ngwm)

% [gain] = getngain(theta,ngtcf,ngwm)
% retrieve gain matrix for night mode 2, used in simulations
% theta in ** RADIANS **, 0 <= theta <= 2*pi
% last updated: 2/27/90 DHC

p = theta/(2*pi);
pv = [p^8 p^7 p^6 p^5 p^4 p^3 p^2 p 1]';
gt = ngtcf*pv;
gain = [gt(1) 0 gt(2) gt(3) 0 gt(4); ngwm];
```

A-5b Night Controller Stability Evaluation

Full Authority Actuator Spin Plane Controller- Figures 5.7 thru 5.9

```
% design plant model

[ib ip cpb] = inert('dry', 'depl');
iw = 6.3e-3;
omo = (3000*2*pi)/60;
hn = iw*omo;
il = ip(1,1); i3 = ip(3,3);

a = [0 hn/il 0 0; -hn/i3 0 0 0;
     1/2 0 0 0; 0 1/2 0 0];
b = [1/il 0; 0 1/i3; 0 0; 0 0];
c = [zeros(2,2) eye(2)];
[phi gam] = c2d(a,b,4);

h = dlqe(phi,gam,c,1.0001*eye(2),1e-1*eye(2));
g = dlqr(phi,gam,diag([1 1 1e-2 1e-2]),eye(2));

% nominal plant = DPM, design closed loop poles are:
phicl = [phi-gam*g*h*c      -gam*g*(eye(4)-h*c);
         (phi-gam*g)*h*c      (phi-gam*g)*(eye(4)-h*c)];
pnom = eig(phicl);
wn = hn/sqrt(il*i3);

% perturbed plant iteration
clear pv
clear wv
ipv = [1.2:-0.0025:0.8];
hpv = [0.8:0.0025:1.2];
%hpv = ones(ipv);

for jj = 1:length(ipv),
   ipert = ipv(jj); hpert = hpv(jj);
   ilp = il*ipert; i3p = i3*ipert; hp = hn*hpert;
   ap = [0 hp/ilp 0 0; -hp/i3p 0 0 0;
         1/2 0 0 0; 0 1/2 0 0];
   bp = [1/ilp 0; 0 1/i3p; 0 0; 0 0];
   [phip gamp] = c2d(ap,bp,4);
   phicl = [phip-gamp*g*h*c      -gamp*g*(eye(4)-h*c);
            (phi-gam*g)*h*c      (phi-gam*g)*(eye(4)-h*c)];
   pv(jj,:) = eig(phicl)';
   wv(jj) = hp/sqrt(ilp*i3p);
end

hold off, clg
axis('square'); axis([-1.2 1.2 -1.2 1.2])
plot(pv,'.'); unitcircle(1); plot(pnom,'*');
grid
```

Partial Authority Actuator Spin Plane Controller- Figures 5.10 thru 5.12

```
r2d = 180/pi; d2r = pi/180; tp = 2*pi;

% design plant model
[ib ip cpb] = inert('dry', 'depl');
iw = 6.3e-3;  omo = (3000*2*pi)/60;  hn = iw*omo;
i1 = ip(1,1);  i3 = ip(3,3);

a = [0 hn/i1 0 0; -hn/i3 0 0 0;
     1/2 0 0 0; 0 1/2 0 0];
b = [1/i1 0; 0 1/i3; 0 0; 0 0];
c = [zeros(2,2) eye(2)];
[phi gam0] = c2d(a,b,4);
h = dlqe(phi,gam0,c,1.0001*eye(2),1e-1*eye(2));

% perturbed plant:

ipert = 0.9;
hpert = 1.10;
ilp = i1*ipert;  i3p = i3*ipert;  hp = hn*hpert;

ap = [0 hp/ilp 0 0; -hp/i3p 0 0 0;
      1/2 0 0 0; 0 1/2 0 0];
bp = [1/ilp 0; 0 1/i3p; 0 0; 0 0];
[phip gamp0] = c2d(ap,bp,4);

nutrat = (hp/sqrt(ilp*i3p))/(hn/sqrt(i1*i3))

thv = [0:5:355]'; delth = -40;

for jj = 1:length(thv),
    th = thv(jj)*d2r;
    thp = (thv(jj)+delth)*d2r;
    if thp > tp,
        thp = thp - tp;
    elseif thp < 0,
        thp = thp + tp;
    end
    gbig = getngain(thp,ngtcf,ngwm);
    g = [gbig(1,1) gbig(1,3) gbig(1,4) gbig(1,6)];
    bdirn = [cos(th) -sin(th)]';
    bdirp = [cos(thp) -sin(thp)]';
    gamp = gamp0*bdirn;
    gam = gam0*bdirp;
    phicl = [phip-gamp*g*h*c     -gamp*g*(eye(4)-h*c);
                  (phi-gam*g)*h*c     (phi-gam*g)*(eye(4)-h*c)];
    pv(jj,:) = eig(phicl)';
    pv2(jj,:) = log(pv(jj,:))./4;
end

hold off, clg
axis('square'); axis([-1.2 1.2 -1.2 1.2])
plot(pv,'x'),unitcircle(0), grid
```

Spin Axis Controller- Figures 5.5 and 5.6

```
% examines stability robustness of decoupled night spin-axis loop
% with discrete time root loci and bode plots

% define DPM
a = [0 0; 0.5 0];
b = [1/i2; 0];
c = [0 1];
[phi gam] = c2d(a,b,4);
h = dlqe(phi,gam,c,1,0.1);
g = dlqr(phi,gam,diag([1 1e-2]),1);

% perturbed plant root locus
clear pv;
nvec = [0.6:0.02:3.2];
for jj = 1:length(nvec),
    bp = [nvec(jj)/i2; 0];
    [phip gamp] = c2d(a,bp,4);
    phicl = [phip-gamp*g*h*c          -gamp*g*(eye(2)-h*c);
             (phi-gam*g)*h*c          (phi-gam*g)*(eye(2)-h*c)];
    pv(jj,:) = eig(phicl)';
end
phiclnom = [phi-gam*g*h*c          -gam*g*(eye(2)-h*c);
            (phi-gam*g)*h*c        (phi-gam*g)*(eye(2)-h*c)];
pnom = eig(phiclnom);

% open loop with unperturbed plant, DT bode plot
phiol = [phi          -gam*g*(eye(2)-h*c);
         zeros(2,2)   (phi-gam*g)*(eye(2)-h*c)];
gamol = [gam*g*h; -(phi-gam*g)*h];
col = [0 1 0 0];
w = logspace(-4,log10(pi/4.5),100);
[mag db phase] = dbode1(phiol,gamol,col,0,1,w,4);
```

A-5c Disturbance Rejection Analysis

```
% various SVD plots for the night mode 1 controller

deltat = 4;
[phi gam] = c2d(a,b,deltat);

%** relevant augmented system matrices:

phiol = [phi          gam*ngd*(nhd*cnite-eye(nhd*cnite));
         zeros(6,6)   (phi-gam*ngd)*(eye(nhd*cnite)-nhd*cnite)];
gamol = [gam*ngd*nhd; -(phi-gam*ngd)*nhd];
phicl = [phi-gam*ngd*nhd*cnite    gam*ngd*(nhd*cnite-eye(nhd*cnite));
         (phi-gam*ngd)*nhd*cnite  (phi-gam*ngd)*(eye(nhd*cnite)-
nhd*cnite)];

ccll = [cnite zeros(3,6)];    % extract epsilons's
ccl2 = [eye(3) zeros(3,9)];   % extract omega's
dcl = zeros(3,3);
w = logspace(-4,0,100)';

%** loop and closed loop TFM:

teps = dsig(phiol,gamol,ccll,dcl,1,w,deltat);
ceps = dsig(phicl,gamol,ccll,dcl,1,w,deltat);

%** disturbance rejection- TFM between input torques outputs

dlcl = [gam; zeros(6,3)];
seps = dsig(phicl,dlcl,ccll,dcl,1,w,deltat);
somega = dsig(phicl,dlcl,ccl2,dcl,1,w,deltat);
```

```
% various SVD plots for the night mode 2 controller at one value
% of theta

deltat = 4;
th = 330*(pi/180);
ngs = getngain(th,ngtcf,ngwm);
gams = gam*[cos(th)   0; 0 1; -sin(th)   0];

%** relevant augmented system matrices:

phiol = [phi          gams*ngs*(nhd*cnite-eye(nhd*cnite));
         zeros(6,6)   (phi-gams*ngs)*(eye(nhd*cnite)-nhd*cnite)];
gamol = [gams*ngs*nhd; -(phi-gams*ngs)*nhd];
phicl = [phi-gams*ngs*nhd*cnite    gams*ngs*(nhd*cnite-eye(nhd*cnite));
         (phi-gams*ngs)*nhd*cnite  (phi-gams*ngs)*(eye(nhd*cnite)-
nhd*cnite)];

ccll = [cnite zeros(3,6)];    % extract epsilons's
ccl2 = [eye(3) zeros(3,9)];   % extract omega's
dcl = zeros(3,3);
```

```
w = logspace(-4,0,100)';

%** loop and closed loop TFM:

teps = dsig(phiol,gamol,ccll,dcl,1,w,deltat);
ceps = dsig(phicl,gamol,ccll,dcl,1,w,deltat);

%** disturbance rejection- TFM between input torques outputs

dlcl = [gam; zeros(6,3)];
seps = dsig(phicl,dlcl,ccll,dcl,1,w,deltat);
somega = dsig(phicl,dlcl,ccl2,dcl,1,w,deltat);
```

```
function [sv] = dsig(a,b,c,d,itype,w,dt)

% DSIG: generate discrete SVD magnitudes
%
% sv = dsig(a,b,c,d, itype, w, dt)
%
% Arguments:
%       [a b c d]: defines OPEN LOOP system
%       itype: 1 produces open loop transmission SVD
%              2 produces return difference transmission SVD
%              3 produces closed loop (unity feedback assumed)
%       w: frequency vector in rad/s
%       dt: sampling time in seconds
%
% Output:   [sv]  ordered singular value magnitudes ** in dB **
%
% Comparison between SIG and DSIG
%
%       SIG:  s = j*w               in kernel c*inv(s*I-a)*b+d
%       DSIG: z = exp(j*w*dt)       in kernel c*inv(z*I-a)*b+d
%
% D. H. Chang, 8/16/80
% last revised: 8/16/90

[m,ns]=size(c);
j=sqrt(-1);
i=eye(ns);
im=eye(m);
npts = length(w);

z = exp(j*w*dt);

    if itype == 1,
        for k=1:npts,
                g=c*inv(z(k,1)*i-a)*b+d;
            sv(:,k)=svd(g);
        end,
    elseif itype == 2,
        for k=1:npts,
            g=c*inv(z(k,1)*i-a)*b+d;
```

```
        s=inv(im+g);
        sv(:,k)=svd(s);
    end,
elseif itype == 3,
    for k=1:npts,
        g=c*inv(z(k,1)*i-a)*b+d;
        cs=inv(im+g)*g;
        sv(:,k)=svd(cs);
    end,
else error('itype must be 1, 2, or 3'),
end

sv = 20*log10(sv);
```

A-5d Night Actuator Quantization Response

```
% night controller, actuator quantization study
% assume appropriate design code has been run

q1 = 6.25e-7;      % quantization interval in N*m for body torque 1 & 3
q3 = q1;           % approx 8/256 A*m^2 in 20 uT field
q2 = 5e-6;         % quantization in N*m for body torque #2

cnite = [zeros(3,3) eye(3)]; % feedback: extract attitude from DPM
com = [eye(3) zeros(3,9)];   % extract omega's from augmented system
ii = eye(nhd*cnite);

% create closed loop matrices
ack = 1;               % flag 1 for mode 1, other for mode 2
if ack == 1,
   phicl = [phi-gam*ngd*nhd*cnite  gam*ngd*(nhd*cnite-ii);
            (phi-gam*ngd)*nhd*cnite  (phi-gam*ngd)*(ii-nhd*cnite)];
   dd = zeros(3,3);
   dlcl = [gam; zeros(6,3)];
   rw = diag([(q1*q1)/12 (q2*q2)/12 (q3*q3)/12]);
else
   theta = 180*(pi/180);
   ngs = getngain(theta,ngtcf,ngwm);
   gams = gam*[cos(theta)  0; 0 1; -sin(theta)  0];
   phicl = [phi-gams*ngs*nhd*cnite  gams*ngs*(nhd*cnite-ii);
            (phi-gams*ngs)*nhd*cnite (phi-gams*ngs)*(ii-nhd*cnite)];
   dlcl = [gams;zeros(6,2)];
   rw = diag([(q1*q1)/12 (q2*q2)/12]);
end

% lyapunov, DT version
rx = dlyap(phicl, dlcl*rw*dlcl');  % state autocovariance
rom = com*rx*com';                 % omega (rates) autocovariance

% RMS outputs, convert to degrees/hr
dh2rs = pi/(3600*180);
omlim = diag(sqrt(rom))./dh2rs

% Loop to create figure 5.16 (assume ack = 1 above)

q2vec = 1e-6*[1:20];
clear rms2
for jj = 1:length(q2vec),
   q2 = q2vec(jj);
   rw = diag([(q1*q1)/12 (q2*q2)/12 (q3*q3)/12]);
   rx = dlyap(phicl, dlcl*rw*dlcl');
   rom = com*rx*com';
   rms2(jj) = sqrt(rom(2,2))/dh2rs;
end
```

A-5e Day Spin-Plane Controller Design

```
% code to generate day spin-plane controller gains scheduled
% to theta, then piecewise fit them to polynomials
% DHC 4/12/91

d2r = pi/180;

% Design Plant Model
h = iw*omo;
wpf = 0.75;              % prefilter knee in rad/s
azi = 0*d2r;            % slew azimuth angle
ar = [0            h/i1        0      0;
        -h/i3           0        0      0;
         cos(azi)   -sin(azi)   0      0;
         0              0        wpf   -wpf];
b0r = [1/i1 0; 0 1/i3; 0 0; 0 0];
cr = [0 0 0 1];

% Filter design
dprocwt = 10*eye(2);
dmeaswt = 1;
dhc = lqe(ar,b0r,cr,dprocwt,dmeaswt);


% Select regulator loop weights
if azi == 0,
    dswt = diag([1 1 1e-2 eps]);
else
    dswt = diag([1 1 eps eps]);
end
dcwt = 1;

% run through theta's
tvec = [1/99:1/99:1-1/99];
clear dgdat

for jj = 1:length(tvec),
   theta = tvec(jj)*(2*pi);
   bdir = [cos(theta) -sin(theta)]';
   br = b0r*bdir;
   dgdat(jj,:) = lqr2(ar,br,dswt,dcwt);
end

if azi ~= 0,
   dgdat(:,3) = zeros(dgdat(:,3));
end
dgdat(:,4) = zeros(dgdat(:,4));

% piecewise polynomial fit
tp = 2*pi; l = length(tvec);
b1 = round((azi/tp)*l);
b2 = round(((azi+pi)/tp)*l);
if b1 > 0
    t1 = tvec(1:b1); d1 = dgdat(1:b1,:);
else
```

```
    t1 = 0; d1 = [0 0 0 0];
end
t2 = tvec(b1+1:b2); d2 = dgdat(b1+1:b2,:);
t3 = tvec(b2+1:1); d3 = dgdat(b2+1:1,:);

for jj = 1:2,
   dgcf1(jj,:) = polyfit(t1,d1(:,jj)',8);
   dgcf2(jj,:) = polyfit(t2,d2(:,jj)',8);
   dgcf3(jj,:) = polyfit(t3,d3(:,jj)',8);
end
dg31 = d1(length(t1)/2,3);
dg32 = d2(length(t2)/2,3);
dg33 = d3(length(t3)/2,3);

dgcfmat = [dgcf1; dgcf2; dgcf3];
dg3vec = [dg31; dg32; dg33];

