AIAA-91-0590

A Scheme to Generate Thruster Firing Times for the Time-Optimal Reorientation of a Spinning Missile

E. Jahangir and R. M. Howe
University of Michigan
Ann Arbor, Michigan

29th Aerospace Sciences Meeting
January 7-10, 1991/Reno, Nevada

AIAA-91-0590

A SCHEME TO GENERATE THRUSTER FIRING TIMES FOR THE TIME-OPTIMAL REORIENTATION OF A SPINNING MISSILE

E. Jahangir* and R. M. Howe**
University of Michigan
Ann Arbor, MI 48109-2140

Abstract

A scheme to generate thruster firing times as functions of the initial and the desired state of a spinning missile is described. The missile is modeled as a rigid body which is symmetric about one axis and which has a large roll rate about this axis of symmetry. Control is achieved by a single reaction jet which, when fired, provides a constant moment about a transverse axis. Disturbance torques are assumed to be zero. The missile is assumed to have some arbitrary initial transverse angular velocity and it is desired to take it to some final attitude in minimum time while reducing the transverse angular velocity to zero. This minimum-time reorientation problem is usually handled by solving a TPBVP (Two Point Boundary Value Problem). The control history thus obtained is stored in an on-board computer and implemented on-line by table look-up. We describe a scheme which does not need to solve a TPBVP. Instead, coordinate transformations are used such that each backwards integration of the transformed state and costate equations yields a unique time-optimal trajectory. By storing the state and the associated time-optimal control at discrete points in time, a set of boundary condition points can be generated for which the minimum-time control is known. This set of points can later be used to generate a table of thruster firing times as functions of the current and the desired state of the missile. Some examples are plotted to illustrate the application of the concepts presented.

1. Introduction

Over the past three decades many papers and reports have treated various aspects of homing schemes and trajectory control associated with these schemes. Most of these papers consider surface-to-air or air-to-air missiles which use aerodynamic forces for trajectory control. With the advent of SDI, much attention has been focused on the interception of satellites or ICBM's outside the sensible atmosphere. Hence, aerodynamic forces cannot be generated for vehicle control. Instead, the thrust of a rocket engine is used to provide the necessary maneuver forces, with vehicle attitude control employed to point the thrust in the desired direction. Conventional thrust vector control systems tend to add both weight and complexity, and as a result counter the objective of minimizing the weight of the guided warhead. The simplest control involves a single thruster at right angles to the spin axis of the missile. In this scheme the missile is given a large roll rate and the thruster is turned on for a fraction of each revolution in roll and at the right time during each roll cycle so that the desired attitude changes are achieved. Meanwhile the main thruster, by producing a thrust component perpendicular to the flight path, provides the necessary trajectory changes.

The problem of attitude control of spinning rigid bodies has not received much attention recently, although some research has been reported on this topic in the 1960's. The reorientation problem of a spinning rigid body is conceptually different than the simple rest-to-rest maneuver of a non-spinning rigid body. Because of the spin of the body about its symmetry axis, application of any moment about the transverse axes generates a precessional motion. If the initial transverse angular velocity is not zero, the problem becomes even more difficult because the problem loses its symmetry.

Athans and Falb[2] consider the problem of time-optimal velocity control of a rotating body with a single axis of symmetry. They show that for a single fixed control jet, the system has the properties of a harmonic oscillator. Thus, a switching curve can be derived to implement the control scheme. The cases of a gimballed control jet and two control jets are also considered. No mention is made of the complete attitude reorientation problem, however. Howe[8] pro-

