N91-17093

TWO AXIS POINTING SYSTEM (TAPS)
Attitude Acquisition, Determination, and Control

John D. Azzolini
David E. McGlew

Guidance and Control Branch
Space Technology Division
Engineering Directorate
Goddard Space Flight Center

ABSTRACT

The TAPS is a 2 axis gimbal system designed to provide fine pointing of STS borne instruments. It features center-of-mass instrument mounting and will accommodate instruments of up to 1134 kg (2500 pounds) which fit within a 1.0 by 1.0 by 4.2 meter (40 by 40 by 166 inch) envelope. The TAPS system is controlled by a microcomputer based Control Electronics Assembly (CEA), a Power Distribution Unit (PDU), and a Servo Control Unit (SCU). A DRIRU-II inertial reference unit is used to provide incremental angles for attitude propagation. A Ball Brothers STRAP star tracker is used for attitude acquisition and update.

The theory of the TAPS attitude determination and error computation for the Broad Band X-ray Telescope (BBXRT) are described. The attitude acquisition is based upon a 2 star geometric solution. The acquisition theory and quaternion algebra are presented. The attitude control combines classical position, integral and derivative (PID) control with techniques to compensate for coulomb friction (bias torque) and the cable harness crossing the gimbals (spring torque). Also presented is a technique for an adaptive bias torque compensation which adjusts to an ever changing frictional torque environment. The control stability margins are detailed, with the predicted pointing performance, based upon simulation studies, presented. Finally the TAPS user interface, which provides high level operations commands to facilitate science observations, is outlined.

NOTATION AND NOMENCLATURE

Matrices will be written as bold faced capital letters, e.g. **A**. Vectors will be written as a lower case letter e.g. $\vec{v}$. The vector inner product will be written as $\langle \vec{a}, \vec{b} \rangle$ (i.e. $\vec{a}$ *dot* $\vec{b}$). Quaternions will be written as lower case barred letters, e.g. $\bar{q}$. A quaternion is used to represent a the TAPS coordinate system with respect to an inertial coordinate system. It may be expressed as a vector part and a scaler part. If we define a rotation about a unit vector, $\hat{\varepsilon}$, of angle ϕ, it may be written as:

$$\bar{q} = (\sin\frac{\phi}{2}\,\hat{\varepsilon}, \cos\frac{\phi}{2})$$

The inverse rotation is denoted $\bar{q}'$ and is formed by negating the vector part of $\bar{q}$. For every attitude quaternion, $\bar{q}$, there is a unique equivalent attitude direction cosine matrix, **A**. For our purposes, $\bar{q}$ and **A** represent inertial to TAPS coordinate transformations. The column vectors of **A** are the inertial basis vectors expressed in TAPS coordinates. It follows that the row vectors of **A** are the TAPS basis vectors expressed in inertial coordinates.

ATTITUDE DETERMINATION

When the TAPS is in parked or stowed position, its coordinate system is defined to be nominally parallel to the STS coordinate system. The x axis is the shuttle roll axis, positive forward. The z axis points downward through the floor of the orbiter bay. The y axis points to starboard, forming a right handed cartesian frame. The inertial reference frame is the geocentric inertial coordinate system, 1950 mean (M50). The components of the TAPS and its coordinate system are shown in figure 1.

The attitude quaternion is initialized from the STS state vector. An attitude acquisition sequence is performed to remove the misalignments and uncertainties in the STS state vector. This sequence is discussed in the section on attitude acquisition.

The quaternion is then propagated using a closed form solution to the kinematic equations of motion.

$$\bar{q}(t+T) = e^{\frac{\Omega T}{2}}\bar{q}(t) \qquad (1)$$

where, T is the propagation sample period, and Ω is the skew symmetric matrix,

$$\Omega = \begin{bmatrix} 0 & \omega_z & -\omega_y & \omega_x \\ -\omega_z & 0 & \omega_x & \omega_y \\ \omega_y & -\omega_x & 0 & \omega_z \\ -\omega_x & -\omega_y & -\omega_z & 0 \end{bmatrix}$$

This solution assumes that the rate vector over the propagation interval is constant. By using the average rate over this interval, this approximation is close to ideal. The equivalent attitude direction cosine matrix, **A**, is computed each control cycle (16 Hz). The input to these computations are the scaled, drift corrected gyro rates.

ATTITUDE ERROR COMPUTATION

The attitude error is computed using the attitude direction cosine matrix, **A**, and the desired inertial target unit vector, $\hat{s}$. The target vector is transformed into TAPS coordinates.

$$\hat{s}_{TAPS} = \mathbf{A}\hat{s} \tag{2}$$

The required rotation vector is computed by taking the cross product:

$$\vec{e} = \hat{s}_{TAPS} \times \hat{b} \tag{3}$$

where $\hat{b}$ is the reference unit vector in the TAPS coordinate frame, (the vector we are trying to point). In normal operations, $\hat{b} = \hat{b}_i$ the BBXRT instrument boresight unit vector. During acquisition, $\hat{b} = \hat{b}_s$ the star tracker boresight unit vector. In all cases $\hat{b}$ is nominally aligned with the $-z_{TAPS}$ axis.

The magnitude of $\vec{e}$ is given by:

$$|\vec{e}| = |\hat{s}_{TAPS}||\hat{b}| \sin\theta = \sin\theta$$

where θ is the angle between $\hat{s}_{TAPS}$ and $\hat{b}$. We want the projection of θ on the x and y axes. One way to do this is to normalize $\vec{e}$ as:

$$\hat{e} = \frac{\vec{e}}{|\vec{e}|} = (e'_x, e'_y, e'_z)$$

Now, $\hat{e}$ is the unit rotation vector. The projection of θ on the x axis is then:

$$x_e = \sin^{-1}|\vec{e}|\, e'_x = \frac{\theta\, e_x}{\sin\theta}$$

and for small θ, $\sin\theta \sim \theta$ and:

$$x_e \sim e_x \tag{4}$$

similarly,

$$y_e \sim e_y \tag{5}$$

For TAPS, θ can be as large as 56 degrees. However, the control law limits the position error to a much smaller value which allows us to use this approximation. This is discussed in the attitude control section.

ATTITUDE ACQUISITION

The attitude acquisition sequence is designed to eliminate the alignment and other errors inherent in using the STS state vector to initialize the TAPS attitude quaternion. The sequence assumes that the TAPS has been deployed to parked index position, nominally along the shuttle -z axis, and that the shuttle state vector has been used to initialize the attitude quaternion. From that point the following steps are taken.