[phitc gam0tc] = c2d(ar,b0r,1);
dcnm = ['slew',int2str(azi/d2r),'gains.mat']
strn = ['save ' dcnm ' dgcfmat dg3vec dhc tvec azi'];
strn = [strn ' dprocwt dmeaswt dswt dcwt dgdat phitc gam0tc'];


% -----------------------------------------------------------------------


function gain = getdgain(azi,th,dgcfmat,dg3v)

% gain = getdgain(azi,th,dgcfmat,dg3v)
% calculate spin-plane day controller gains as function of theta
% using piecewise polynomial approximations

azi2 = azi+pi;
pn = th/(2*pi);
pv = [pn^8 pn^7 pn^6 pn^5 pn^4 pn^3 pn^2 pn 1]';

dgcf1 = [dgcfmat(1,:); dgcfmat(2,:)];
dgcf2 = [dgcfmat(3,:); dgcfmat(4,:)];
dgcf3 = [dgcfmat(5,:); dgcfmat(6,:)];

if th < azi,
    gain = [(dgcf1*pv)' dg3v(1) 0];
elseif (th >= azi) & (th <= azi2),
    gain = [(dgcf2*pv)' dg3v(2) 0];
elseif (th > azi2),
    gain = [(dgcf3*pv)' dg3v(3) 0];
end
```

A-5f Day Controller Stability Evaluation

```
% define nominal plant
plantdef, hn = iw*omo;   wpf = 0.75;

ar = [0            hn/i1      0      0;
      -hn/i3          0          0      0;
        cos(azi)   -sin(azi)  0      0;
        0            0          wpf   -wpf];
b0r = [1/i1 0; 0 1/i3; 0 0; 0 0];
cr = [0 0 0 1]; cp = cr;
[phir gamr0] = c2d(ar,b0r,1);


tp = 2*pi;
tv = [[25:5:155] [205:5:335]]*(pi/180);
diagoffvec = [0   -0.20   -0.20 -0.20 +0.20 +0.20 +0.20];
hoffvec =    [1   +1.10   +1.10 +1.10 +0.90 +0.90 +0.90];
deltathvec = [0    0      -20   +20    0    -20   +20];

for kk = 1:length(diagoffvec)
  nstr1 = ['pv',int2str(kk)]
  nstr2 = ['pvd',int2str(kk)]
  eval(['clear ',nstr1])
  eval(['clear ',nstr2])
  doff = diagoffvec(kk);
  hoff = hoffvec(kk);
  delth = deltathvec(kk)*(pi/180);
  [ib ip cpb] = pertinert('dry', 'depl', doff, 0);
  hp = hn*hoff;
  i1p = ip(1,1);
  i3p = ip(3,3);

  % generate perturbed plant+prefilter
  ap = [0            hp/i1p     0      0;
        -hp/i3p         0          0      0;
          cos(azi)   -sin(azi)  0      0;
          0            0          wpf   -wpf];
  b0p = [1/i1p 0; 0 1/i3p; 0 0; 0 0];
  [phip gamp0] = c2d(ap,b0p,1);

  for jj = 1:length(tv),
    thp = tv(jj)+delth;
    if (thp > tp),
       thp=thp-tp;
    elseif (thp < 0),
       thp=thp+tp;
    end

    g = getdgain(azi,thp,dgcfmat,dg3vec);
    bdir = [cos(tv(jj)) -sin(tv(jj))]';
    bp = b0p*bdir;          % plant's B matrix
    gamp = gamp0*bdir;
    bdirp =[cos(thp) -sin(thp)]';
    br = b0r*bdirp;         % compensator's B matrix
    gamr = gamr0*bdirp;
    acl = [ap -bp*g; dhc*cp ar-br*g-dhc*cr];
```

```
      temp1 = eig(acl)';
      phicl = [phip-gamp*g*dhc*cp    -gamp*g*(eye(4)-dhc*cr);
              (phir-gamr*g)*dhc*cp   (phir-gamr*g)*(eye(4)-dhc*cr)];
      temp2 = eig(phicl)';
      eval([nstr1,'(jj,:) = temp1;']);
      eval([nstr2,'(jj,:) = temp2;']);
   end
end
```

A-5g Day Actuator/Sensor Quantization Response

Spin-axis loop:

```
% calculate the RMS response of omega-2 from discretization
% of both wheel and sun-sensor in the spin-axis loop

% specify compensator, without prefilter
z1 = -0.03; z2 = -0.03; p1 = -0.75;
k = (-p1/(z1*z2))*(10^(-63/20));
numc = k*[1 -(z1+z2) z1*z2];
denc = [1 -p1 0];
[ac bc cc dc] = tf2ss(numc,denc);
[phic gamc] = c2d(ac,bc,1);

% specify plant+prefilter
% state sequence: [omega2 lambda1 filter]'
p2 = -0.75;
ap = [0 0 0; 1 0 0; 0 -p2 p2];
bp = [1/i2 0 0]';
cp = [0 0 1]; dp = 0;
[phip gamp] = c2d(ap,bp,1);

% closed loop matrices
phcl = [phip-gamp*dc*cp      gamp*cc;
        -gamc*cp             phic];
gamq = [gamp*dc gamp; gamc zeros(2,1)];

% white noise intensities
q1 = (10/1024)*(pi/180);        % sun sensor discretization, rad
q2 = 5e-6;                      % wheel torque discretization, N*m
rw = diag([(q1*q1)/12 (q2*q2)/12]);

rx = dlyap(phcl, gamq*rw*gamq');
ccl = [1 0 0 0 0];              % pick out omega-2
rom = ccl*rx*ccl';
dh2rs = pi/(3600*180);
om2rms = sqrt(rom)/dh2rs        % omega-2 RMS response
```

Spin-plane loop:

```
% calculate RMS response of rates omega-1 and omega-3
% given quantization levels for the sun sensor and the torque
% generated by the spin-axis coil

dh2rs = pi/(3600*180); d2r = pi/180;
h = iw*omo;
wpf = 0.75;                     % prefilter knee in rad/s

% white noise intensities
q1 = (10/1024)*(pi/180);        % sun sensor discretization, rad
q2 = 6e-7;                      % mag torque discretization, N*m
```

```
rw = diag([(q1*q1)/12 (q2*q2)/12]);

% plant model
ap = [0          h/i1        0     0;
      -h/i3       0          0     0;
       cos(azi)  -sin(azi)   0     0;
       0          0          wpf  -wpf];

bp = [1/i1 0; 0 1/i3; 0 0; 0 0];
cp = [0 0 0 1];
[phip gamp0] = c2d(ap,bp,1);
cc1 = [eye(2) zeros(2,6)];        % pick out omega1 & 3

% cycle through thetas
tvec = [0:5:355];
clear omrms
for jj = 1:length(tvec),
   theta = tvec(jj)*d2r;
   g = getdgain(azi,theta,dgcfmat,dg3vec);
   gamp = gamp0*([cos(theta) -sin(theta)]');
   phic = (phip-gamp*g)*(eye(4)-dhc*cp);
   gamc = -(phip-gamp*g)*dhc;
   cc = -g*(eye(4)-dhc*cp);  dc = g*dhc;

   phcl = [phip-gamp*dc*cp     gamp*cc;
           -gamc*cp            phic];
   gamq = [gamp*dc gamp; gamc zeros(4,1)];

   rx = dlyap(phcl, gamq*rw*gamq');
   rom = cc1*rx*cc1';
   temp = diag(sqrt(rom))/dh2rs;
   omrms(jj,:) = temp';
end
```

A-5h Simulation Codes

code common to both day and night simulator:

```
% define some products of plant dynamics constants to make the
% integration kernel run faster by taking advantage of MATLAB's
% fast matrix element multiply
% also define some other constants needed
% last updated 4/9/91 DHC

ii1 = -(i3-i2)/i1; ii2 = -(i1-i3)/i2; ii3 = -(i2-i1)/i3;
invi1 = 1/i1; invi2 = 1/i2; invi3 = 1/i3;
oo11 = -c32*iw/i1; oo12 = +c22*iw/i1;
oo21 = -c12*iw/i2; oo22 = +c32*iw/i2;
oo31 = -c22*iw/i3; oo32 = +c12*iw/i3;
nlsm2 = [diag([ii1 ii2 ii3]) diag([invi1 invi2 invi3])];
nlsm1 = [0    oo11  oo12;
         oo22  0    oo21;
         oo31 oo32   0];
id3 = eye(3);
```

```
% Fehlberg coefficients: (note transposes on beta & gamma)
alpha = [0 1/4  3/8  12/13  1  1/2]';
beta  = [ [    1      0      0      0      0     0]/4
          [    3      9      0      0      0     0]/32
          [ 1932  -7200   7296      0      0     0]/2197
          [ 8341 -32832  29440   -845      0     0]/4104
          [-6080  41040 -28352   9295  -5643     0]/20520 ]';
beta = [zeros(6,1) beta];
gamma = [ [902880  0  3953664  3855735  -1371249  277020]/7618050
          [ -2090  0    22528    21970    -15048  -27360]/752400  ]';
```

```
% Runge-Kutta kernel to calculate plant dynamics FOR TIME POINT (tt+ts)
% by evaluating derivatives at tnow = tt + alpha(jj)*ts. Comes out with
% xx: column of plant states at time (tt+ts). Used by simulations. Note
% this is a procedure, not a function- all variables are of the same
% scope as the code which calls this procedure.
%
% 4th & 5th order Runge-Kutta-Fehlberg algorithm as detailed in:
%
%     Forsythe, Malcolm, Moler: Computer Methods for Mathematical
%          Computations, Prentice Hall, 1977.
%
% and also used in MATLAB's ODE45 routine. Variable stepping has
% been suppressed- the 5th order results are compared to the 4th
% order to keep track of tolerance violations in 'tolflag'.
% last updated: 4/9/91 DHC

for jj = 1:6
  tnow = tt+alpha(jj)*ts;            % time considered this iteration
  xnow = xx + ts*sd*beta(:,jj);      % state "      "   "      "

  % diff. eq. for principal frame rates
  omega = xnow(1:3,1);               % current principal rates
  ownow = omo + xnow(8);             % current wheel speed
  nlst = [omega;
          omega(2)*omega(3); omega(1)*omega(3); omega(1)*omega(2);
          uach+taud];
  nlsm = [(ownow*nlsm1) nlsm2];
  xdot(1:3,1) = nlsm * nlst;         % omega dot

  % diff. eq. for body frame attitude
  ombod = cpb'*omega;                % current body frame rates
  ep = xnow(4:6,1);                  % current top 3 quaterions
  ecross = [0       ep(3)  -ep(2);
            -ep(3)  0       ep(1);
            ep(2)  -ep(1)   0];
  xdot(4:6,1) = 0.5*(ecross+xnow(7)*id3)*ombod;  % epsilon dot
  xdot(7,1) = -0.5*ep'*ombod;        % eta dot

  xdot(8,1) = domdot;                % wheel kinematics
  sd(:,jj) = xdot;                   % accumulate SD matrix
end
```

```
xx = xx + ts*sd*gamma(:,1);
```

file NITESIM: main orbit night simulator:

```
% Main simulation for orbit NIGHT controller.
%
% Mapping of plant state definitions to position in state vector:
%
% pos:  name:    purpose:
% ----  -----    --------
%  1    omega1   rate in principal direction 1 (close to body x)
%  2    omega2   rate in principal direction 2 (close to body y)
%  3    omega3   rate in principal direction 3 (close to body z)
%  4    ep1      quaternion component #1  (these measure attitude
%  5    ep2      quaternion component #2   parameters of the *body*
%  6    ep3      quaternion component #3   frame)
%  7    eta      quaternion component #4
%  8    dom      deviation of wheel speed about equilibrium
%
% last updated: 4/9/91 DHC

tfinal = length(bodfield);       % integration end time
index = 1;                       % index to service arrays
mode = 1;                        % starting actuator mode

iw = 6.3e-3;             % wheel axial moment of inertia
omo = (3000*2*pi)/60;    % nominal wheel spin rate in rad/s

% perturb plant- comment out appropriate ones

% perturb nutation frequency up 23.5%, offset Fb by 10 deg
%[ib ip cpb] = pertinert('dry', 'depl', -0.15, 10); omo = omo*1.05;

% perturb nutation frequency down 17%, offset Fb by 10 deg
[ib ip cpb] = pertinert('dry', 'depl', +0.15, 10); omo = omo*0.95;

% do not perturb plant
%[ib ip cpb] = pertinert('dry', 'depl', 0, 0);

c12 = cpb(1,2); c22 = cpb(2,2); c32 = cpb(3,2);
il = ip(1,1); i2 = ip(2,2); i3 = ip(3,3);

% set plant & compensator initial conditions here:
r2d = 180/pi;
dh2rs = (pi/(3600*180));         % conversion const from deg/hr to
rad/s;
convr = (1/dh2rs)*cpb';
x0(1:3,1) = dh2rs*[30 30 -30]';  % initial angular rates
x0(4:7,1) = [0 0 0 1]';          % initial attitude quaternions
x0(8,1) = 0;                     % initial dom
x0c = [0 0 0 0 0 0]';            % compensator initial states

% define & initialize other simulation variables:
ts = 1;                 % integration time step (in sec)
tsample = 4;            % sampling period of CCD camera
```

```
tcount = 0;              % CCD camera delay model time counter
index2 = 0;              % history index
tt = 0;                  % time variable, now set to start time
xx = x0;                 % plant state vector, now set to init conds
sd = xx*zeros(1,6);      % plant state derivative matrix, now set to 0
xbarnext = x0c;          % compensator first propagated estimate
magnext = [0 0 0]';      % moment vector command sent to torquers
ddotnext = 0;            % wheel acceleration command
magmom = magnext;
domdot = ddotnext;
psi = 0;                 % initialize psi and theta
theta = 0;

clear tout;                 % time history
clear xout;                 % plant state history
clear magout;               % mag. torquer moment history [A*m^2]
clear alfout;               % wheel acceleration history [rad/s]
clear tchist;               % check this for breaches in camera model
clear modehist;             % control scheme history
clear psihist;              % psi history
clear thetahist;            % theta history

tolflag = 0;             % counts number of tolerance breaches
tol = 1e-7;              % integration fineness tolerance

% sensor/actuator imperfection modelling control variables:
realism = zeros(4,3);
realism(1,1) = 1;        % mag torquer quantization on/off
realism(1,2) = 1;        % mag torquer bias and misalignment on/off
realism(2,1) = 1;        % wheel torque quantization on/off
realism(2,2) = 1;        % wheel torque misalignment on/off
realism(3,1) = 1;        % magnetometer measurement quantization on/off
realism(3,2) = 1;        % magnetometer measurement bias & misalignment
on/off
tk_align = misalign(5);          % torquer misalignment dir-cos matrix
mom_bdir = sign(rand(3,1)-0.5);  % mag moment bias directions
wh_align = misalign(5);          % wheel misalignment matrix
wh_bdir = sign(rand-0.5);        % wheel moment of inertia offset
direction
mag_align = misalign(5);         % magnetometer misalignment matrix
mag_bdir = sign(rand(3,1)-0.5);  % magnetometer bias directions

% define some constants to speed up execution
intconsts

% initialize sensors
magsen = magnetometer(index,bodfield,realism,mag_align,mag_bdir);

% **** The main loop ****

while (tt < tfinal)