*Ph.D. Candidate in Aerospace Engineering
Student Member, AIAA
**Professor of Aerospace Engineering
Associate Fellow, AIAA

poses an attitude control scheme for sounding rockets. The main feature of this scheme is that it uses a single control jet. The control jet is fired for a fixed duration whenever certain conditions on direction cosines or transverse angular velocity are satisfied. This results in the alternate reduction of attitude error and transverse angular velocity, finally ending in a limit cycle. Some other references[16,3,1,5,6,7,12,15,13] discuss the problem of reorienting a rotating rigid body which has no initial transverse angular velocity. Windenknecht[16] proposes a simple system for sun orientation of spinning satellites. In this scheme the desired attitude is achieved by a succession of 180° precessional motions, each resulting in a small attitude change (small-angle approximations assumed valid), until the spin axis arrives at an attitude corresponding to the dead zone of the sun sensors. Cole *et al.*[3] prescribe the desired attitude change and solve for the necessary torques but give no details on mechanization. Other papers which propose active attitude control systems for spin stabilized vehicles have been published by Adams[1], Freed[5], and Grasshoff[6], but none of these explicitly discusses the reorientation problem. Grubin[7] uses the concept of finite rotations to mechanize a two-impulse scheme for reorienting the spin axis of a vehicle. If the torques are ideally impulsive, then the scheme is theoretically perfect. But in the case of finite-duration torquing, considerable errors can result. Wheeler[15] extends Grubin's work to include asymmetric spinning satellites, but the underlying philosophy is the same. Porcelli and Connolly[13] use a graphical approach to obtain control laws for the reorientation of a spinning body. Their results are only valid for small angles and small angular velocities. For this linearized case they prove that a two-impulse control scheme is fuel-optimal. Two sub-optimal control laws are then derived for the case of limited thrust based on the two-impulse solution. Most recently, Jahangir and Howe[11] have proposed a time-optimal scheme which does not require solving a TPBVP. This scheme can be used for the specific case when only two thruster firings are sufficient to complete the time-optimal attitude change maneuver. If the boundary conditions happen to lie outside this subset of the state space, the algorithm given by Jahangir and Howe fails to converge, since there does not exist a two-pulse time-optimal solution for such a case. If a control law is desired for boundary conditions which require more than two thruster pulses, we must solve a TPBVP involving ten nonlinear differential equations.

Both the two-pulse solution and the multiple-pulse solution require iterations and, therefore, can be costly in terms of the computer time required for the solution to converge and also in terms of the complexity of the iterative update scheme. Hence, an online iterative procedure does not appear to be practical for a real-time control algorithm. One possible alternative is to precompute the thruster firing times by solving a TPBVP for discrete values of the desired boundary conditions. These thruster firing times can then be stored as a table in an on-board computer and the control scheme can be implemented in real-time by table look-up and interpolation.

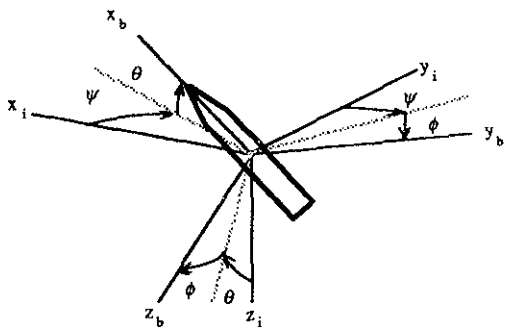

Figure 1: Axis systems.

Since we must store the control history as a function of the boundary conditions, we look for ways to generate a set of boundary condition points for which the thruster firing times are known without solving an iterative problem. To this end a new state vector is introduced in Section 4 which is related to the original state vector by a transformation. We will show in Sections 5 and 6 that we can generate a trajectory on the boundary of the set of reachable states by assuming a set of final conditions and integrating the transformed state and costate equations backwards in time. Since the boundary of the set of reachable states defines all the minimum-time trajectories, we can obtain all the desired boundary conditions and the associated time-optimal control histories by varying the final conditions over the range of their possible values. Finally, in Section 7 we illustrate the procedure by plotting some example time-optimal trajectories.

2. Equations of Motion

Figure 1 shows the orientation of the moving body axes x_b, y_b, z_b relative to the inertial reference axes x_i, y_i, z_i, and also the Euler angles ψ, θ, ϕ relating the two axis systems. The body axes origin is at the missile c.g. with the x_b-axis assumed to be the axis of symmetry; the y_b- and z_b-axes lie in a plane perpendicular to the longitudinal axis, x_b. The missile is modeled as a rigid cylindrical body. We also assume that the control jet is located in the x_b-z_b plane and pointed in the direction of the z_b-axis. When fired, the control jet generates a constant positive moment about the y_b-axis.