For each pair of stars:

1) Slew on gyros to the first star.
2) Store attitude information $\bar{q}_1$, $\hat{z}_1$, where $\hat{z}_1$ is the inertial TAPS z axis vector (the third row of the **A** matrix).
3) Null on the star using star tracker position.
4) Replace the attitude quaternion by the stored quaternion.
5) Slew on gyros to the second star.
6) Store attitude information $\bar{q}_3$, $\hat{z}_3$.
7) Null on the star using star tracker position.
8) Store attitude information $\bar{q}_4$, $\hat{z}_4$.
9) Compute updated attitude quaternion from the stored information.

If there are more than two stars, we may repeat the sequence using subsequent pairs. In this case, we repeat steps 1) through 4) for the last star to remove residual roll and pitch errors caused by numerical computational errors.

The mathematics of this acquisition sequence will now be explored. This analysis assumes perfect control, alignment ($\hat{b} = -\hat{z}_{TAPS}$), and sensors. We will use the subscript, t, to represent true attitude.

At the start of the sequence,

Let $\bar{q}_{t0}$ be the true attitude and let $\bar{q}_0$ be our on-board attitude, such that

$$\bar{q}_{t0} = \bar{q}_0 \, \bar{q}_{e0}$$

where $\bar{q}_{e0}$ is the error in initializing $\bar{q}_0$. The slew of step 1) of our attitude sequence defines an attitude rotation, $\bar{q}_{s1}$, after which,

$$\bar{q}_1 = \bar{q}_0 \, \bar{q}_{s1}$$

and,

$$\bar{q}_{t1} = \bar{q}_{t0} \, \bar{q}_{s1} = \bar{q}_0 \, \bar{q}_{e0} \, \bar{q}_{s1} = \bar{q}_1 \, \bar{q}_{e1}$$

The values of $\bar{q}_1$ and $\hat{z}_1$ are stored in step 2).

It is important to note that the first slew results in,

$$\vec{e} = 0 \rightarrow \mathbf{A}\,\hat{s} \times \hat{b} = 0 \rightarrow$$

$$\mathbf{A}(\bar{q}_1)\,\hat{s} = -\hat{z}_{TAPS} \qquad (6)$$

Step 3) of our sequence produces the rotation $\bar{q}_{n2}$, after which,

$$\bar{q}_2 = \bar{q}_1 \, \bar{q}_{n2}$$

and,

$$\bar{q}_{t2} = \bar{q}_{t1} \, \bar{q}_{n2} = \bar{q}_1 \, \bar{q}_{e1} \, \bar{q}_{n2} = \bar{q}_2 \, \bar{q}_{e2}$$

Again note that the star tracker null results in,

$$\mathbf{A}(\bar{q}_{t2})\ \hat{s} = -\hat{z}_{TAPS} \qquad (7)$$

From equations (6) and (7),

$$\mathbf{A}(\bar{q}_1)\ \hat{s} = \mathbf{A}(\bar{q}_{t2})\ \hat{s}$$

so,

$$\hat{s} = \mathbf{A}'(\bar{q}_1)\ \mathbf{A}(\bar{q}_{t2})\ \hat{s} = \mathbf{A}(\bar{q}_{t2}\,\bar{q}_1')\ \hat{s}$$

and,

$$\bar{q}_{t2}\,\bar{q}_1' = \bar{q}_s = \bar{q}_{-z}$$

where $\bar{q}_{-z}$ is some rotation about the $\hat{s}$ vector or, equivalently, the $-\hat{z}_{TAPS}$ vector expressed in inertial coordinates.

We can thus write,

$$\bar{q}_{t2} = \bar{q}_{-z}\,\bar{q}_1$$

Step 4) replaces the on-board quaternion $\bar{q}_2$ by the stored quaternion $\bar{q}_1$.

Proceeding with step 5),

$$\bar{q}_{t3} = \bar{q}_{t2}\,\bar{q}_{s3} = \bar{q}_{-z}\,\bar{q}_1\bar{q}_{s3} = \bar{q}_{-z}\,\bar{q}_3$$

The quaternion, $\bar{q}_3$, and $\hat{z}_3$ are stored in step 6).

Step 7) yields,

$$\bar{q}_{t4} = \bar{q}_{t3}\,\bar{q}_{n4} = \bar{q}_{-z}\,\bar{q}_3\,\bar{q}_{n4} = \bar{q}_{-z}\,\bar{q}_4$$

In step 8) we store $\bar{q}_4$ and $\hat{z}_4$.

We can compute the correction quaternion, $\bar{q}_{-z}$, from the stored attitude information at the three corners of the spherical triangle defined by our sequence. Figure 2 shows the acquisition sequence. The angle Γ is the gyro slew from the star 1 to star 2 and the angle Γ' is the angle between $\hat{z}_1$ and $\hat{z}_4$. The angle β is the slew produced by nulling on star 2. The angle α is the error about $\hat{z}_1$ in $\bar{q}_1$.

The quaternion $\bar{q}_{-z}$ is computed as follows. From figure 2 the angle α is given by:

$$\cos\alpha = \frac{\cos\beta - \cos\Gamma\cos\Gamma'}{\sin\Gamma\sin\Gamma'}$$

By assuming that $\Gamma \approx \Gamma'$ we can make the simplification,

$$\cos\alpha \approx \frac{\cos\beta - \cos^2\Gamma}{\sin^2\Gamma}$$

where,

$$\cos\Gamma = \langle \hat{z}_1, \hat{z}_3 \rangle$$

and,

$$\cos\beta = \langle \hat{z}_3, \hat{z}_4 \rangle$$

We compute the quaternion $\bar{q}_{-z}$ as:

$$\bar{q}_{-z} = (\pm\sin\frac{\alpha}{2}\,\hat{z}_1, \cos\frac{\alpha}{2}) \qquad (8)$$

where,

$$\cos\frac{\alpha}{2} = \sqrt{\frac{1+\cos\alpha}{2}}$$

and,

$$\sin\frac{\alpha}{2} = \sqrt{\frac{1-\cos\alpha}{2}}$$

The sign ambiguity in (8) is resolved by setting the sign equal to:

$$sgn\{\langle \hat{z}_1, (\hat{z}_4 - \hat{z}_1)\times(\hat{z}_3 - \hat{z}_1)\rangle\} = sgn\{\langle \hat{z}_1, \hat{z}_4\times\hat{z}_3\rangle\}$$

Note that the order of the cross product is chosen to provide the correct sign, since we are pointing the -z axis but storing the +z axis for use in these computations. The on-board attitude quaternion is then computed as:

$$\bar{q} = \bar{q}_{-z}\,\bar{q}_4 \qquad (9)$$

This computation eliminates the z axis error.