% CCD camera computation delay model: put out new control if proper
delay
% has passed since sampling time (when tcount was reset)
% (recall ulaw was calculated at sampling time in this simulation)
```

```
if tcount == 1,
   magmom = magnext;
   domdot = ddotnext;
end

% disturbance and achieved control torques
taud = dtork(index,:)';        % disturbance torque [N*m] 3x1
[uach magbad ddotbad] = niteactu(magmom,domdot,index,bodfield,iw,...
                        realism,tk_align,wh_align,wh_bdir,mom_bdir);
taud = cpb*taud;               % convert to principal frame for use below
uach = cpb*uach;               % uach is the achieved control torque

% integration kernel:
runkutkern

% Now increment tcount, bringing tcount to correspondence with
% mod4(tt+ts). If this represnts the passing of a full sampling period,
% measure the state JUST calculated and reset tcount, then cycle
% the appropriate control law

tcount = tcount + ts;
if tcount == tsample,
   tcount = 0;

   % read magnetometer
   magsen = magnetometer(index,bodfield,realism,mag_align,mag_bdir);

   % measure attitude in body frame- assume perfect sensor
   samp = xx(4:6,1);

   % set up compensator calculation, implement initial value intercept
   modelast = mode;
   [gain mode psi theta bdir] = ncstat(magsen,ngtcf,ngwm,ngd,mode);
   if mode ~= modelast,
      xx(4:6,1) = [0 0 0]';
      samp = [0 0 0]';
      xbarnext = [0 0 0 0 0 0]';
   end

   % cycle compensator
   [xbarnext magnext ddotnext] =...
      nitecomp(samp,xbarnext,nhd,cnite,gain,mode,magsen,iw,phi,gam);
end

% *** Update all histories:

tt = tt + ts;                      % now increment tt
index = index +1;
index2 = index2 + 1;
tout(index2,1) = tt;               % time history
xout(index2,:) = xx';              % plant state history
magout(index2,:) = magbad';
alfout(index2,1) = ddotbad;
tchist(index2,1) = tcount;         % check for breaches in camera model
modehist(index2,:) = mode;
psihist(index2,:) = psi;
thetahist(index2,:) = theta;
```

```
% periodic status report
if (index2/200) == (floor(index2/200)),
    [index mode tolflag]
    [(convr*xx(1:3))' theta psi]
end

% check for integration fineness tolerance breaches:
tolchk(1) = norm(ts*sd*gamma(:,2), 'inf');
tolchk(2) = tol*max(norm(xx,'inf'),1.0);
if tolchk(1) > tolchk(2), tolflag = tolflag + 1; end

end                                     % terminator of main 'while' loop

% ** wrap things up

% break up state output for convenience
yout = (cpb'*xout(:,1:3)')';            % rates in body frame
yout = (1/dh2rs)*yout;                  % convert from rad/s to deg/hr
epsout = xout(:,4:6);
etaout = xout(:,7);
domout = xout(:,8)*(60/(2*pi));         % convert from rad/s to RPM
alfmax = (20e-3/iw);
alfcomp = alfout ./ alfmax;             % compare to max capacity
alfout = alfout ./ 8e-4;                % convert to quantized units
clear xout

tolflag
length(tout)
```

NITESIM support functions:

```
function [gain] = getngain(theta,ngtcf,ngwm)

% [gain] = getngain(theta,ngtcf,ngwm)
% retrieve gain matrix for night mode 2, used in simulations
% theta in ** RADIANS **, 0 <= theta <= 2*pi
% last updated: 2/27/90 DHC


p = theta/(2*pi);
pv = [p^8 p^7 p^6 p^5 p^4 p^3 p^2 p 1]';
gt = ngtcf*pv;
gain = [gt(1) 0 gt(2) gt(3) 0 gt(4); ngwm];
```

```
function [magsen] = magnetometer(j, bf, r, ma, mbd)

% [magsen] = magnetometer(j, bf, r, ma, mbd)
% model of sampling the magnetometers- add in bias and noise here
%
% input: j = simulation index to disturbance and earth field vectors
%        bf = earth magnetic field in BODY frame in [Tesla]
%        r = realism matrix
%        ma = mag_align matrix (see misalign.m)
%        mbd = 3 x 1 mag_bdir vector [+/- 1   +/- 1   +/- 1]'
%
% output: magsen = sensed measurement vector in body frame in [Tesla]
%                  3 x 1 vector;
%
% last updated: 1/17/90 DHC

magsen = bf(j,:)';

% bias and misalignment model here: realism matrix element (3,2)

if r(3,2) == 1,
  bias = 3e-6;                 % bias in Tesla
  biasvec = bias*mbd;
  magsen = magsen + biasvec;
  magsen = ma * magsen;
end

% quantization model here: realism matrix element (3,1)

if r(3,1) == 1,
  q = (120e-6)/1024;          % quantization in Tesla
  magsen = q*round(magsen ./ q);
end
```

```
function [m, a] = nactucalc(uc,magsen,iw,mode)
```

```
%  [m, a] = actucalc(uc,magsen,iw,mode)
%  called by control loop simulation routine calculate actuator vector
%
%  inputs:
%      uc = commanded torque in BODY frame [N*m]
%      magsen = sensed (ie corrupted) magnetometer reading in [Tesla]
%               and in body frame
%      iw = wheel moment of inertia [kg*m^2]
%      mode = control law mode currently in use
%
%  outputs:
%      m = pristine magnetic moment command [A*m^2] limited for anti-reset
%      a = pristine acceleration command [rad/s]
%
%
%  last updated 2/27/90 DHC

ss = 3.5;          % A*m^2, software saturation for anti-reset

if mode == 1,

  u = [uc; 0];
  z = [-crossmat(magsen) ([0 iw 0]'); [magsen' 0]];
  cv = inv(z)*u;
  m = cv(1:3);
  a = cv(4);

  if abs(m(1)) > ss, m(1) = ss*sign(m(1)); end   % software saturation
  if abs(m(2)) > ss, m(2) = ss*sign(m(2)); end
  if abs(m(3)) > ss, m(3) = ss*sign(m(3)); end

elseif mode == 2,                               % mode 2

  bpn = norm([magsen(1) magsen(3)]);            % recall uc(1) definition
  m2 = uc(1)/bpn;
  if abs(m2) > ss, m2 = ss*sign(m2); end        % software saturation
  m = [0 m2 0]';
  a = uc(2)/iw;
end

------------------------------------------------------------------------

function [gain,mode,psi,theta,bdir] = ncstat(magsen,ngtcf,ngwm,gd,mode)

%  [gain, mode, psi, theta, bdir] = ncstat(magsen,ngtcf,ngwm,gd,mode)
%
%  returns appropriate LQR gain matrix and mode for NITESIM
%  Schmidt trigger transition points set in this routine
%  mode 1: sum tork mode, subject to switch
%  mode 2: nite spin axis coil control mode, subject to switch
%  last updated: 2/27/90 DHC


theta = atan2(magsen(1),magsen(3))*(180/pi);
if theta < 0, theta = 360+theta; end

bnorm = magsen ./ norm(magsen);
```

```
psi = (180/pi)*acos(bnorm(2));
if psi > 90, psi = 180-psi; end

if mode == 1,

    if psi > 70,                    % mode 1 to 2 trigger point here
        th = theta*(pi/180);
        gain = getngain(th,ngtcf,ngwm);
        bdir = [cos(th)   0; 0 1; -sin(th)   0];
        mode = 2;
    else,                           % stay in mode 1
        mode = 1; gain = gd;
    end

elseif mode == 2,

    if psi < 20,                    % mode 2 to 1 trigger point here
        mode = 1; gain = gd;
    else,                           % stay in mode 2
        th = theta*(pi/180);
        gain = getngain(th,ngtcf,ngwm);
        bdir = [cos(th)   0; 0 1; -sin(th)   0];
        mode = 2;
    end

else error('invalid mode')
end

----------------------------------------------------------------------


function [ua,m,a] = niteactu(m,a,j,bf,iw,  r,  mm,  wm,  wbd,  mbd)

% [ua] = niteactu(m,a,j,bf,iw,  r,  mm,  wm,  wbd)
% calculate achieved torque [ua] in N*m in BODY frame
% also returns corrupted versions of m and a
%
% inputs:
%     m = magnetic moment command sent to torquers [A*m^2] in BODY frame
%     a = acceleration command sent to wheel [rad/s]
%     j = index to disturbance and earth field vectors
%     bf = earth mag field in BODY frame [Tesla]
%     iw = wheel moment of inertia [kg*m^2]
%     r = realism matrix
%     mm = tk_align magnetic torquer misalignment matrix
%     wm = wh_align wheel misalignment matrix
%     wbd = wh_bdir wheel moment of inertia offset direction (+/- 1)
%     mbd = 3 x 1 mom_bdir vector [+/- 1    +/- 1    +/- 1]'
%
% All imperfections of actuators (wheel and mags) are modelled here
%
% last updated 2/27/91 DHC


%*** The magnetic torquer model
% offset pristine command first, then quantize, then hard saturate

% model coil factor offset in the moment generated: realism (1,2)
```

```
if r(1,2) == 1,
    frac = 0.1;              % model percentage offset in coil factor
    m = m + frac*diag(mbd)*m;
end


% quantize the magnetic moment generated: realism(1,1)
if r(1,1) == 1,
    q = 8/256;              % quantization, note +/- 4 A*m^2
    m = q*round(m ./ q);
end


% simulate hardware saturation limit (should not be exceeded due to
% software saturation)

ss = 4;   % [A*m^2]
if abs(m(1)) > ss, m(1) = ss*sign(m(1)); end
if abs(m(2)) > ss, m(2) = ss*sign(m(2)); end
if abs(m(3)) > ss, m(3) = ss*sign(m(3)); end


% calculate actual torque generated:

mag = bf(j,:)';                     % get actual field, not measured
magtk = cross(m,mag);               % cross product
if r(1,2) == 1,                     % misalign torque vector, realism (1,2)
    magtk = mm*magtk;
end


%*** The wheel model:

% quantize the wheel acceleration: realism(2,1)
if r(2,1) == 1,
    q = 8e-4;              % [rad/s^2] <--> 5.04e-6 N*m for iw = 6.3e-3 Kg*m^2
    a = q*round(a/q);
end


whtk = [0 iw*a 0]';

if r(2,2) == 1,                     % wheel misalignment & offset, realism
(2,2)
    frac = 0.05;                    % assume 5% wheel inertia offset
    whtk = whtk*(1+wbd*frac);
    whtk = wm*whtk;                 % misalignment deflection
end


ua = magtk + whtk;                  % output achieved torque in body frame

--------------------------------------------------------------------

function [xn,m,a] = nitecomp(samp,xn,h,cn,g,mode,ms,iw,phi,gam)

% [xn m a] = nitecomp(samp,xn,h,cn,g,mode,ms,iw,phi,gam)
% outputs:
%    xn = next compensator state
%    m: coil command vector in A*m^2
%    a: wheel acceleration command in rad/s^2
% inputs:
%    samp: attitude measurement vector
```

```
%    xn: "next" compensator state from previous call of this function
%    h: compensator gain matrix
%    cn: c matrix
%    g: controller gain matrix, appropriately selected
%    mode: control scheme flag
%    ms: magnetometer signal
%    iw: wheel axial inertia (kg*m^2)
%    phi, gam: DT plant dynamics model for estimator use

err = -samp;                    % feedback error signal
xbar = xn;                      % propagate previous compensator state
xhat = xbar - h*(err+(cn*xbar));  % current state estimate

ucalc = -g*xhat;

% now calculate values of 'm' and 'a' given 'ucalc' and others
[m a] = nactucalc(ucalc,ms,iw,mode);

% anti-reset: function nactucalc contains software saturation limits
% so use its outputs in estimator prediction step
ufeed = cross(m,ms) + [0 iw*a 0]';
xn = phi*xhat+gam*ufeed;
```

file DAYSIM: main orbit day simulator

```
% Main simulation for orbit DAY controller.
%
% Mapping of plant state definitions to position in state vector:
%
% pos:   name:   purpose:
% ----   -----   --------
% 1      omega1  rate in principal direction 1 (close to body x)
% 2      omega2  rate in principal direction 2 (close to body y)
% 3      omega3  rate in principal direction 3 (close to body z)
% 4      ep1     quaternion component #1  (these measure attitude
% 5      ep2     quaternion component #2   parameters of the *body*
% 6      ep3     quaternion component #3   frame)
% 7      eta     quaternion component #4
% 8      dom     deviation of wheel speed about equilibrium
%
% last updated: 4/18/91 DHC


tfinal = length(bodfield);  % integration end time
index = 1;                  % index to service arrays
tareflag = 1;               % lambda2 tare switch

iw = 6.3e-3;                % wheel axial moment of inertia
omo = (3000*2*pi)/60;       % nominal wheel spin rate in rad/s

% perturb plant- comment out appropriate ones

% perturb nutation frequency up 23.5%, offset Fb by 10 deg
%[ib ip cpb] = pertinert('dry', 'depl', -0.15, 10); omo = omo*1.05;
```

```
% perturb nutation frequency down 17%, offset Fb by 10 deg
[ib ip cpb] = pertinert('dry', 'depl', +0.15, 10); omo = omo*0.95;

% do not perturb plant
%[ib ip cpb] = pertinert('dry', 'depl', 0, 0);

c12 = cpb(1,2); c22 = cpb(2,2); c32 = cpb(3,2);
i1 = ip(1,1); i2 = ip(2,2); i3 = ip(3,3);

% set plant & compensator initial conditions here:
r2d = 180/pi;
dh2rs = (pi/(3600*180));                % conversion const from deg/hr to
rad/s;
convr = (1/dh2rs)*cpb';
x0(1:3,1) = dh2rs*[-100 120 100]';   % initial principal rates

% initial body attitude quaternions
x0(4:6,1) = (pi/180)*[-1.5 -2.5 0]';
x0(7,1) = 1;
x0(8,1) = 50*sign(rand-0.5)*((2*pi)/60);   % initial dom in rad/s

% define & initialize other simulation variables:
ts = 1;              % integration time step (in sec)
index2 = 0;          % history index
tt = 0;              % time variable, now set to start time
xx = x0;             % plant state vector, now set to init conds
sd = xx*zeros(1,6);  % plant state derivative matrix, now set to 0
domdot = 0;          % wheel acceleration command
xcaxn = [0 0]';      % initial axial compensator states
xctn = [0 0 0 0]';   % initial tranverse compensator states
idle = 0;            % assume initially not idling

% sun sensor pre-filter in DT
wpf = 0.75;     % filter knee in rad/s
[sspfphi sspfgam] = c2d(diag([-wpf -wpf]),diag([wpf wpf]),1);

% specify wheel axis compensator (assume transverse comp data loaded):
% note the pre-filter pole is included in sun sensor model
z1 = 0.03; z2 = 0.03; p1 = 0.75;
k = (p1/(z1*z2))*(10^(-68/20));
numc = k*[1 (z1+z2) z1*z2];
denc = [1 p1 0];
[acax bcax ccax dcax] = tf2ss(numc,denc);
[phiaxc gamaxc] = c2d(acax,bcax,1);

% simulation histories
clear tout;
clear xout;
clear magout
clear alfout
clear dihist
clear thetahist
clear bpnhist
clear lamout
clear tarehist
clear idlehist
```

```
tolflag = 0;          % counts number of tolerance breaches
tol = 1e-7;           % integration fineness tolerance