We have assumed no disturbances such as aero-

dynamic forces, gravity, solar radiation pressures, or structural damping. Because of the short flight times, these disturbances have negligible effect on the dynamics of the missile. Since no moment is applied about the x_b-axis, and since $I_y = I_z$ (the moments of inertia about the y_b- and z_b-axes are equal for a missile that is axially symmetric about its x_b-axis), it turns out that ω_x, the missile angular velocity component along the x_b-axis, is a constant equal to the initial spin velocity of the missile. We then obtain a set of five state equations: two dynamical equations involving the transverse angular velocities and three kinematical equations giving the rates of change of Euler angles. Thus

$$\dot{\omega}_y = \left(1 - \frac{I_x}{I_y}\right)\omega_x\omega_z + \frac{M_y}{I_y} \tag{1}$$

$$\dot{\omega}_z = -\left(1 - \frac{I_x}{I_y}\right)\omega_x\omega_y \tag{2}$$

$$\dot{\psi} = (\omega_y \sin\phi + \omega_z \cos\phi)\sec\theta \tag{3}$$

$$\dot{\theta} = \omega_y \cos\phi - \omega_z \sin\phi \tag{4}$$

$$\dot{\phi} = \omega_x + (\omega_y \sin\phi + \omega_z \cos\phi)\tan\theta \tag{5}$$

where

ω_y, ω_z = trasverse angular velocity components along the y- and z-axes, respectively,

ψ, θ, ϕ = Euler angles corresponding to yaw, pitch and roll, respectively,

I_x, I_y = the moments of inertia about the longitudinal and transverse axes, respectively,

M_y = the thruster torque about the y-axis.

For convenience we choose to write Eqs. (1)-(5) in terms of dimensionless variables and parameters in accordance with the following definitions:

$$\Omega_y = \frac{\omega_y}{\omega_x},\ \Omega_z = \frac{\omega_z}{\omega_x}.$$
$$A = 1 - \frac{I_x}{I_y},\ \lambda_y = \frac{M_y}{I_y\omega_x^2}$$
$$\text{dimensionless time } T = \omega_x t$$

Now, if we redefine the $-$ operator as differentiation with respect to the dimensionless time T, the equations become

$$\dot{\Omega}_y = A\Omega_z + \lambda_y \tag{6}$$

$$\dot{\Omega}_z = -A\Omega_y \tag{7}$$

$$\dot{\psi} = (\Omega_y \sin\phi + \Omega_z \cos\phi)\sec\theta \tag{8}$$

$$\dot{\theta} = \Omega_y \cos\phi - \Omega_z \sin\phi \tag{9}$$

$$\dot{\phi} = 1 + (\Omega_y \sin\phi + \Omega_z \cos\phi)\tan\theta \tag{10}$$

In order to write a state variable description of the system, we define the state $\mathbf{x}$ of the system as

$$\mathbf{x} = \begin{bmatrix} \Omega_y & \Omega_z & \psi & \theta & \phi \end{bmatrix}^T$$

and the control u as

$$u = \lambda_y$$

Eqs. (6)-(10) can now be written in the standard form.

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}u \tag{11}$$

where

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} Ax_2 \\ -Ax_1 \\ (x_1 \sin x_5 + x_2 \cos x_5)\sec x_4 \\ x_1 \cos x_5 - x_2 \sin x_5 \\ 1 + (x_1 \sin x_5 + x_2 \cos x_5)\tan x_4 \end{bmatrix} \tag{12}$$

$$\mathbf{g} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{13}$$

We assume that at the initial time, the missile body axis system coincides with the inertial axis system. The initial transverse angular velocity of the missile, however, is non-zero. We thus obtain the following *initial condition:*

$$\mathbf{x}_0 = \begin{bmatrix} x_{1,0} & x_{2,0} & 0 & 0 & 0 \end{bmatrix}^T \tag{14}$$