The effects of misalignments, gyro and star tracker quantization and noise, control errors, and STS inertial hold limit cycle motion have all been evaluated in simulation studies. These studies show that the worst case attitude acquisition error is approximately 2 arcminutes.

ATTITUDE CONTROL -- BASIC

In the x axis, the classic second order control law is,

$$T_x = -(K_p x_e + K_r \omega_x) \qquad (10)$$

where T_x is the x axis gimbal motor torque, and ω_x is the drift corrected, measured gyro rate about the x axis. Here, K_p is the position gain and K_r is the rate gain. They are chosen to achieve the desired control bandwidth, ω, and damping, ζ. The relations are:

$$\omega = \sqrt{\frac{K_p}{I_{xx}}}$$

and,

$$\zeta = \frac{K_r}{(2\omega I_{xx})}$$

where I_{xx} is the inertia about the x axis.

For large rotations, we wish to limit the rate, ω_x, to 1 degree/sec ($\frac{\pi}{180}$ rad/sec). From equation (10) notice that when

$$x_e = -\frac{K_r \omega_x}{K_p},$$

$$T_x = 0$$

We can exploit this by limiting the magnitude of x_e, as

$$\tilde{x}_e = sgn[min(x_{\lim}, |x_e|), x_e]$$

where,

$$x_{\lim} = \frac{K_r}{K_p}\frac{\pi}{180}$$

For typical values of K_r and K_p, $x_{\lim}$ is less than 1 degree and our small angle approximation is acceptable.

When we are in the normal pointing mode, and the position and rate errors are small, we add an integral control term to compensate for torque hangoff effects. The TAPS basic control law is then,

$$T_x = -(K_p \tilde{x}_e + K_r \omega_x + K_i \tilde{x}_\Sigma) \qquad (11)$$

where,

$$\tilde{x}_\Sigma = \sum \tilde{x}_e$$

Note that K_i must be adjusted from the continuous case by multiplying by the control interval, T, in seconds. This is because of the way we are approximating the integral of position.

$$\int_0^{nT} \tilde{x}_e(\tau)\, d\tau = \sum_{i=1}^{n} \int_{(i-1)T}^{iT} \tilde{x}_e(\tau)\, d\tau$$

By the mean value theorem,

$$= T\sum_{i=1}^{n} \tilde{x}_e(\xi_i) \sim T\sum_{i=1}^{n} \tilde{x}_e(iT) = T\tilde{x}_\Sigma$$

The development for the y axis follows the above exactly.

In order to allow slewing and to avoid stability problems, the integral compensation must only be added when we are truly holding. To accomplish this we compute the boolean variable:

$$HOLD := (|\tilde{x}_e| < p_{hold}) \wedge (|\tilde{y}_e| < p_{hold}) \wedge (|\omega_x| < \omega_{hold}) \wedge (|\omega_y| < \omega_{hold})$$

where, p_{hold} and ω_{hold} are the position and rate values we choose to consider the limits of hold mode pointing. When HOLD is true, we add in the integral compensation. When HOLD is false, we set $\tilde{x}_\Sigma$ and $\tilde{y}_\Sigma$ to zero and do not add the integral term.

During attitude acquisition and update, we must be able to control the TAPS pointing based upon the star tracker error signals. To accomplish this x_e and y_e in equations (4) and (5) are replaced by x_t and y_t, the star tracker error signals. The rest of the control law is unchanged.

The basic control law is also modified to limit the range of motion. We must avoid gimbal contact with limit stops. We achieve this by providing a software limit of 19 degrees from the gimbal index position. When the position error exceeds 19 degrees minus our current gimbal encode position, in the direction we are moving, we replace the position error by this difference. For the x axis the logic is,

```
if g_rate_x > 0 then
  if (g_limit - g_position_x) < x_e then
    x_e := g_limit - g_position_x
else
  if (-g_limit - g_position_x) > x_e then
    x_e := -g_limit - g_position_x
```

where g_rate_x, g_limit, and $g\_position_x$ are the x axis tachometer rate, the gimbal software limit, and the x axis encoder angle respectively.

ATTITUDE CONTROL -- FRICTION AND SPRING TORQUE COMPENSATION

The modelling and early operation of the TAPS gimbal provided insight into the frictional torque characteristics of the gimbal. The gimbal physically exhibits friction, which may be approximated by a coulomb friction model, and a spring torque, due to the bending of the cable harnesses, which may be approximated by an angular hook's law spring.

Early simulation studies demonstrated that we could not maintain the required 0.5 arcminute pointing jitter, during vernier thruster firings, with the basic control law alone. In order to improve this transient performance, 3 additional compensation terms are optionally added to the basic control law. These terms were a constant torque bias in the direction of motion, a spring torque term proportional to the gimbal encoder angle, and an adaptive bias torque term to correct for errors in the modelling and parameters of the other two terms.

The bias torque term is given by,

$$T_{xbias} = sgn(K_{xbias}, g\_rate_x) \quad (12)$$

The spring torque term is given by,

$$T_{xspring} = K_{xspring}\ (g\_position_x - g\_zero_x) \quad (13)$$

where g_zero_x is the encoder reference point for zero spring torque.

The adaptive bias torque term is computed as follows,

$$T_{xadap_{i+1}} = K_{adap}\, T_{xadap_i} + K_\tau\, T_{xbasic_i} \quad (14)$$

which is a first order difference equation in T_{xadap}. In this equation, K_{adap} is chosen to be less than one for stability, and K_τ if chosen as the inverse of the desired time constant to reach steady state. The torque, T_{xbasic_i}, is the torque computed by the basic TAPS control law. A separate adaptive term is computed for each direction of motion, for each axis. If we assume a constant motion in one direction,

$$T_{xbasic_i} = T_{friction} - T_{adap_i}$$

where $T_{friction}$ is a constant frictional torque to be overcome. The steady state solution of equation (13) is then given by,

$$T_{xadap_{ss}} = K_f\, T_{friction}$$

where,

$$K_f = \frac{K_\tau}{1 - K_{adap} + K_\tau}$$

By proper specification of the gains, K_f, the fraction of the frictional torque which will be removed by the adaptive torque term, can be selected.