% sensor/actuator imperfection modelling control variables:
realism = zeros(4,2);
realism(1,1) = 1;     % mag torquer quantization on/off
realism(1,2) = 1;     % mag torquer bias and misalignment on/off
realism(2,1) = 1;     % wheel torque quantization on/off
realism(2,2) = 1;     % wheel torque misalignment on/off
realism(3,1) = 1;     % magnetometer measurement quantization on/off
realism(3,2) = 1;     % magnetometer measurement bias & misalignment
on/off
realism(4,1) = 1;     % sun sensor quantization
tk_align = misalign(5);          % torquer misalignment dir-cos matrix
mom_bdir = sign(rand(3,1)-0.5);  % mag moment bias directions
wh_align = misalign(5);          % wheel misalignment matrix
wh_bdir = sign(rand-0.5);        % wheel moment of inertia offset
direction
mag_align = misalign(5);         % magnetometer misalignment matrix
mag_bdir = sign(rand(3,1)-0.5);  % magnetometer bias directions

% define some constants to speed up execution
intconsts

% **** The main loop ****
sunsensinit
while (tt < tfinal)

% sensor models
magsen = magnetometer(index,bodfield,realism,mag_align,mag_bdir);
[samp sspfn tare lam] = sunsensor(xx,realism,sspfphi,sspfgam,...
                              sspfn,azi,tare,tdcnst,tareflag);

% wheel axis compensator loop:
xcax = xcaxn;
xcaxn = phiaxc*xcax - gamaxc*samp(1);
u2c = ccax*xcax - dcax*samp(1);

% wheel desaturation loop: comment out appropriate line to turn on/off
%tdsc = 0; mag13 = [0 0 0]'; di = 0;
[tdsc mag13 di] = desat(xx,magsen);

% check transverse controller idle flag, reset sun sensor if waking up
idlelast = idle;
[bpn,theta,idle,gain,bdir] = transcstat(magsen,azi,dgcfmat,dg3vec);
if (idle == 0) & (idlelast == 1),
   sunsens2reset,
end

% transverse axis compensator loop
if idle == 1,
   mag2 = 0; xctn = [0 0 0 0]';
else
   [mag2 xctn] = transcomp(phitc,gam0tc,samp,xctn,gain,dhc,bdir,bpn);
end

% get disturbance and achieved control torques
```

```matlab
taud = cpb*dtork(index,:)';       % disturbance torque [N*m], 3x1 column
u2 = u2c-tdsc;                     % axial compens - desat loop outputs
mag = [mag13(1) mag2 mag13(3)]';  % uncorrupted mag coil commands
[uach domdot mag] = dayactu(u2,mag,iw,bodfield,index,realism,...
                     wh_align,wh_bdir,mag_align,mag_bdir);
uach = cpb*uach;                  % convert achieved control torque to Fp

% integration kernel:
runkutkern

% *** Update all histories:

tt = tt + ts;                     % now increment tt
index = index +1;
index2 = index2 + 1;
tout(index2,1) = tt;              % time history
xout(index2,:) = xx';            % state history
magout(index2,:) = mag';         % mag. torquer moment history [A*m^2]
alfout(index2,1) = domdot;       % wheel acceleration history [rad/s]
thetahist(index2,:) = theta;
bpnhist(index2,:) = bpn;
dihist(index2,:) = di;
tarehist(index2,:) = tare';
idlehist(index2,:) = idle;

% periodic status report
if (index2/200) == (floor(index2/200)),
   [index idle tolflag]
   [(convr*xx(1:3))' theta tare']
 end

% ** check for integration fineness tolerance breaches:
tolchk(1) = norm(ts*sd*gamma(:,2), 'inf');
tolchk(2) = tol*max(norm(xx,'inf'),1.0);
if tolchk(1) > tolchk(2), tolflag = tolflag + 1; end

end                               % terminator of main 'while' loop

% ** wrap things up

% break up state output for convenience
yout = (cpb'*xout(:,1:3)')';
yout = (1/dh2rs)*yout;            % convert from rad/s to deg/hr
epsout = xout(:,4:6);            % body frame quaternions 1~3
etaout = xout(:,7);
domout = xout(:,8)*(60/(2*pi));  % convert from rad/s to RPM
clear xout

alfmax = (20e-3/iw);
alfcomp = alfout ./ alfmax;      % compare to max capacity
alfout = alfout ./ 8e-4;         % convert to quantized units
lam2 = cos(azi)*epsout(:,1)-sin(azi)*epsout(:,3);
lamout = 2*r2d*[epsout(:,2) lam2];

tolflag
length(tout)
```

DAYSIM support functions:

```
function [uach,a,mag] = dayactu(u2,mag,iw,bf,ind,r,wm,wbd,mm,mbd);

% [uach,domdot,mag] = dayactu(u2,mag,iw,bf,ind,r,wm,wbd,mm,mbd);
% day controller actuator model
%
% inputs:
%    u2 = commanded spin axis control torque vector in BODY frame
%    mag = commanded pristine torquer coil commands (A*m^2)
%    iw = wheel inertia
%    bf = "actual" local field in body frame
%    ind = current index to bf
%    r = realism matrix
%    wm = wh_align wheel misalignment matrix
%    wbd = wh_bdir wheel moment of inertia offset direction (+/- 1)
%    mm = mag_align torquer misalignment matrix
%    mbd = mag_bdir torquer offset directions
%
% outputs:
%    uach = achieved torque in BODY frame
%    a = resultant wheel acceleration in rad/s^2
%
% All imperfections of the actuators (wheel & torquers) modelled here
%
% last updated 4/10/91 DHC

% deal with wheel
a = u2/iw;

if r(2,1) == 1,        % quantize the wheel acceleration: realism(2,1)
    q = 8e-4;          % [rad/s^2] <--> 5.04e-6 N*m for iw = 6.3e-3 Kg*m^2
    a = q*round(a/q);
end

whtk = [0 iw*a 0]';
if r(2,2) == 1,                    % wheel misalignment & bias, realism (2,2)
    frac = 0.05;                   % assume 5% wheel inertia offset
    whtk = whtk*(1+wbd*frac);
    whtk = wm*whtk;                % misalignment deflection
end

%*** The magnetic torquer model
% offset pristine command first, then quantize, then hard saturate
% model coil factor offset in the moment generated: realism (1,2)

if r(1,2) == 1,
    frac = 0.1;                % model 10% percent offset in coil factor
    mag = mag + frac*diag(mbd)*mag;
end

% quantize mag moment if necessary
if r(1,1) == 1,
  q = 8/256;
```

```
    mag = q*round(mag./q);
end


% simulate hardware saturation limit (should not be exceeded due to
% presence of software saturation)
ss = 4;    % [A*m^2]
if abs(mag(1)) > ss, mag(1) = ss*sign(mag(1)); end
if abs(mag(2)) > ss, mag(2) = ss*sign(mag(2)); end
if abs(mag(3)) > ss, mag(3) = ss*sign(mag(3)); end


% magnetic torque generated torque
magtk = cross(mag,bf(ind,:)');    % cross product with "actual" field
if r(1,2) == 1,                   % misalign torque vector, realism (1,2)
    magtk = mm*magtk;
end


uach = magtk+whtk;


-----------------------------------------------------------------------


function [tdsc, mag, di] = desat(xx,magsen)

% [tds mag di] = desat(xx,magsen,bf,index,r,mm)
% desaturation loop calculation
% outputs:
%    tdsc = calculated (ie without knowledge of magnetometer and
%            torquer offsets, misalignments, quantization etc.)
%            desat torque to be used to command wheel with
%    mag = magnetic moment commands issued, quantized if necessary
%    di = desaturation index
% inputs:
%    xx = current simulation state vector
%    magsen = current magnetometer reading, corrupted, quantized etc.
% 2/10/91 DHC


omega = xx(8);
k = 6.3e-6;      % de-sat loop time constant = 1000s for Iw = 6.3e-3
tdmax = 4e-5;    % max tolerable N*m
b13n = norm([magsen(1) magsen(3)]);
di = abs(magsen(2))/b13n;


tdm = abs(k*omega);
if (tdm*di) > tdmax,
    tdm = tdmax/di;
end


tdsc = sign(-omega)*tdm;
c = tdsc/(b13n*b13n);
m1 = magsen(3)*c;
m3 = -magsen(1)*c;
mag = [m1 0 m3]';                      % pristine coil command


-----------------------------------------------------------------------
```

```
function gain = getdgain(azi,th,dgcfmat,dg3v)

% gain = getdgain(azi,th,dgcfmat,dg3v)
% calculate spin-plane day controller gains as function of theta
% using piecewise polynomial approximations

azi2 = azi+pi;
pn = th/(2*pi);
pv = [pn^8 pn^7 pn^6 pn^5 pn^4 pn^3 pn^2 pn 1]';

dgcf1 = [dgcfmat(1,:); dgcfmat(2,:)];
dgcf2 = [dgcfmat(3,:); dgcfmat(4,:)];
dgcf3 = [dgcfmat(5,:); dgcfmat(6,:)];

if th < azi,
    gain = [(dgcf1*pv)' dg3v(1) 0];
elseif (th >= azi) & (th <= azi2),
    gain = [(dgcf2*pv)' dg3v(2) 0];
elseif (th > azi2),
    gain = [(dgcf3*pv)' dg3v(3) 0];
end
```

                                                                                  .

```
function [magsen] = magnetometer(j, bf, r, ma, mbd)

% [magsen] = magnetometer(j, bf, r, ma, mbd)
% model of sampling the magnetometers- add in bias and noise here
%
% input: j = simulation index to disturbance and earth field vectors
%        bf = earth magnetic field in BODY frame in [Tesla]
%        r = realism matrix
%        ma = mag_align matrix (see misalign.m)
%        mbd = 3 x 1 mag_bdir vector [+/- 1    +/- 1    +/- 1]'
%
% output: magsen = sensed measurement vector in body frame in [Tesla]
%                  3 x 1 vector;
%
% last updated: 1/17/90 DHC

magsen = bf(j,:)';

% bias and misalignment model here: realism matrix element (3,2)

if r(3,2) == 1,
  bias = 3e-6;                % bias in Tesla
  biasvec = bias*mbd;
  magsen = magsen + biasvec;
  magsen = ma * magsen;
end

% quantization model here: realism matrix element (3,1)

if r(3,1) == 1,
  q = (120e-6)/1024;         % quantization in Tesla
  magsen = q*round(magsen ./ q);
end
```

```
--------------------------------------------------------------------

% initialize the sun sensor prefilter states and tare values

tare = [0 0]';
tdcnst = diag([0 0]);
[samp sspfn tare lam] = sunsensor(xx,realism,sspfphi,sspfgam,...
                                  sspfn,azi,tare,tdcnst,tareflag);


tare = [-lam(1) -lam(2)]';
td2steps = abs(lam(2))*r2d*(1000/0.4);
if td2steps < 500, td2steps = 500; end
if td2steps > 3000, td2steps = 3000; end
tdcnst = diag([(0.05)^(1/250) (0.05)^(1/td2steps)]);
sspfn = [0 0]';


--------------------------------------------------------------------

% Start the transverse axis compensator by reading the sun sensor,
% determining the desired elevation tare, tare decay constant, and
% in-loop prefilter initial states. This procedure is called when      .
% returning from spin-plane controller idle

tare(2) = 0;
tdcnst(2,2) = 0;

[samp sspfn tare lam] = sunsensor(xx,realism,sspfphi,sspfgam,...
                                  sspfn,azi,tare,tdcnst,tareflag);


tare(2) = -lam(2);
td2steps = abs(lam(2))*r2d*(1000/0.4);
if td2steps < 500, td2steps = 500; end
if td2steps > 3000, td2steps = 3000; end
tdcnst(2,2) = (0.05)^(1/td2steps);
sspfn(2) = 0;


--------------------------------------------------------------------

function [out,pfn,en,lam] = sunsensor(x,r,phi,gam,pfs,azi,e,d,f)

% [out,pfn,en,lam] = sunsensor(x,r,phi,gam,pfs,azi,e,d,f)
% the sun sensor model
% output:   out: 2 x 1 vector of prefilter outputs, quantized
%           pfn: next prefilter state
%           en: next tare elevation *in radians*
%           lam: provided to simulate "direct" (e.g. thru a higher
%                BW filter than the prefilter) sun-sensor model outputs
% inputs:
%     x: current daysim state vector
%     r: realism matrix
%     phi, gam: digitized prefilter system matrix
%     pfs: state of prefilters, which is usually the last 'pfn'
%     azi: azimuth in *radians*
%     e: tare elevation in *radians*, usually the last 'en', [2 x 1]
```

```
%     d: tare decay constants (assume diagonal [2 x 2])
%     f: flag- 0 = disable lambda2 tare, other = enable lambda2 tare
%
% no misalignment/bias modelled- see notes for justification.
% generalized for slewed attitude
% last updated: 4/9/91 DHC


sa = sin(azi); ca = cos(azi);
ep = x(4:6,1);

% processed sunsensor outputs
l1 = 2*ep(2) + e(1);
if f == 0,
   l2 = 2*(ca*ep(1)-sa*ep(3));
else
   l2 = 2*(ca*ep(1)-sa*ep(3)) + e(2);
end

lam = [l1; l2];

pfn = phi*pfs+gam*lam;              % drive the pre-filter
out = pfs;                          % current pre-filter output
en = d*e;                          % decay the tare values           .

% quantization: realism matrix element (4,1)
if r(4,1) == 1,
   q = 10/1024;      % quantization level, note +/- 5 deg
   out = q*round(((180/pi)*out) ./ q);
   out = (pi/180)*out;
   lam = q*round(((180/pi)*lam) ./ q);
   lam = (pi/180)*lam;
end


-------------------------------------------------------------------------


function [m2,xn] = transcomp(phi,gam0,s,xbar,g,h,bdir,bpn)

% [m2,xn] = transcomp(phi,gam0,s,xbar,g,h,bdir,bpn)
%
% the spin-plane compensator
% inputs:
%     phi, gam0: controller system matrices
%     s: sun-sensor pre-filter output , properly corrupted
%     xbar: current compensator state, usually previous 'xn'
%     g: controller scheduled LQR gains
%     h: controller estimator gains, unchanging
%     bdir: direction component of controller B (actually gam) matrix
%     bpn: magnitude of spin plane components of measured field
% outputs:
%     m2: pristine spin axis coil command A*m^2
%     xn: compensator next state
% 2/25/91 DHC

% state estimation step
e = -s(2) + xbar(4);               % estimator innovation, of correct sign
xhat = xbar - h*e;
```

```
% control step and anti-windup
tpc = -g*xhat;          % pseudo-control signal
m2 = tpc/bpn;           % spin axis coil command A*m^2 (pristine)
if abs(m2) > 3.5,       % saturation element to implement anti-windup by
  m2 = sign(m2)*3.5;    % limiting m2 and, more importantly, the tpc
which is
  tpc = bpn*m2;         % fed back to the compensator state prediction
step
end

% estimator state prediction step
xn = phi*xhat+gam0*bdir*tpc;


----------------------------------------------------------------------


function [bpn,theta,idle,gain,bdir] = transcstat(m,azi,gcfm,g3v)

% [bpn,theta,idle,gain,bdir] = transcstat(m,azi,gcfm,g3v)
% spin plane compensator status selector
% inputs:  m: output of magnetometer model
%          azi: slew azimuth angle *in radians*
%          gcfm, g3v: gain-scheduling parameters
% outputs: bpn: magnitude of spin plane components of measured field
%          theta: (in degrees)
%          idle: transverse axis control loop idle flag
%          gain: scheduled LQR gain matrix to be used
%          bdir: directional component of B matrix
% last updated 4/10/91 DHC

bpn = norm([m(1) m(3)]);
th = atan2(m(1),m(3));
theta = th*(180/pi);
if th < 0,
   th = pi+pi+th;
end

gain = getdgain(azi,th,gcfm,g3v);
bdir = [cos(th) -sin(th)]';
idle = 0;

thzone = 20*(pi/180);
if (abs(azi-th) < thzone) | (abs(azi+pi-th) < thzone),
  idle = 1;
end
if bpn < 12.8e-6,
  idle = 1;
end
```

Appendix B IGRF Function Implementation

A function to calculate the spherical harmonic expansions to arbitrary order is given below along with the 1985 IGRF coefficients used. The function has been tested against published field measurement data and found to be accurate to within the expected bounds. For algorithm description, see [Mail81].