We want to find a control which will take this initial state to a desired state, described by some non-zero desired yaw and pitch angles and zero final transverse angular velocity, in minimum time. The desired final state vector, $\mathbf{x}_d$, can be written as:

$$\mathbf{x}_d = \begin{bmatrix} 0 & 0 & x_{3,d} & x_{4,d} & \text{free} \end{bmatrix}^T \tag{15}$$

We also assume an upper bound u_{max} on the control u. Thus, the constraint on the control can be written as:

$$0 \le u \le u_{max} \tag{16}$$

The numerical values for the two parameters, A and u_{max}, which will be used later in examples, are

$$A = 0.9,\ u_{max} = 0.02$$

This value of A corresponds to a length to diameter ratio of 3.775 for a cylindrical body of uniform density. A missile weighing 10 *lbs.* and having a uniform mass density of aluminum would have the following dimensions:

$$\text{length} = 12.30\ in.,\ \text{diameter} = 3.26\ in.$$

If the moment arm is half the length and the spin velocity is 50 *rad/sec*, $u_{max} = 0.02$ corresponds to a thrust of 2.79 *lbs.*

3. Time-Optimal Control Formulation

The problem, as stated in the previous section, is to find a control $u(T)$ which takes the initial state, $\mathbf{x}_0$, to the desired state, $\mathbf{x}_d$, in minimum time while satisfying the constraints $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}u$ and $0 \leq u \leq u_{max}$. This is one specific case of a general Mayer problem. Filippov[4] gives a theorem and proves the existence of an optimal control for a Mayer problem. At this time no general theorems are available on the uniqueness of optimal solutions for the one-sided controls, *i.e.*, $0 \leq u \leq u_{max}$. Therefore, we can only give necessary conditions for u^* to be an optimal control.

In order to derive an expression for the time-optimal control, we write the performance index

$$J = \int_{T_0}^{T_f} 1 dt \tag{17}$$

We want to minimize the performance index J under the constraints of Eq. (11) and (16). Thus, we can write the Hamiltonian

$$H = \mathbf{p}^T \dot{\mathbf{x}} - 1 \tag{18}$$

where $\mathbf{p}$ is the costate vector. The necessary conditions for u^* to be an optimal control are

$$\dot{\mathbf{x}}^* = \frac{\partial H}{\partial \mathbf{p}} = \mathbf{f}(\mathbf{x}^*) + \mathbf{g}u^* \tag{19}$$

$$\dot{\mathbf{p}}^* = -\frac{\partial H}{\partial \mathbf{x}} = \mathbf{H}(\mathbf{x}^*)\mathbf{p}^* \tag{20}$$

$$u^* = \begin{cases} u_{max} & \text{if } p_1^* > 0 \\ 0 & \text{if } p_1^* < 0 \end{cases} \tag{21}$$

and

$$\mathbf{x}(T_0) = \mathbf{x}_0 \tag{22}$$

$$\mathbf{x}(T_f) = \mathbf{x}_d \tag{23}$$

$$\mathbf{p}(T_0) = \begin{bmatrix} \text{free} & \text{free} & \text{free} & \text{free} & \text{free} \end{bmatrix}^T \tag{24}$$

$$\mathbf{p}(T_f) = \begin{bmatrix} \text{free} & \text{free} & \text{free} & \text{free} & 0 \end{bmatrix}^T \tag{25}$$

$$H(T_f) = 0 \tag{26}$$

where $\mathbf{H}(\mathbf{x}) = -\partial \mathbf{f}/\partial \mathbf{x}$. Eqs. (19) and (20) are the differential equations for the state and costate vector. Eq. (21) is derived from the optimality condition, *i.e.*, maximizing the Hamiltonian H. Eqs. (22) and (23) are the given boundary conditions and Eqs. (24)-(26) are derived from the transversality conditions. Furthermore, we note from the theory of necessary conditions that

$$\frac{\partial H(\mathbf{x}^*, \mathbf{p}^*, T)}{\partial T} = \frac{dH(\mathbf{x}^*, \mathbf{p}^*, T)}{dT} = 0$$