The final control law is then given by,

$$T_x = -(K_p x_e + K_r \omega_x + K_i x_\Sigma) + T_{xbias} + T_{xspring} + T_{xadaptive} \quad (15)$$

ATTITUDE CONTROL -- ANALYSIS

Both the roll and pitch control loops are analyzed with appropriate rate, position and integral gains selected to provide at least 6 db of gain margin and approximately 30 degrees of phase margin. The analysis was performed using the Interactive Controls Analysis (INCA) program. The Nyquist frequency responses of the system are shown in figure 3 and 4. From the plots, the gain and phase margins are,

Roll Axis

Upper Gain Margin:	7.89 db at 11.7 radians/second
Lower Gain Margin:	11.5 db at 1.83 radians/second
Phase Margin:	28 degrees at 5.05 radians/second

Pitch Axis

Upper Gain Margin: 7.76 db at 11.7 radians/second
Lower Gain Margin: 11.6 db at 1.84 radians/second
Phase Margin: 28 degrees at 5.12 radians/second

Once the gains were selected, planer simulations representing the shuttle and experiment interconnected via the TAPS were developed. The desire was to show that the pointing stability requirements of maintaining a peak pointing error of less than 0.5 arcminutes could be achieved in the presence of vernier jet firings.

The MODEL translator was used to generate the FORTRAN rigid body simulations to evaluate the performance of the control laws developed in the above analysis. The x-z planer plant model equations, cast into matrix form, is given by,

$$M\frac{d\vec{X}}{dt} = \vec{T}$$

In this equation,

$$\vec{X} = \begin{bmatrix} X \\ Z \\ \theta_o \\ \theta_g \end{bmatrix}$$

where,

X ≙ x axis translational coordinate of shuttle motion
Z ≙ z axis translational coordinate of shuttle motion
θ_o ≙ Shuttle y axis rotational coordinate
θ_g ≙ Instrument y axis rotational coordinate

and,

$$\vec{T} = \begin{bmatrix} F_X \\ F_Z \\ T_o \\ T_g \end{bmatrix}$$

where,

F_X ≙ x axis component of vernier thrust
F_Z ≙ z axis component of vernier thrust
T_o ≙ Shuttle y axis torque
T_g ≙ Instrument y axis torque

and,

$$M = \begin{bmatrix} M_o+M_g & 0 & M_g r_{cm}\cos\theta_{o_1} & M_g r_g \cos\theta_{g_1} \\ 0 & M_o+M_g & -M_g r_{cm}\sin\theta_{o_1} & -M_g r_g \sin\theta_{g_1} \\ M_g r_{cm}\cos\theta_{o_1} & -M_g r_{cm}\sin\theta_{o_1} & M_g r_{cm}^2 + I_o & M_g r_{cm} r_g(\cos\theta_{o_1}\cos\theta_{g_1}+\sin\theta_{o_1}\sin\theta_{g_1}) \\ M_g r_g \cos\theta_{g_1} & -M_g r_g \sin\theta_{g_1} & M_g r_{cm} r_g(\cos\theta_{o_1}\cos\theta_{g_1}+\sin\theta_{o_1}\sin\theta_{g_1}) & M_g r_g^2 + I_g \end{bmatrix}$$

where,

M_o ≙ Orbiter mass

M_g ≙ Gimbal and instrument mass
I_o ≙ Orbiter moment of inertia
I_g ≙ Gimbal moment of inertia
r_{cm} ≙ Distance from orbiter center of mass to gimbal axis
r_g ≙ Distance from gimbal axis to center of mass of load
θ_{o_i} ≙ Initial shuttle y axis angle
θ_{g_i} ≙ Initial instrument y axis angle

A similar set was developed for the Y-Z plane.

For the simulation, a simple orbiter limit cycle was developed to illustrate the performance of the controller when the relative rate between orbiter and experiment changed sign.

The initial conditions were established so that the orbiter was in a limit cycle at a rate of 0.02 degrees/sec with a dead-zone of ± 0.5 degrees. The experiment was held inertially fixed while the orbiter was rotated underneath it. As can be observed in Figure 5, the experiment position error (top channel) exceeded 4 arcminutes, which is well beyond the 0.5 arcminutes pointing requirement, during the vernier jet firings.

The same initial conditions as above were established for the results shown in Figure 6 with the adaptive bias logic enabled. Initially the position error peaks well beyond the 0.5 arcminutes pointing requirement whenever a vernier jet fires. However, after several jet firings the bias term has adjusted to the point where the friction level is compensated by the bias term and not the integrator.

Illustrated in Figure 7 is the effect of a bias level nearly twice as high as required. The result is a temporary oscillation of the position error at the very beginning of the simulation run. The bias term being too high overdrives the experiment resulting in a switching between the positive and negative bias levels until they are adjusted to the appropriate value.

The curves shown in Figure 8 illustrate the adaptive changes that are occurring in the positive and negative bias terms. Since the bias levels were selected to be nearly twice as high as required, the positive adaptive term must adjust to decrease the positive term. Similarly the negative bias term must adjust to decrease the negative bias. In order to avoid this temporary oscillation an initial bias level lower than the anticipated friction should be selected and allowed to adjust upward.

Ideally the adaptive bias term is used to generate a torque that is equal to the friction level opposing the motion. Any torque opposing the motion that appears as a spring contaminates the bias capability. If such a spring torque exists in orbit, it will be possible to compensate for this torque by using the spring torque

control term. This will maintain the capability of the adaptive bias to effectively handle the friction level in the bearings.

In the above linear analysis it has been shown that the basic control law is stable with adequate margins while the simulation has shown that the pointing requirement of 0.5 arcminutes can be met with the additions to the basic control law.

TAPS USER INTERFACE

One of the design goals of the TAPS system was to provide science experimenters a user interface which allows them to control the pointing of their instrument. The TAPS Ground Support Equipment (TGSE) provides a complete command and telemetry interface to the user. Commands may be composed and processed in real time, processed from a command disk file, or stored in on-board command timelines for timed, sequenced execution. Telemetry is captured, processed, and displayed in real time.

The commands, called TAPS Mission Operations Commands (TMOCs), are of high level with a descriptive, English-like syntax. A sampling of commands which are of interest to an experimenter illustrates this.