```
function [bx,by,bz] = igrf(nmax, colat, longi, alt, g, h)

%  [bx,by,bz] = igrf(nmax, colat, longi, alt, g, h)
%  IGRF: calculates the International Geomagnetic Reference Field
%  model of arbitrary order for a given point in geocentric frame
%
%  inputs:
%  -------
%  nmax: highest order harmonic term evaluated in expansion
%  colat: geocentric colatitude in DEG, 0 <= colat <= 180 only
%  longi: geocentric east longitude in DEG, ± values ok (ie -70 = 290)   .;
%  alt: altitude above mean sea level in NAUTICAL MILES
%  g,h: IGRF coefficient matrices
%      *** format (m+1 by n) *** m+1 necessary to get zero'th entry
%      example: g for n = 7, m = 3 would be in g(4,7)
%      NaN's where the specified entry doesn't exist
%  outputs:
%  --------
%  [bx by bz]: mag field components in whatever units g & h are in
%      x is local north, y local east, z local nadir
%      this is the 'geomagnetic' system is used most often by geoscience
%      textbooks and data tables (eg. IAGA Bulletins)
%
%  There are many algorithms for grinding thru the spherical harmonics
%  to synthesize the field- this one is based on Malin & Barraclough,
%  Computers & Geosciences, vol 7, no.4, pp 401-405, 1981.
%
%  *** Note that the current implementation crashes when the colatitude
%  is 0 or 180 (geocentric poles). The small fix to check for this
%  was omitted in the interest of speed, but can be put in easily. See
%  the article mentioned above, p402, 2nd paragraph
%
%  13 Sept. 1990 D. H. Chang

%  initialize some parameters

d2r = pi/180;
colat = colat * d2r;
longi = longi * d2r;
st = sin(colat);
ct = cos(colat);
sl = sin(longi);
cl = cos(longi);
```

```
% calculate cos(m*l) and sin(m*l) recursively for speed
% format cos(m*l) = cv(m), no offset by 1 needed since no m = 0 case

cv(1) = cl; sv(1) = sl;
for m = 2:nmax,
   cv(m) = cv(m-1)*cl - sv(m-1)*sl;
   sv(m) = sv(m-1)*cl + cv(m-1)*sl;
end

re = 6.371e6;                    % earth radius in meters
ho = alt*1.852e3;               % convert altitude to meters
rat = re/(re+ho);

% initialize matrices p and x
% format: pmn = p(m+1, n+1), xmn = x(m+1, n+1)

p = nan*ones(nmax+1,nmax+1); x = p;
p(1,1) = 1; p(2,2) = st;        % seed: p00 = 1, p11 = sin(colat)
x(1,1) = 0; x(2,2) = ct;        % seed: x00 = 0, x11 = cos(colat)
p(1,2) = ct;
x(1,2) = -st;

% calculate diagonal elements of p and x:
% note these are functions of the previous diagonals only

for n = 2:nmax,
   sq = sqrt(1-1/(2*n));
   p(n+1, n+1) = p(n,n)*st*sq;
   x(n+1, n+1) =  sq*(st*x(n,n) + ct*p(n,n));
end

% calculate off-diags of case m = n-1
% note these values are functions of the diagonal elements only

for n = 2:nmax,
   m = n-1;
   den = sqrt(n*n-m*m);
   a = 2*n-1;
   p(m+1, n+1) = a*ct*p(m+1,n)/den;
   x(m+1, n+1) = a*(ct*x(m+1,n) - st*p(m+1,n))/den;
end,

% calculate off-diagonal elements of p and x, with m up to n-2:
% note these are functions of the all above calculated elements and
% therefore must be calculated last

for n = 2:nmax,
  for m = 0:n-2,
    sq = sqrt((n-1)*(n-1)-m*m);
    den = sqrt(n*n-m*m);
    a = 2*n-1;
    p(m+1, n+1) = (a*ct*p(m+1,n) - sq*p(m+1,n-1))/den;
    x(m+1, n+1) = (a*(ct*x(m+1,n) - st*p(m+1,n)) - sq*x(m+1,n-1))/den;
  end,
end,
```