This, in addition to Eq. (26), shows that

$$H(\mathbf{x}^*, \mathbf{p}^*, T) = 0 \text{ for all } T \in [T_0, T_f]$$

Hence, in the problem we have 10 differential equations (Eqs. (19) and (20)) with 10 boundary conditions (Eqs. (22)-(25)) constituting a TPBVP. As mentioned earlier, the solution of this problem requires iterations and, therefore, is difficult to implement in real-time. In the actual missile, thruster firing times are computed off-line and are stored as a table in an on-board computer. Function generation is then used to compute the thruster turn-on and turn-off times as functions of the boundary conditions.

Instead of obtaining the thruster switch times by solving this iterative problem, we consider an alternative approach in the next section. An alternative optimal control formulation in terms of a new state vector is given. It is shown that, by assuming a set of final conditions and integrating backwards in time, we can generate time-optimal trajectories in the state space.

4. An Alternative Formulation of the Time-Optimal Control Problem

We define a new reference axis system. This axis system is fixed in the target and its x-axis points along the desired direction of the missile x_b-axis. The orientation of the missile with respect to an observer fixed in the target is given by the Euler angles, y_3, y_4, and y_5, where y_3, y_4, and y_5 correspond to yaw, pitch and roll, respectively. We also define $y_1 = x_1$ and $y_2 = x_2$. Thus, we can write a new state vector

$$\mathbf{y} = \begin{bmatrix} y_1 & y_2 & y_3 & y_4 & y_5 \end{bmatrix}^T$$

The two state vectors $\mathbf{x}$ and $\mathbf{y}$ are related by a transformation (see Section 6.3 for the transformation relations). The equations of motion can be written in terms of this new state vector and are given by:

$$\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y}) + \mathbf{g}u \tag{27}$$

We assume the initial and final conditions, respectively, to be

$$\mathbf{y}(T_0) = \mathbf{y}_0 = \begin{bmatrix} y_{1,0} & y_{2,0} & y_{3,0} & y_{4,0} & y_{5,0} \end{bmatrix}^T \tag{28}$$

$$\mathbf{y}(T_f) = \mathbf{y}_f = \begin{bmatrix} 0 & 0 & 0 & 0 & \text{free} \end{bmatrix}^T \tag{29}$$

A time-optimal control problem can be formulated for this system, similar to Section 3. We want to minimize the maneuver time, so we can again write the performance index as

$$J = \int_{T_0}^{T_f} 1 dt \tag{30}$$

under the constraints of Eqs. (23) and (16).

Proceeding with the derivation of the necessary conditions on the time-optimal control, we write the Hamiltonian

$$H = \mathbf{q}^T \dot{\mathbf{y}} - 1 \tag{31}$$

where $\mathbf{q}$ is the costate vector. The necessary conditions for u^* to be an optimal control are

$$\dot{\mathbf{y}}^* = \frac{\partial H}{\partial \mathbf{q}} = \mathbf{f}(\mathbf{y}^*) + \mathbf{g}u^* \tag{32}$$

$$\dot{\mathbf{q}}^* = -\frac{\partial H}{\partial \mathbf{y}} = \mathbf{H}(\mathbf{y}^*)\mathbf{q}^* \tag{33}$$

$$u^* = \begin{cases} u_{max} & \text{if } q_1^* > 0 \\ 0 & \text{if } q_1^* < 0 \end{cases} \tag{34}$$

The boundary conditions on the state variables are given by Eqs. (28) and (29). The boundary conditions on the Hamiltonian and the costate variables are derived from the transversality conditions:

$$H(T_f) = 0 \tag{35}$$

$$\mathbf{q}(T_0) = \mathbf{q}_0 = \begin{bmatrix} \text{free} & \text{free} & \text{free} & \text{free} & \text{free} \end{bmatrix}^T \tag{36}$$

$$\mathbf{q}(T_f) = \mathbf{q}_f = \begin{bmatrix} \text{free} & \text{free} & \text{free} & \text{free} & 0 \end{bmatrix}^T \tag{37}$$