Perform and inertial slew to a given M50 target vector:

```
OPER  ISLEW  S1,S2,S3
```

Perform an inertial slew to a catalogued target vector:

```
OPER  CSLEW    CAT_NO
```

Note that experimenters need not be concerned with slew sequences about gimbal axes to acquire a target. Rather, they may specify targets in inertial coordinates. The TAPS takes care of the rest.

CONCLUSIONS

The TAPS has been designed to provide pointing of the BBXRT instrument to an absolute accuracy of better than 4 arcminutes, with a pointing stability of better than 0.5 arcminutes. The design of the attitude error computation provides for simplified user control of observations. The control law provides rate limiting to avoid gyro saturation and position limiting to keep the gimbals within an acceptable range of motion.

It is our belief that the TAPS provides a flexible pointing capability which may be used for a variety of instruments. The system software has been designed in a highly modular fashion to allow the TAPS to accommodate other instruments and mission profiles. In particular, the architectural and algorithmic design required to point the Wide Angle Michelson Doppler Imaging Interferometer (WAMDII) instrument, which is an Earth limb pointer, had been completed prior to the suspension of the WAMDII program.

ACKNOWLEDGEMENTS

This paper provides a description of the TAPS system for the BBXRT-ASTRO mission. Much of the theory used to develop this system may be found, or derived from material, within Wertz (1). We have found this reference to be extremely valuable for any development work in this field and recommend it highly.

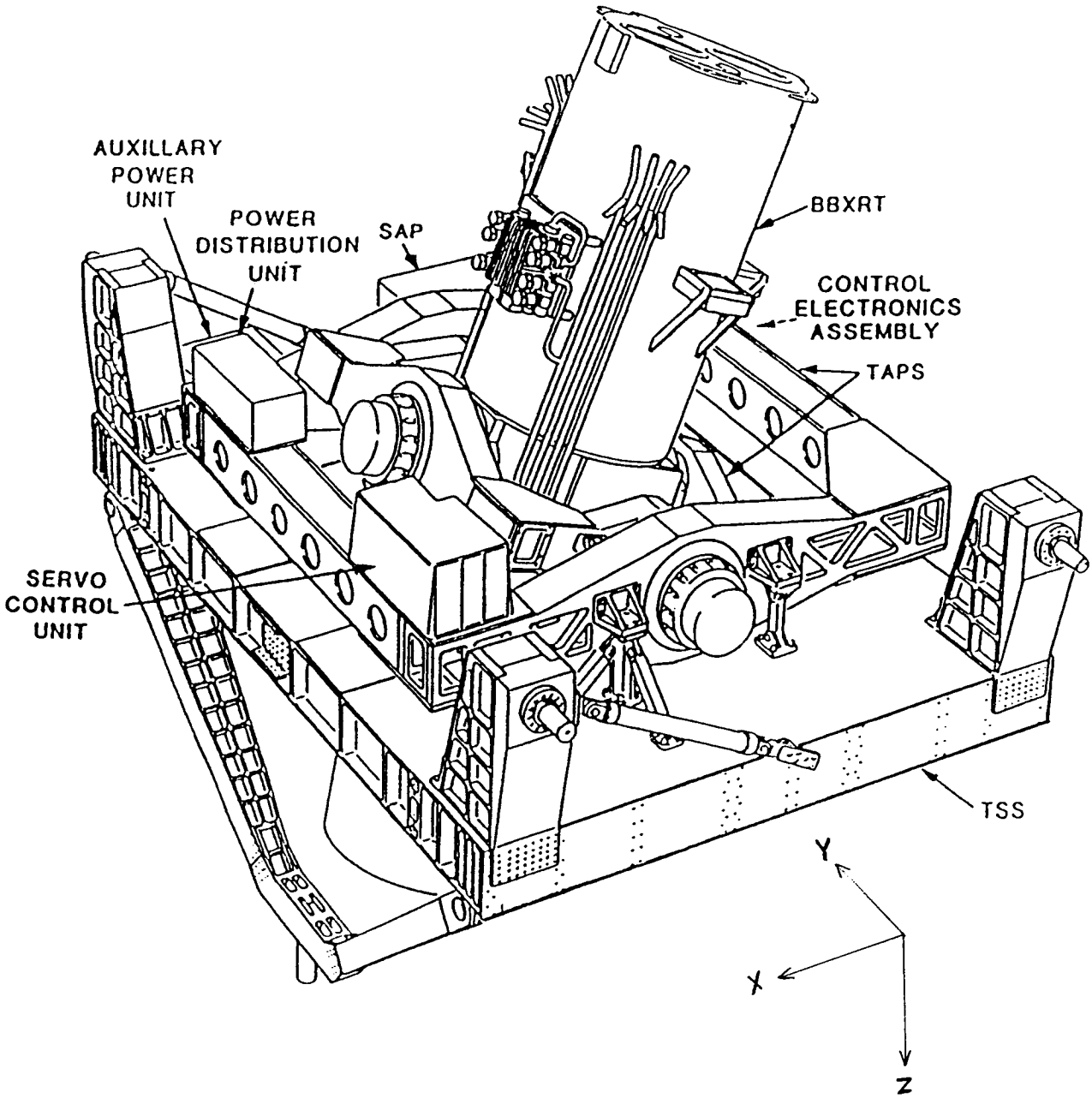
The authors gratefully acknowledge the members of the Guidance and Control Branch for helpful discussions and independent verification of the work described. In particular, we recognize Thomas Flatley for his MACSYMA based, independent verification of the attitude acquisition algorithm, and F. Landis Markley and Michael Femiano for their critical review of this paper and helpful suggestions.

REFERENCES

(1) Wertz, James R. Ed., Spacecraft Attitude Determination and Control, Holland: D. Reidel Publishing Company, 1980