```
% now assemble field components

bx = 0; by = 0; bz = 0;
for n = 1:nmax,
   r = rat^(n+2);
   bx = bx + g(1,n)*r*x(1,n+1);
   z = (n+1)*p(1,n+1);
   bz = bz - g(1,n)*r*z;
   for m = 1:n,
       bx = bx + r*x(m+1,n+1)*(g(m+1,n)*cv(m)+h(m+1,n)*sv(m));
       y = m*p(m+1,n+1)/st;
       by = by + r*y*(g(m+1,n)*sv(m)-h(m+1,n)*cv(m));
       z = (n+1)*p(m+1,n+1);
       bz = bz - r*z*(g(m+1,n)*cv(m)+h(m+1,n)*sv(m));
   end,
end
```

M \ N	N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8	N=9	N=10
M=0	-29877	-2073	1300	937	-215	52	75	21	5	-4
M=1	-1903	3045	-2208	780	356	65	-61	6	10	-4
M=2	NaN	1691	1244	363	253	50	2	0	1	2
M=3	NaN	NaN	835	-426	-94	-186	24	-11	-12	-5
M=4	NaN	NaN	NaN	169	-161	4	-6	-9	9	-2
M=5	NaN	NaN	NaN	NaN	-48	17	4	2	-3	5
M=6	NaN	NaN	NaN	NaN	NaN	-102	9	4	-1	3
M=7	NaN	NaN	NaN	NaN	NaN	NaN	0	4	7	1
M=8	NaN	NaN	NaN	NaN	NaN	NaN	NaN	-6	2	2
M=9	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	-5	3
M=10	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0

Table AB-1: IGRF coefficients "G" for 1985. "NaN' stands for "Not a Number" and is used as a placeholder where the particular value is not defined.

M \ N	N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8	N=9	N=10
M=0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
M=1	5497	-2191	-312	233	47	-16	-82	7	-21	1
M=2	NaN	-309	284	-250	148	90	-26	-21	16	0
M=3	NaN	NaN	-296	68	-155	69	-1	5	9	3
M=4	NaN	NaN	NaN	-298	-75	-50	23	-25	-5	6
M=5	NaN	NaN	NaN	NaN	95	-4	17	11	-6	-4
M=6	NaN	NaN	NaN	NaN	NaN	20	-21	12	9	0
M=7	NaN	NaN	NaN	NaN	NaN	NaN	-6	-16	10	-1
M=8	NaN	NaN	NaN	NaN	NaN	NaN	NaN	-10	-6	4
M=9	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2	0
M=10	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	6

Table AB-2: IGRF coefficients "H" for 1985. "NaN" stands for "Not a Number" and is used as a placeholder where the particular value is not defined.

Appendix C Earth Field Simulation Results

Some straightforward simulations were run for the purpose of visualizing long term field trends in orbit and exercising the code. The strategy was to generate the field in orbit in $\mathcal{F}_N$, then transform to $\mathcal{F}_B$ assuming a sun-pointing attitude at various roll angles about the sun vector. Roll angle is as defined in Section 3.1.6, in the definition of the C_{BS} transformation. The following parameters were used:

Orbit: 300 nautical mile, circular, 28° inclination

Node Regression: -6.7 deg/day

Time per orbit: 5.7371×10^3 sec.

Orbits per day: 15.06

Initial time of day, right ascension: arbitrary

Simulation length: 1978560° true anomaly (5496 orbits, approx. 1 year)

Simulation time step: 30° true anomaly

FFT Info: Window length: 65536 elements
Nyquist freq (times orbit rate): 6
Frequency resolution: (times orbit rate/bin): $12/65536 = 1.831 \times 10^{-4}$
Magnitude Scaling: 2/65536 except at DC for plotting half-window

Appendix C-1: Earth Field in $\mathcal{F}_N$

Figure AC1-1 shows simulation results for the field in $\mathcal{F}_N$. As can be seen, the results match expectations from Section 3.2.4. More information can be obtained from examining the spectra of the field's three components. FFT's are shown in Figures AC1-2, the magnitudes normalized so as to allow the use of the continuous time units $\mu T/Hz$. The frequency components magnitudes match the envelope observed in Figure AC1-1 well. As can be seen, frequency components at once and twice orbit rate, as well as a terrestrial day component at $1/15.06 = 0.0664$ times orbit rate, are prominent in the spectra. Also note the large DC bias on the North component.

Next page: Figure AC1-1: time history plots from the field simulation; little information beyond the envelope size is seen in the plots at this scale.

N-E-D frame, north component

micro-Tesla

orbit true anomaly (deg.) for 1 year   x10⁶

N-E-D frame, east component

micro-Tesla

orbit true anomaly (deg.) for 1 year   x10⁶

N-E-D frame, nadir component

micro-Tesla

orbit true anomaly (deg.) for 1 year   x10⁶
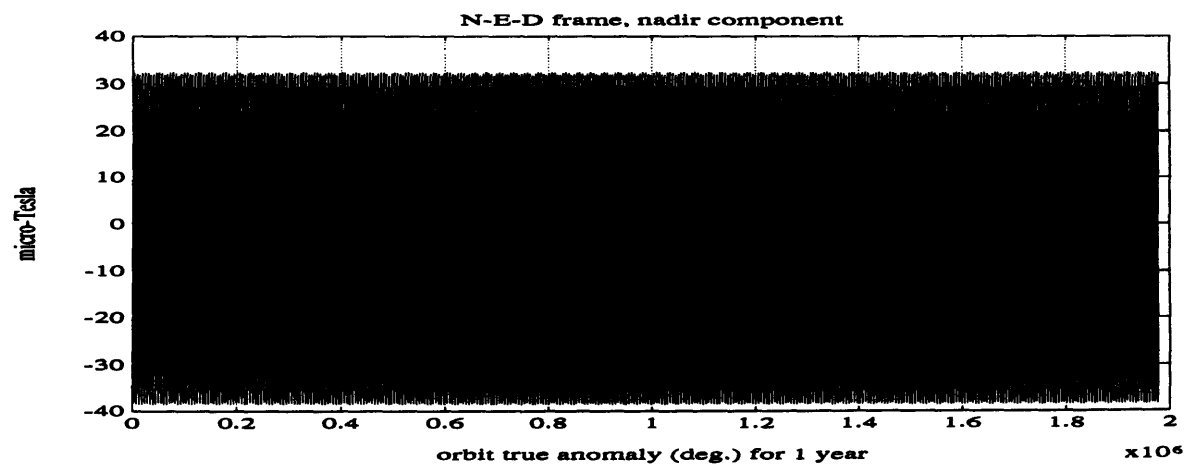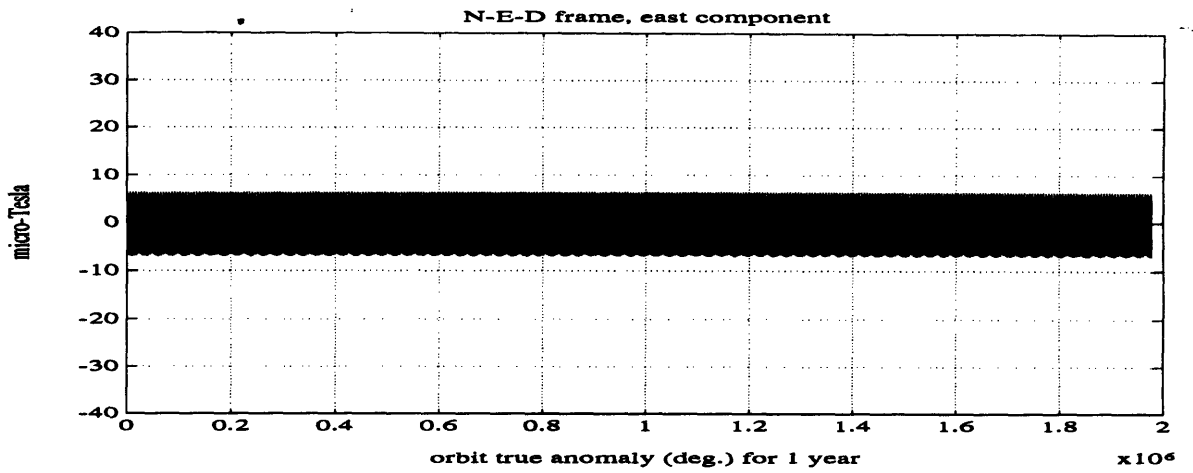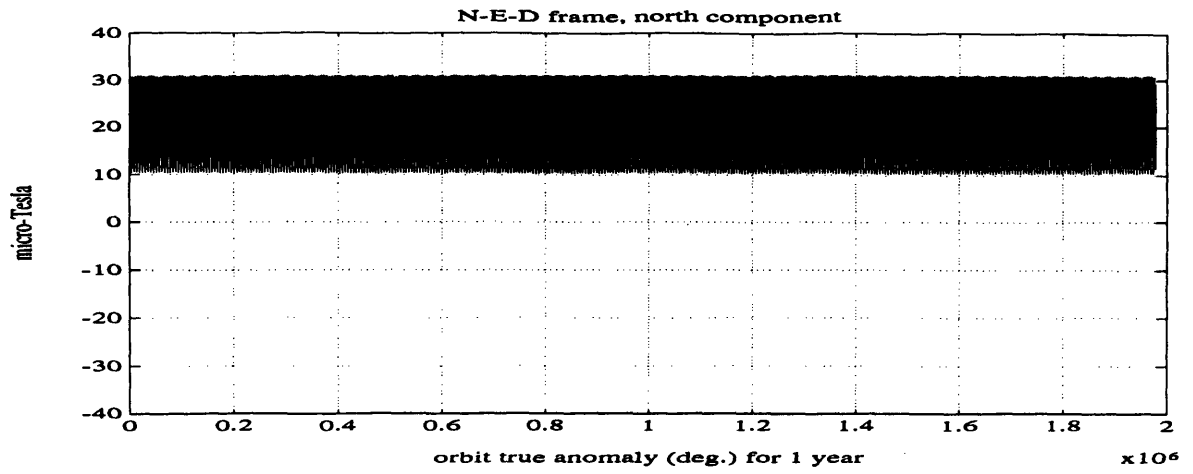
Next 2 pages:

Figures AC1-2a, b: FFT's of North component, (b) is a closeup of (a) at low frequency. Note that (b) plot clearly shows the daily component at 1/15.06 = 0.0664 times orbit rate. Note also the dominance of the DC component, whose value is printed but not shown on the plot for scaling reasons. There is a lack of components past about 6 times orbit rate, as claimed in Chapter 3.

Figures AC1-2c, d: FFT's of the East and Nadir components, showing similar decompositions. The daily component is visible in both. Note the small DC bias of these two plots compared to the North component.

Figure AC1-2a

micro-Tesla/Hz

DC value = 22.42

normalized frequency (1 = orbit rate)
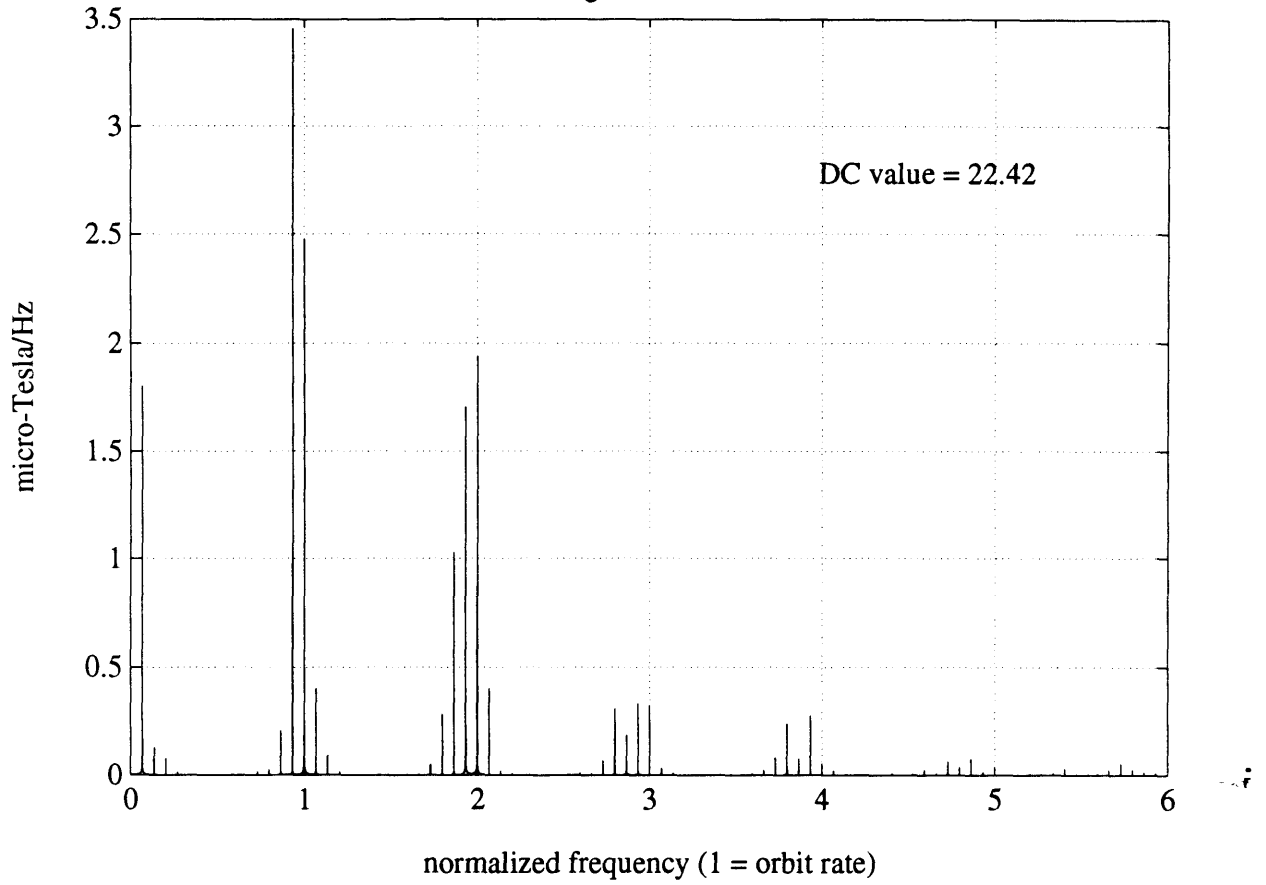
Figure AC1-2b: closeup at low frequency

micro-Tesla/Hz

DC value = 22.42

daily component

normalized frequency (1 = orbit rate)
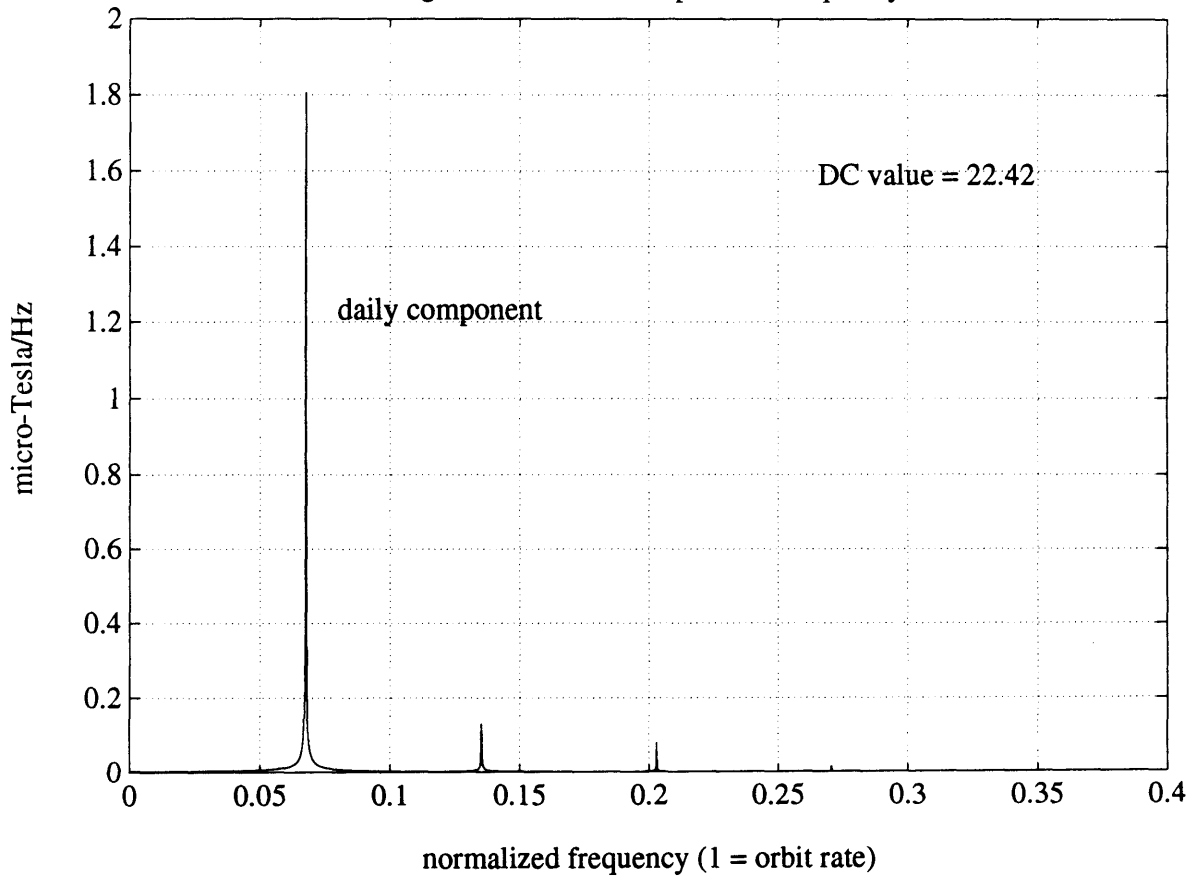
Figure AC1-2c: east component

DC value = 0.0005494
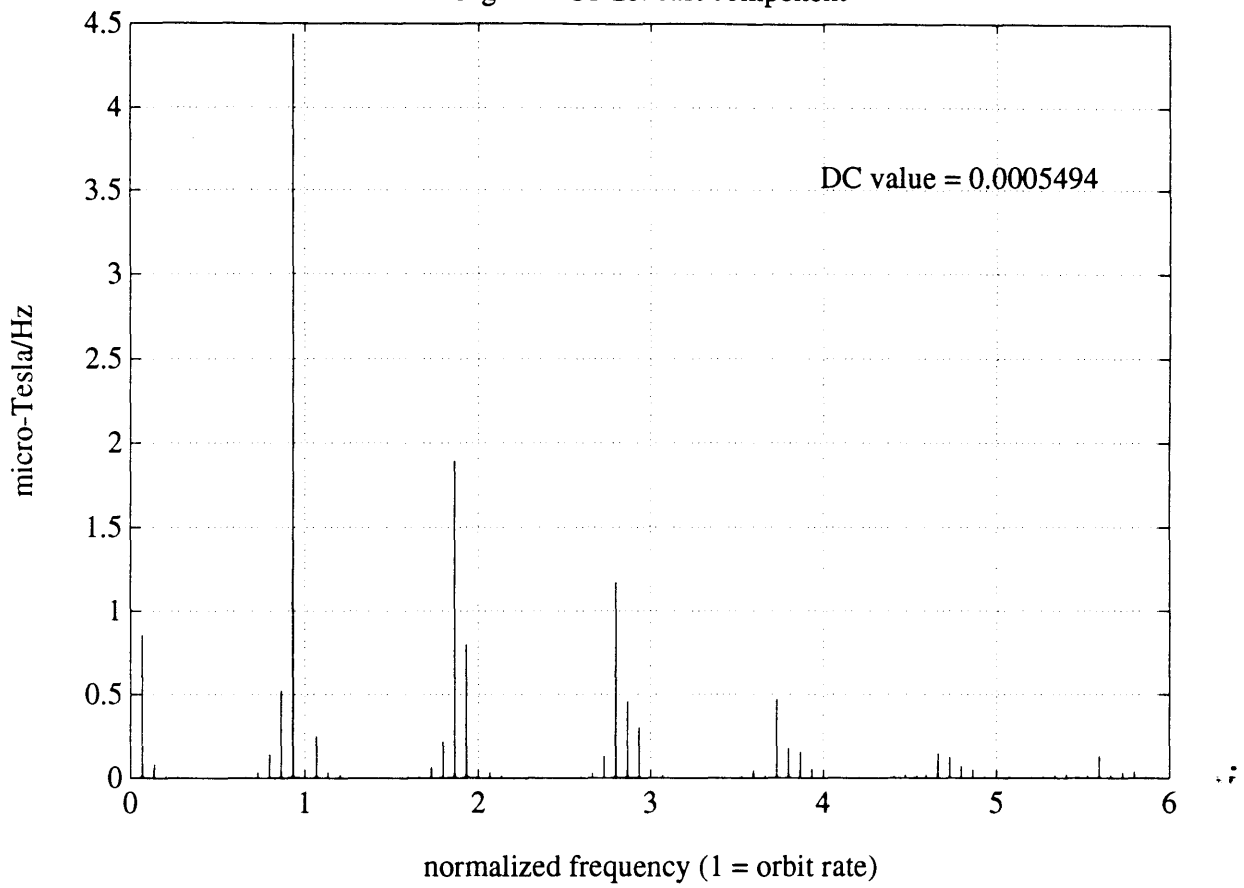
Figure AC1-2d: down component

DC value = 1.617
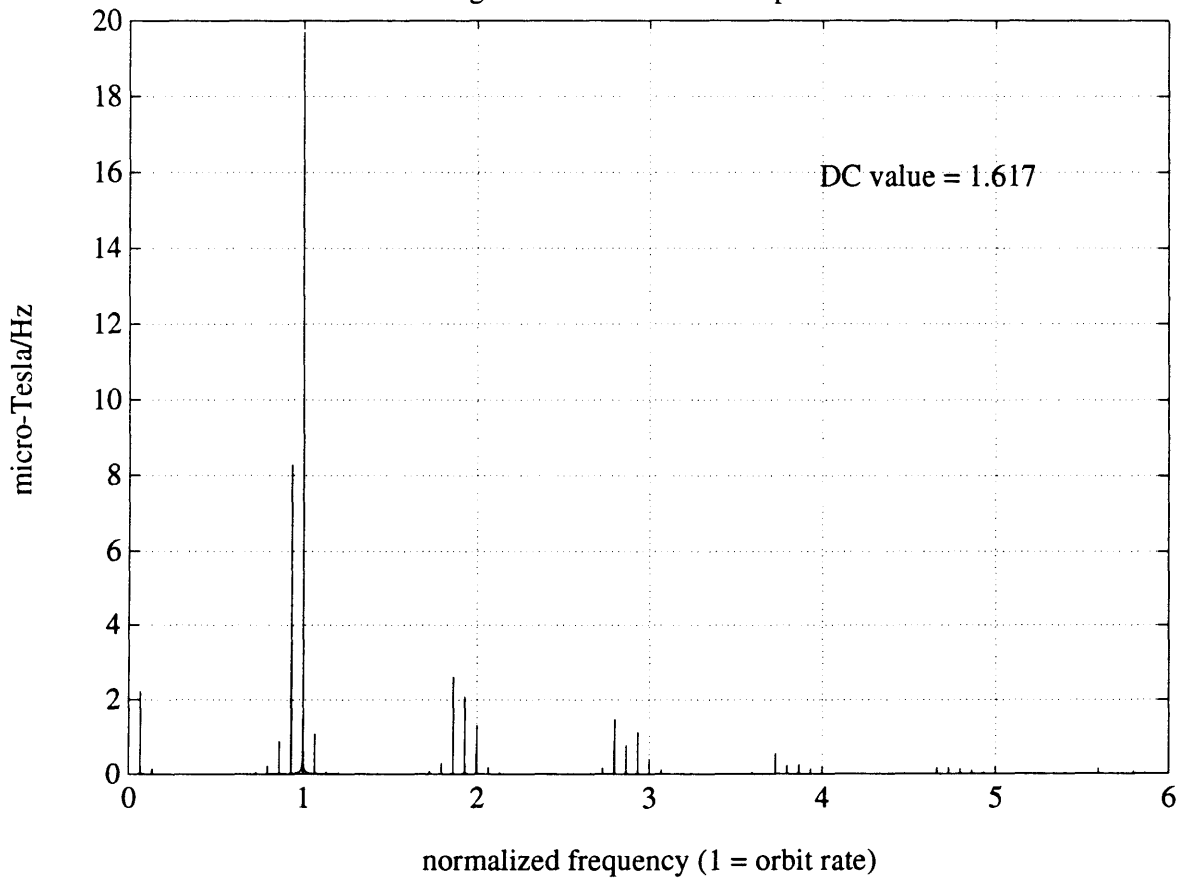
The code to generate the field and produce the plots in this section:

```
function [field] = maghist(inc, gr, asc, alt, ev, g, h, nmax)

% MAGHIST: magnetic field history on orbit
% [field] = maghist(inc, gr, asc, alt, ev, g, h, namx)
%
% inputs:
%     inc: orbit inclination in DEG
%     gr: initial prime meridian offset from vernal equinox in DEG
%     asc: initial orbit ascension offset from vernal equinox in DEG
%     alt: orbit altitude (above sphere surface) in NAUTICAL MILES
%     ev: EVENLY SPACED vector containing true anomalies in DEG
%     g,h, nmax: IGRF coefficients to be used and order of expansion
%                note units of g and h determine units of outputs
%                nmax determines order of expansion
% outputs:
%     ** all same length as ev
%     field: ev X 3 containing local [north east nadir] field components
%
% last updated 1/19/91 DHC

d2r = pi/180; r2d = 180/pi;
nm2km = 1.852; km2nm = 1/nm2km;

% orbit characteristic constants

re = 6371;            % earth mean radius in km
altkm = alt*nm2km;    % altitude in km
mu = 3.986e14;        % earth grav const (in Nm^2/kg)

torbit = (2*pi)/(sqrt(mu/((1000*(altkm+re))^3)));  % orbit period, sec
dgdeta = torbit/(24*60*60);                        % deg earth spin per deg eta

nodeprec = -2.06474e14*((altkm+re)^(-7/2))*cos(inc*d2r);  % node drift
dadeta = nodeprec*dgdeta*(1/360);     % deg node drift per deg eta
deta = ev(2) - ev(1);                 % spacing of eta vector
da = dadeta*deta; dg = dgdeta*deta;