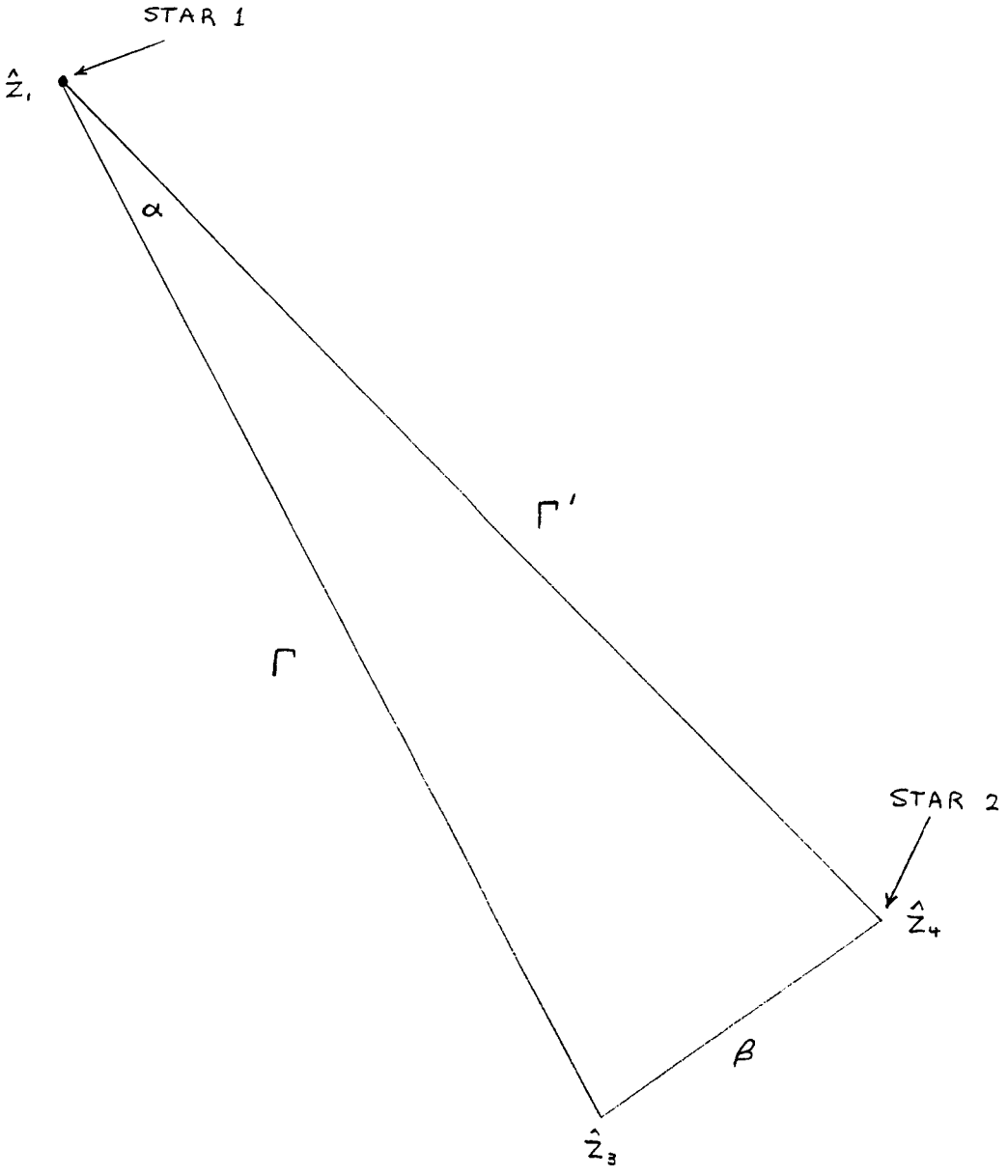
(2) Memorandum, John Azzolini and Frederick Hager to Walter Nagel, "TAPS Baseline Attitude Determination, Control, and Acquisition for the BBXRT SHEAL-II Mission, 1986

(3) Memorandum, David McGlew to John Azzolini, "TAPS-BBXRT Controller Gains and Adaptive Feedback",1989



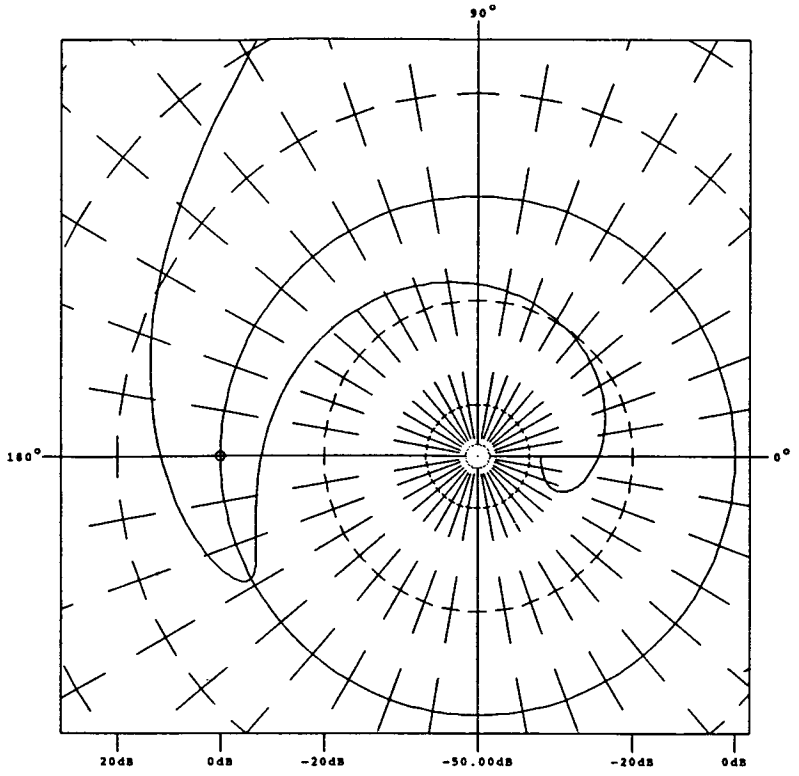
TWO AXIS POINTING SYSTEM
BROAD BAND X-RAY TELESCOPE

FIGURE 1

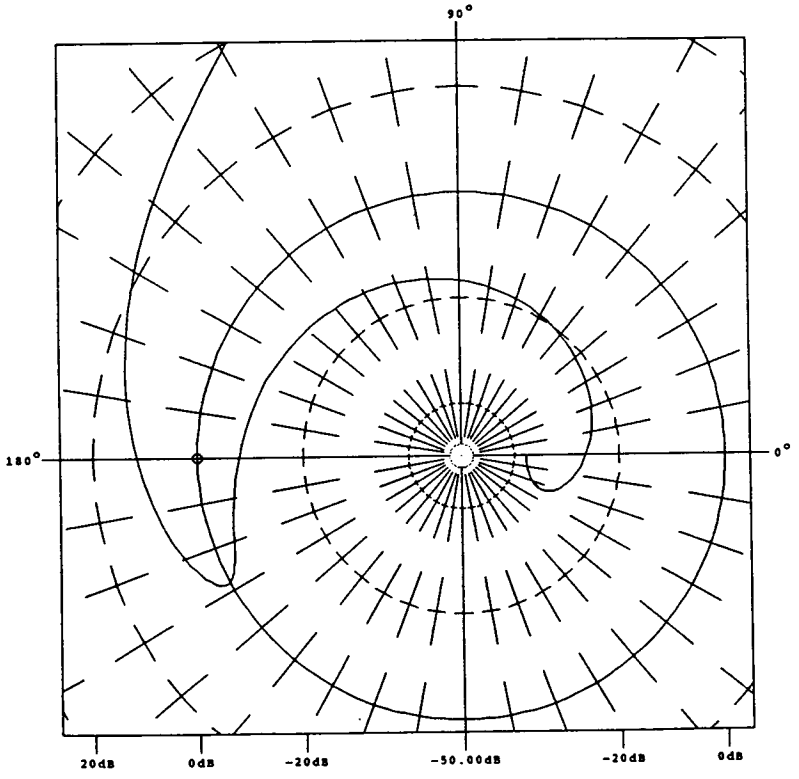


ATTITUDE ACQUISITION GEOMETRY

FIGURE 2



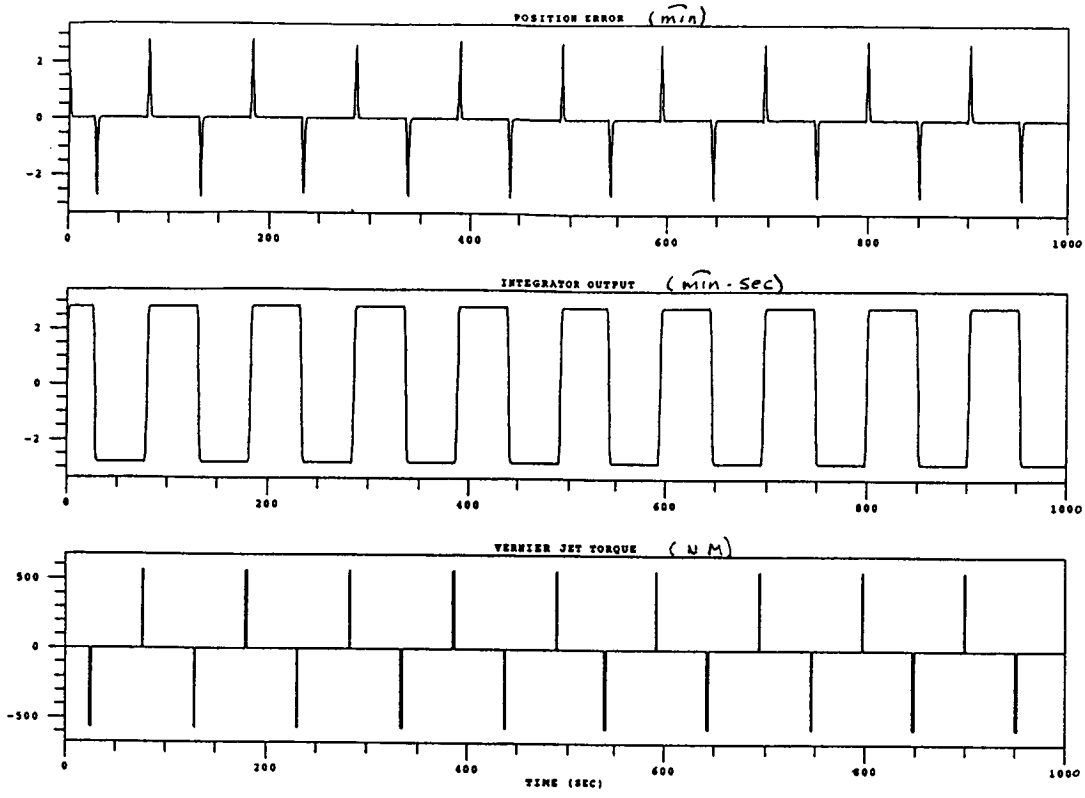
ROLL AXIS



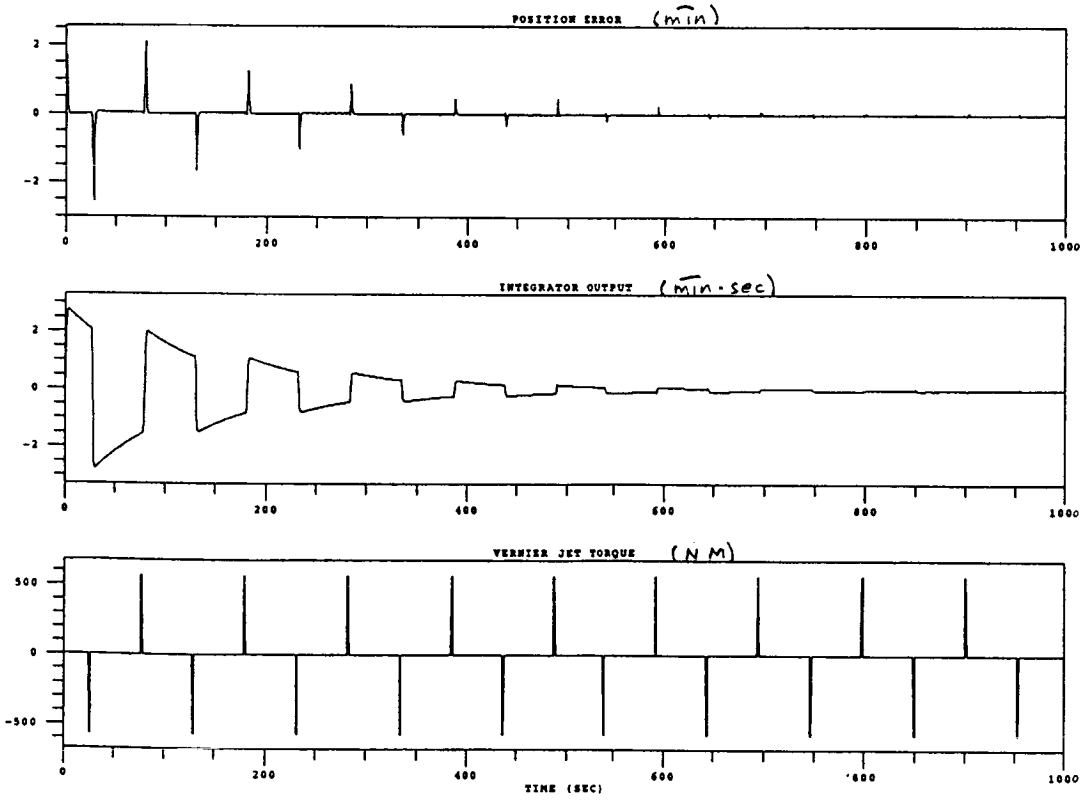
PITCH AXIS