% plow thru the orbits

for jj = 1:length(ev),
  eta = ev(jj);
  % perturb orbit and get local position
  gr = gr + dg;
  asc = asc + da;
  [coe ceo] = eqorb(asc, inc);
  [cge ceg] = greenequ(gr);
  [clo col] = orbloc(eta);
  % get longitude/colatitude, calculate mag field components
  [longi colat] = lvlhcoord(ceo, cge, col);
  [bx by bz] = igrf(nmax,colat,longi,alt,g,h);
```

```
   field(jj,:) = [bx by bz];              % field components
end
```

```
% code to take FFT's of output

[faxis nff] = freqaxis(etavec); % see below
mfft = (2/nff)*(abs(fft(magfield,nff))); % scaled for half window
mfft(1,:) = mfft(1,:)./2;            % DC component correction
```

```
function [faxis, nff] = freqaxis(etavec)

% takes in evenly space vector of true anomalies, creates frequency
% axis in times orbit rate;
% nff = largest power of 2 under length(etavec)

deta = etavec(2)-etavec(1);
nff = 2^floor(log(length(etavec))/log(2));
fstep = 360/(nff*deta);
fmax = 360/deta;
faxis = [0:fstep:fmax-fstep]';
```

Appendix C-2: Earth Field in $\mathcal{F}_B$

For visualization purposes, the results of the simulations in $\mathcal{F}_N$ frame are transformed to the body frame assuming a sun-pointed orientation and a designated roll angle about the sun vector, as defined in Section 3.1.6.

Since roll angle is defined about B_3 (= nominal sun vector), the B_3 field component is invariant with respect to the roll angle. Therefore, only the roll = 0 case is plotted. The B_1 and B_2 components are plotted for roll = 0, 30°, 45°, 60°, 90°, 120°, 135°, and 150° in the figures to follow.

The local field direction angles θ and Ψ, as defined in Chapter 4, are calculated from the field histories and are also plotted in the figures to follow.

The plots show the envelope of the histories, and are useful mostly in pointing out that no "special" roll angle exists about which to design a simplified control strategy, driving the discussion in Chapter 4.

For resolving the dark bands on the plots, some representative spectra are included in Figure AC2-1 for roll = 45°. It is seen that the major effects of transforming $\mathcal{F}_N$ to $\mathcal{F}_B$ are the moving of components at orbit rate to twice orbit rate (justifiable on geometric and multiplication/convolution arguments) and the introduction of a significant component at the node-regression rate of:

$$\frac{1}{15.06 \times \dfrac{360}{6.7}} = 0.0012 \text{ times orbit rate}$$

The claim made in Chapter 3 that the main field as measured in $\mathcal{F}_B$ contains little energy past about 6 times orbit rate is substantiated. Recall this guided magnetometer pre-filter design.

Next page:

Figure AC2-1 a,b: FFT of body component #1 at roll = 45°. Note the daily component is resolvable at the left of the top plot, but the node regression component requires the close-up on the bottom plot and is in fact the largest component. This can be verified by inspection of the roll = 45° time history several pages below .

Figure AC2-1a

DC value = 9.287

micro-Tesla/Hz

normalized frequency (1 = orbit rate)

Figure AC2-1b

DC value = 9.287

micro-Tesla/Hz

normalized frequency (1 = orbit rate)

x10-3
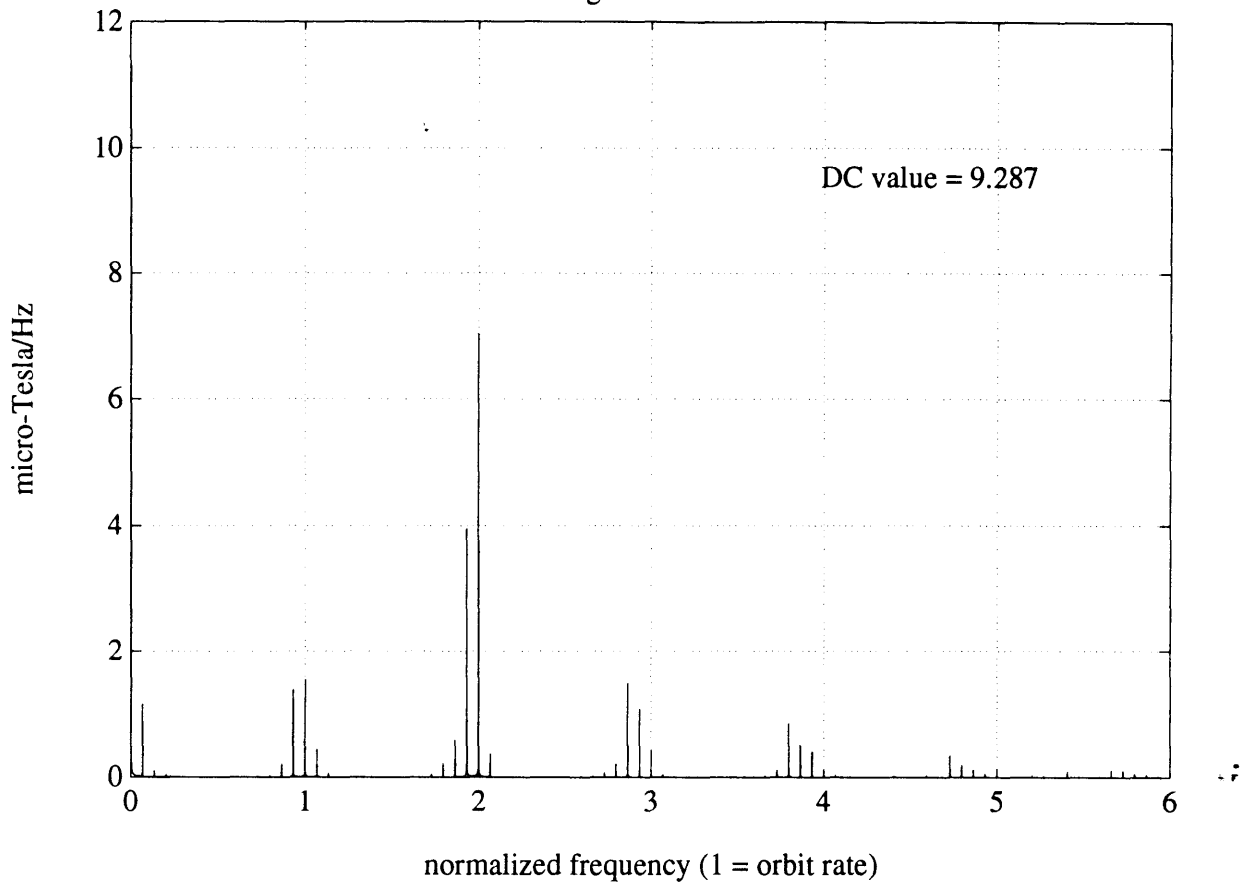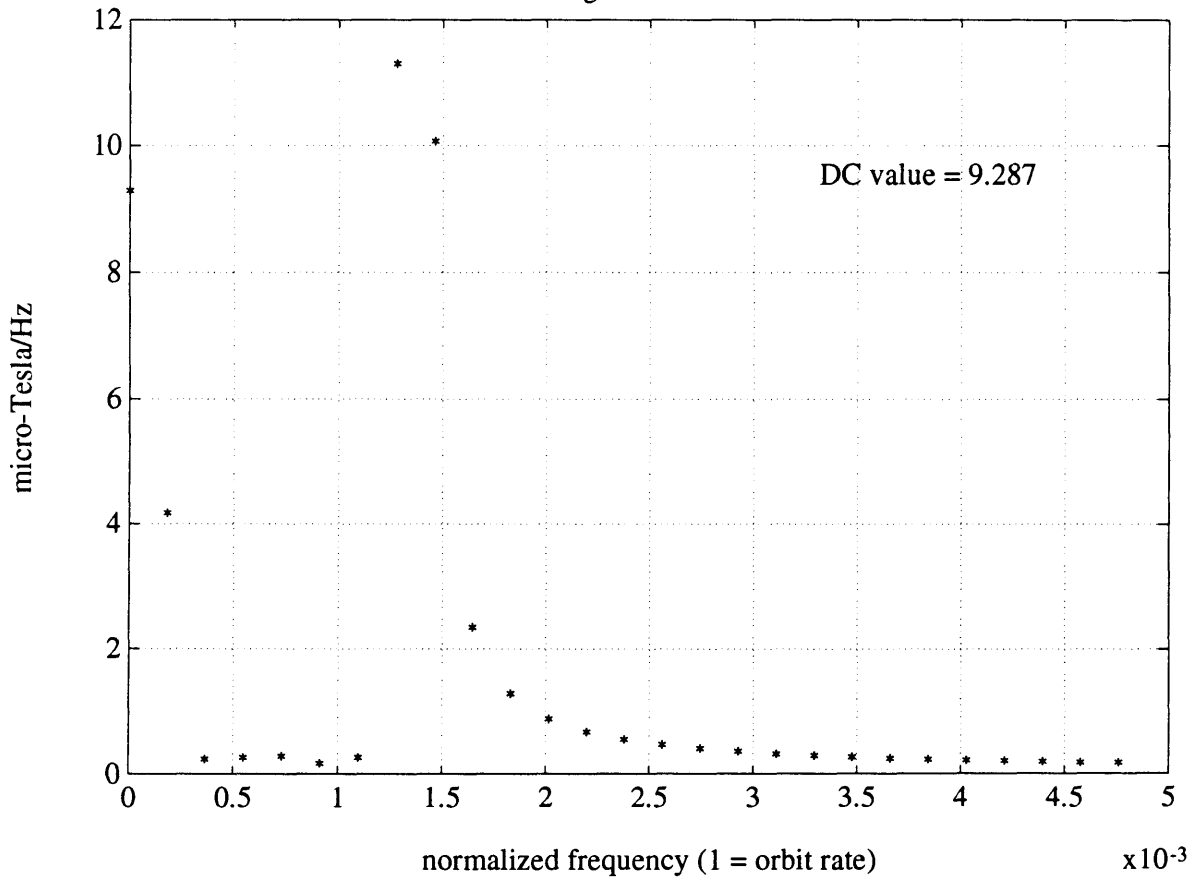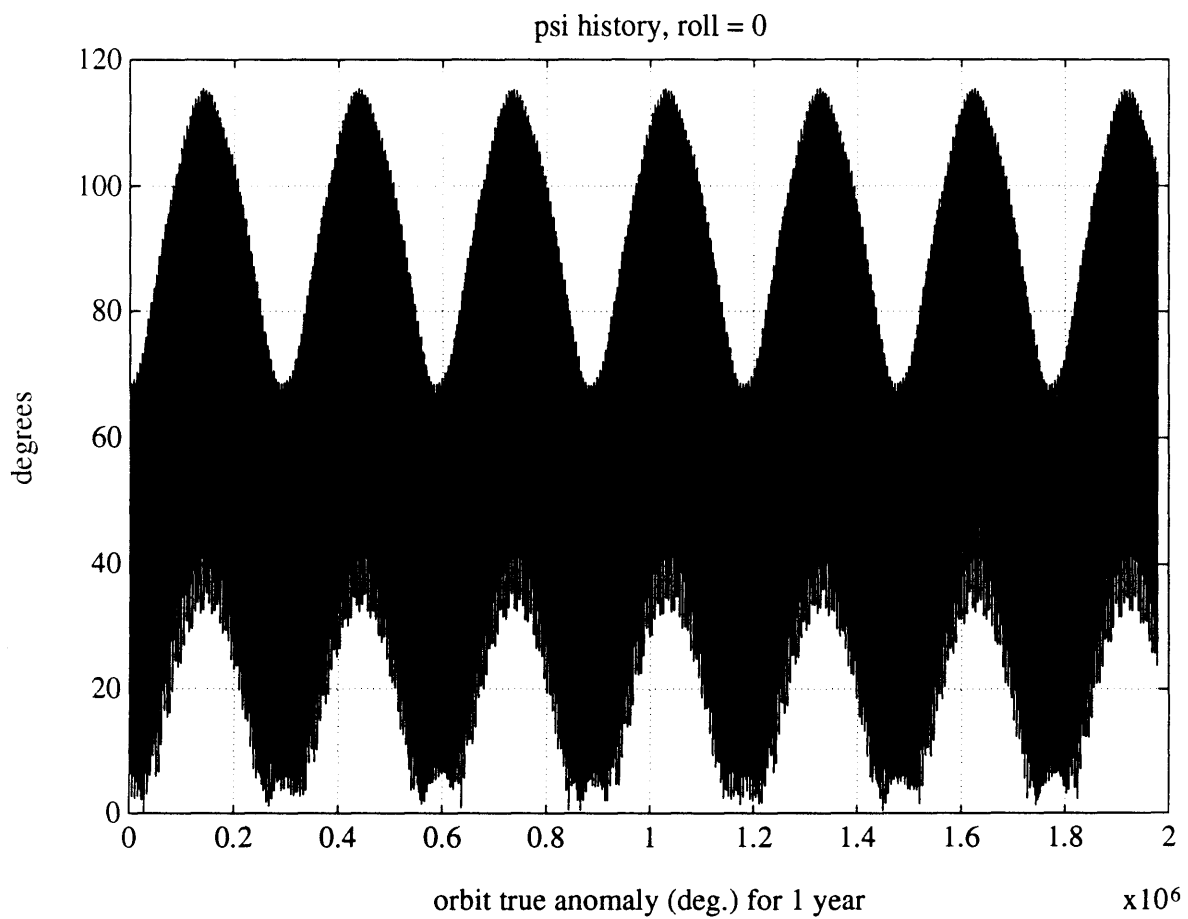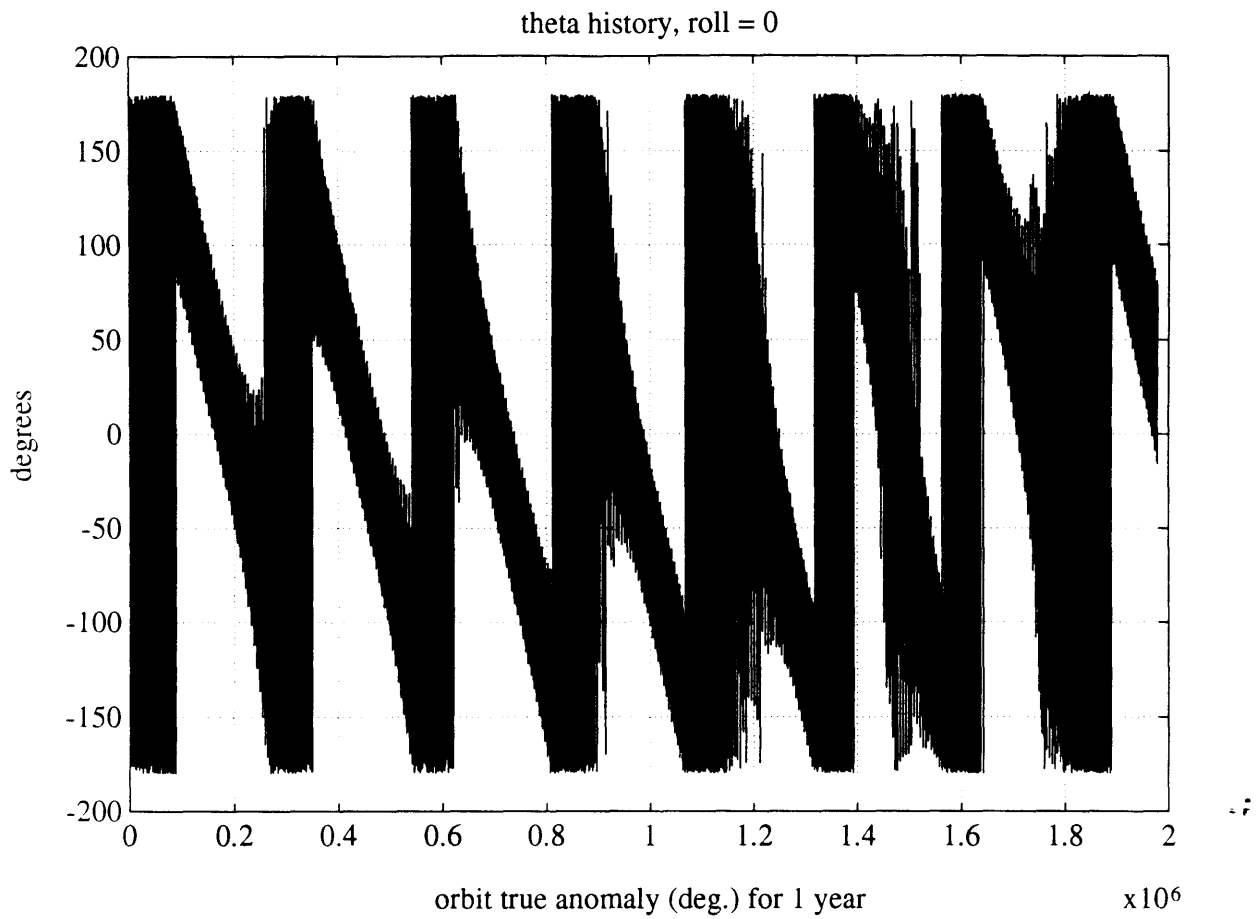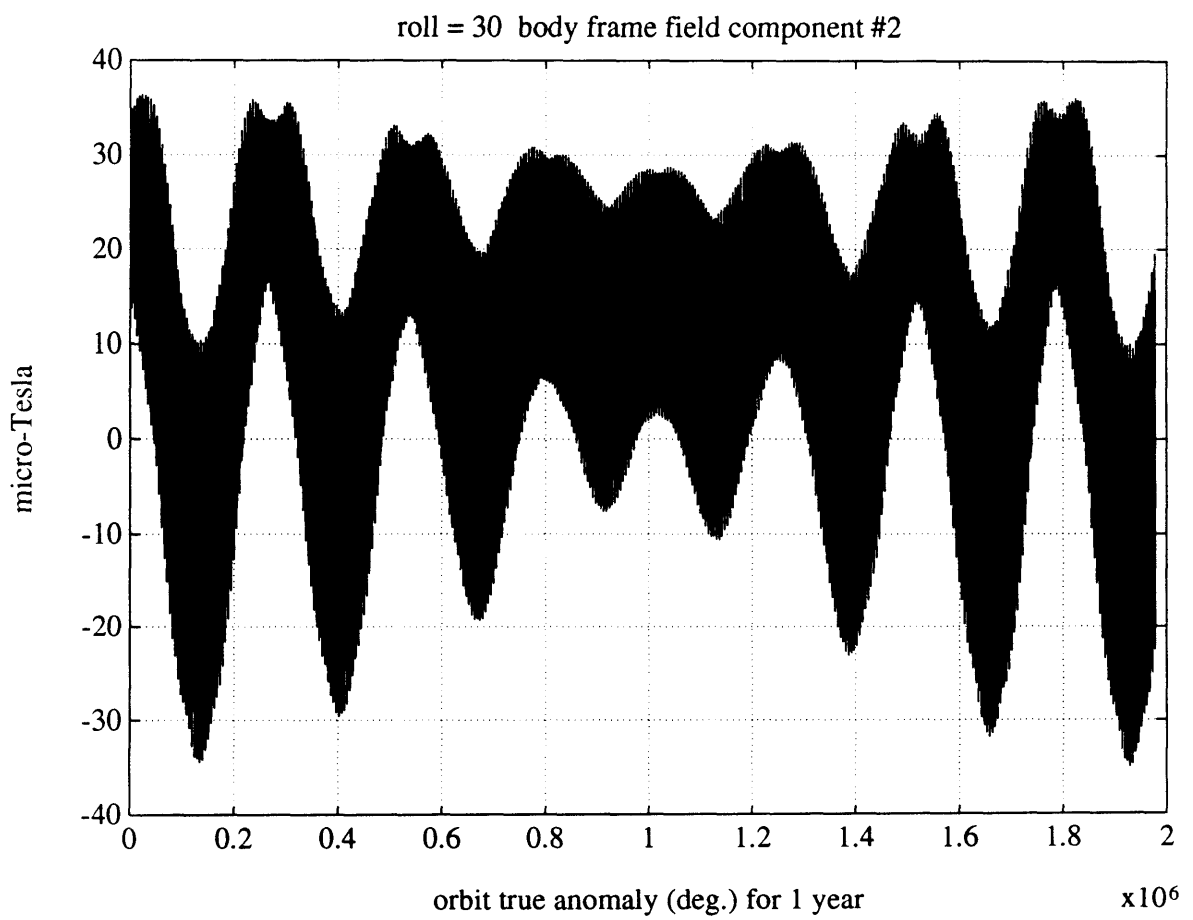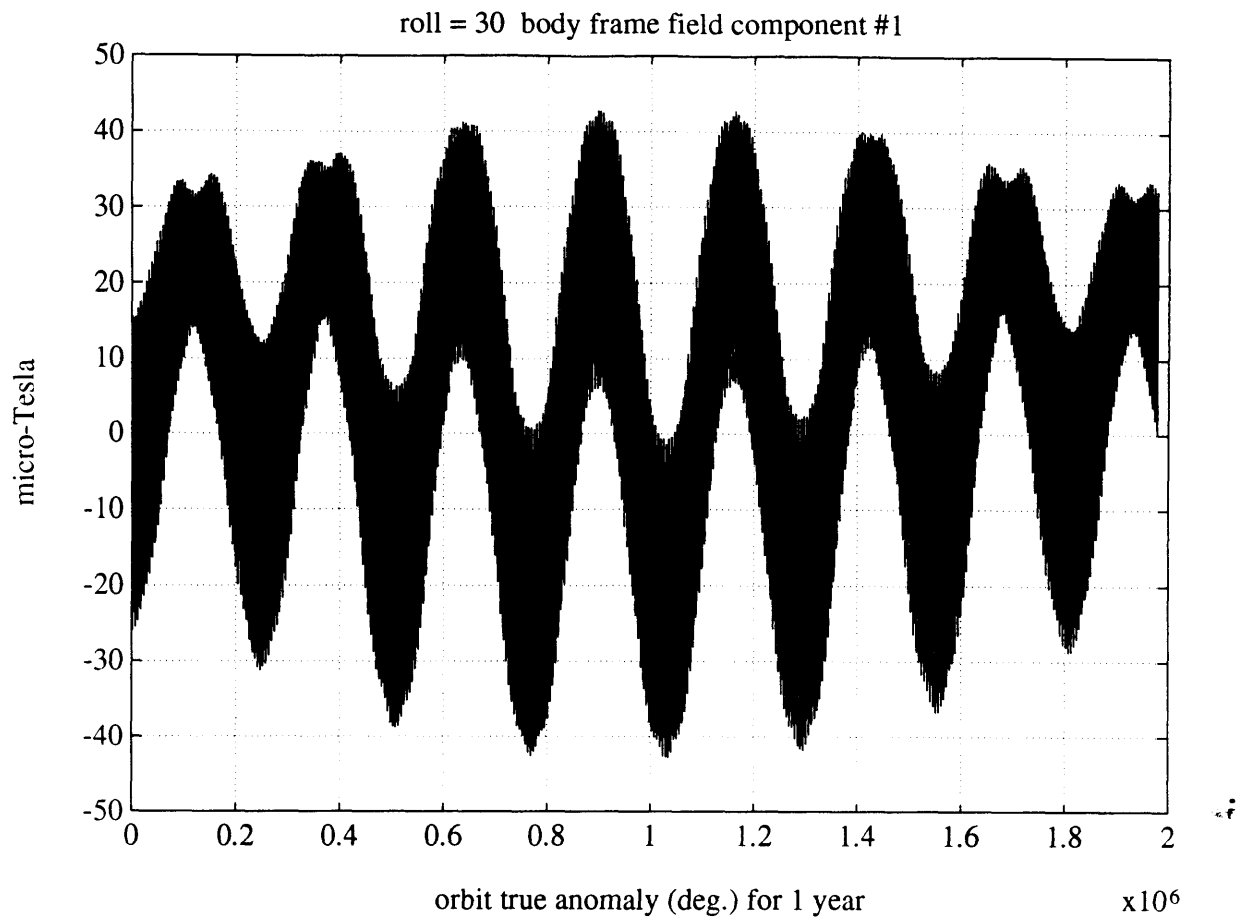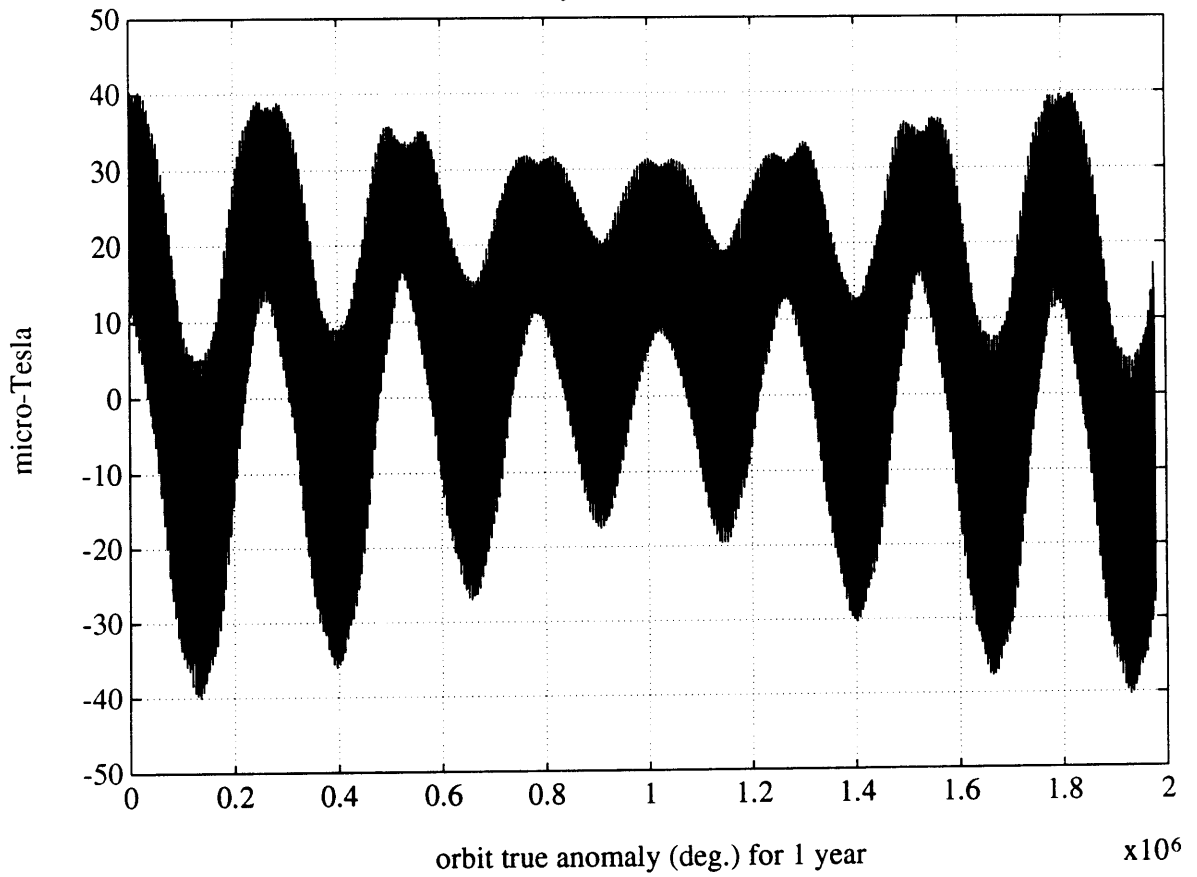