NYQUIST STABILITY DIAGRAMS

FIGURES 3 & 4



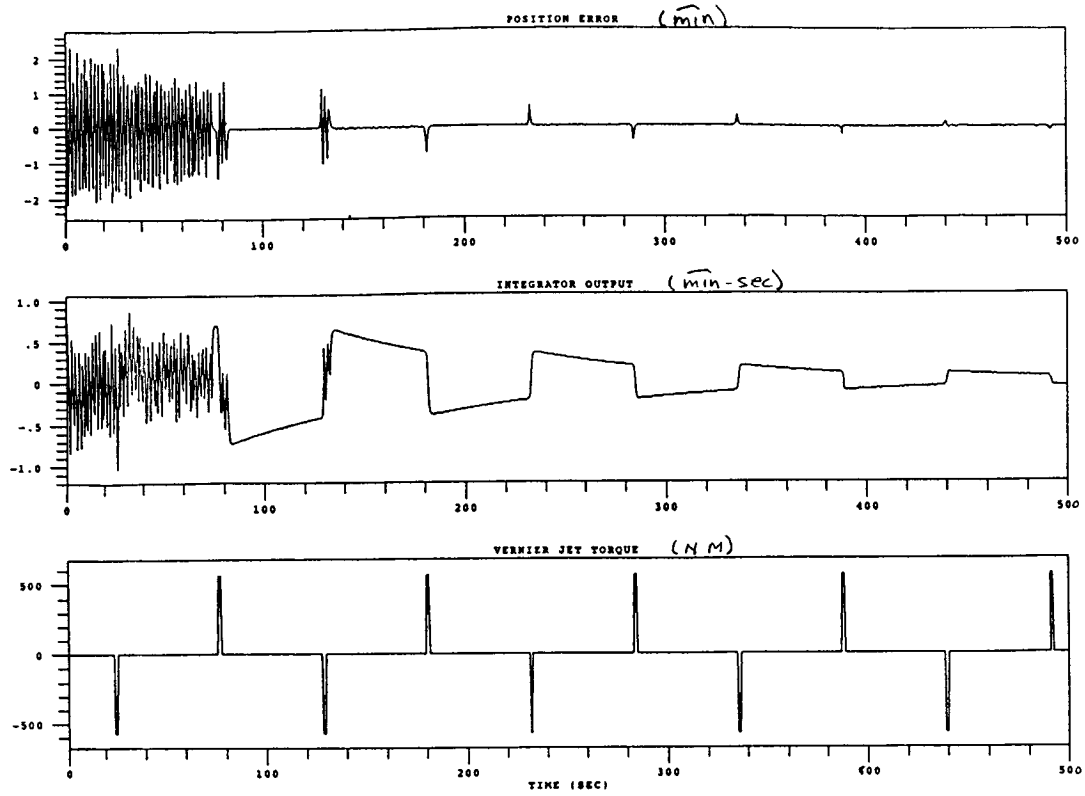
WITHOUT ADAPTIVE BIAS



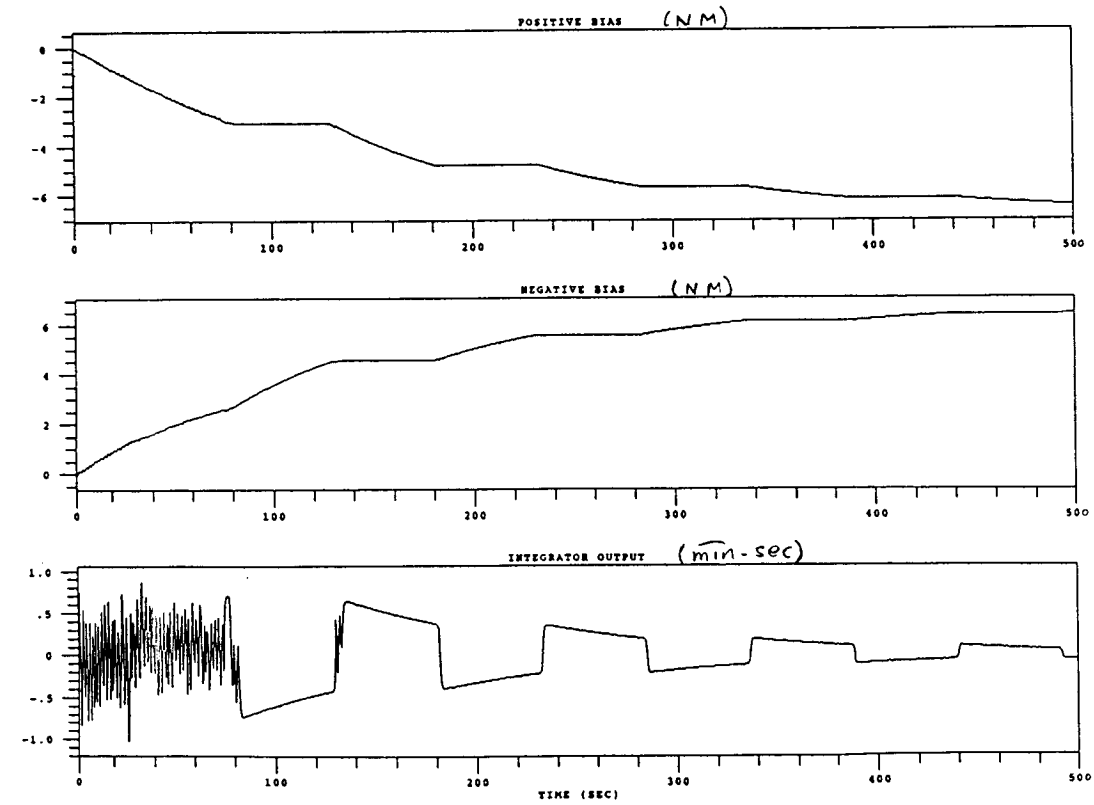
WITH ADAPTIVE BIAS

POINTING PERFORMANCE

FIGURES 5 & 6



POINTING PERFORMANCE



ADAPTIVE BIAS CONVERGENCE

EFFECTS OF INITIAL LARGE BIAS

FIGURES 7 & 8