Next pages:

Simulation results transformed to $\mathcal{F}_B$ using a nominal sun-pointing attitude and for various roll angles. Since the nominal sun-vector lies along B_3, the B_3 field component is invariant with respect to the roll angle. Therefore, only the roll = 0 case is shown, in the first plot. The B_1 and B_2 component histories, along with θ and Ψ histories, are plotted for roll = 0, 30°, 45°, 60°, 90°, 120°, 135°, and 150° .

body frame field component #3 (independent of roll angle)

orbit true anomaly (deg.) for 1 year $\times 10^6$

body frame field component #3 (independent of roll angle)

orbit true anomaly (deg.) for 1 year $\times 10^6$

roll = 0 body frame field component #1

micro-Tesla

orbit true anomaly (deg.) for 1 year   x10⁶

roll = 0 body frame field component #2

micro-Tesla

orbit true anomaly (deg.) for 1 year   x10⁶

theta history, roll = 0

orbit true anomaly (deg.) for 1 year   x10⁶

psi history, roll = 0

orbit true anomaly (deg.) for 1 year   x10⁶

roll = 30 body frame field component #1

micro-Tesla

orbit true anomaly (deg.) for 1 year   x10⁶

roll = 30 body frame field component #2

micro-Tesla

orbit true anomaly (deg.) for 1 year   x10⁶

theta history, roll = 30

orbit true anomaly (deg.) for 1 year   x10⁶

psi history, roll = 30

orbit true anomaly (deg.) for 1 year   x10⁶

roll = 45 body frame field component #1

orbit true anomaly (deg.) for 1 year   x10^6

roll = 45 body frame field component #2

orbit true anomaly (deg.) for 1 year   x10^6

theta history, roll = 45

orbit true anomaly (deg.) for 1 year   x10⁶

psi history, roll = 45

orbit true anomaly (deg.) for 1 year   x10⁶

roll = 60 body frame field component #1

micro-Tesla

orbit true anomaly (deg.) for 1 year   x10⁶

roll = 60 body frame field component #2

micro-Tesla

orbit true anomaly (deg.) for 1 year   x10⁶

theta history, roll = 60

orbit true anomaly (deg.) for 1 year   x10⁶

psi history, roll = 60

orbit true anomaly (deg.) for 1 year   x10⁶

roll = 90 body frame field component #1

micro-Tesla

orbit true anomaly (deg.) for 1 year   x10⁶

roll = 90 body frame field component #2

micro-Tesla

orbit true anomaly (deg.) for 1 year   x10⁶

theta history, roll = 90

orbit true anomaly (deg.) for 1 year   x10⁶

psi history, roll = 90

orbit true anomaly (deg.) for 1 year   x10⁶

roll = 120 body frame field component #1

orbit true anomaly (deg.) for 1 year   x10⁶

roll = 120 body frame field component #2

orbit true anomaly (deg.) for 1 year   x10⁶

theta history, roll = 120

degrees

orbit true anomaly (deg.) for 1 year   x10⁶

psi history, roll = 120

degrees

orbit true anomaly (deg.) for 1 year   x10⁶

roll = 135 body frame field component #1

orbit true anomaly (deg.) for 1 year   x10^6

roll = 135 body frame field component #2

orbit true anomaly (deg.) for 1 year   x10^6

theta history, roll = 135

orbit true anomaly (deg.) for 1 year   x10⁶

psi history, roll = 135

orbit true anomaly (deg.) for 1 year   x10⁶

roll = 150 body frame field component #1

orbit true anomaly (deg.) for 1 year   x10⁶

roll = 150 body frame field component #2

orbit true anomaly (deg.) for 1 year   x10⁶

theta history, roll = 150

orbit true anomaly (deg.) for 1 year

psi history, roll = 150

orbit true anomaly (deg.) for 1 year