We observe that this formulation still results in a TPBVP. If the initial state vector, $\mathbf{y}(T_0)$, is specified, an initial costate vector, $\mathbf{q}(T_0)$, must be determined which results in the desired final state and costate vectors. However, the state and costate vectors at the final time have some simple features. Each of the components of these vectors is either zero or free. Therefore, it is worthwhile to examine the system characteristics if the state and costate equations are integrated backwards in time starting at T_f.

In the next section we discuss a two dimensional system in terms of some simple sets in the state and costate space. By looking at the problem from a geometric point of view, we show for this 2-D system that the origin can be connected to all points in the set of reachable states in minimum time by varying the costate vector over $\mathcal{R}^2$ and integrating the system equations backwards in time. We note that this procedure of integrating the system backwards in time is fairly common. The most familiar example is the simple inertia system, $\ddot{x} = u$. In this system the state equations are integrated backwards in time, using the time-optimal control, to obtain a switching curve in the state space.

In Section 6 we apply this scheme to the complete nonlinear minimum-time attitude control problem of a spinning missile. Using the procedure analogous to Section 5, all minimum-time trajectories can be generated. By storing the state vector at discrete points in time along the minimum-time trajectories, we are able to generate a set of points for which the thruster firing times are known.

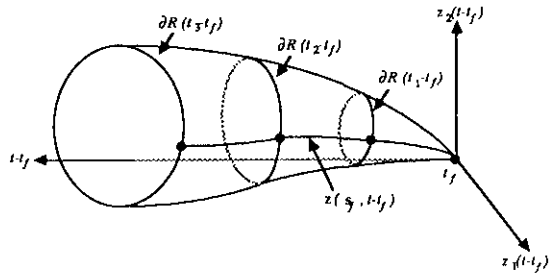

Figure 2: The set of reachable states for a 2-D system.

5. The Set of Reachable States for a 2-D System

In this section we examine the characteristics of the following 2-D system:

$$\dot{\mathbf{z}} = \mathbf{h}(\mathbf{z}, u)$$

where $\mathbf{z}$ is a 2×1 state vector and u is the scalar control. The desired final condition is assumed to be the origin, *i.e.*, $\mathbf{z}(t_f) = 0$. If we subject the system with final state $\mathbf{z}(t_f) = 0$ to all control histories and *integrate the system backwards in time starting at* t_f, we obtain a set of states that are reachable from the origin at time $t - t_f$, or simply the set of reachable states. We denote the set of reachable states simply as $R(t - t_f)$ in Figure 2. In the figure the boundary of the set of reachable states at time $t_i - t_f$ is denoted by $\partial R(t_i - t_f)$. It is well known from the geometric properties of the optimal control that the boundary of the set of reachable states defines all the minimum-time solutions. We, therefore, conduct the following experiment:

Let $u_{min}(\mathbf{z}, \mathbf{s})$ be the optimal control which is obtained by minimizing the Hamiltonian with respect to the control, where $\mathbf{s}$ is the 2×1 costate vector. Hence, if $\mathbf{z}^*$ is an optimal motion, it satisfies

$$\dot{\mathbf{z}}^* = \mathbf{h}(\mathbf{z}^*, u_{min}(\mathbf{z}^*, \mathbf{s}^*))$$

where $\mathbf{s}^*$ is a solution to the related costate equations. Clearly, $\mathbf{z}^*(t_f) = 0$.

In order to obtain a specific time-optimal trajectory, we need to assume some final conditions on the costate vector. We let $\mathbf{s}(t_f) = \mathbf{s}_f$, where $\mathbf{s}_f$ is an arbitrary constant vector. The system equations, $\dot{\mathbf{z}} = \mathbf{h}(\mathbf{z}, u_{min}(\mathbf{z}, \mathbf{s}))$, can now be integrated backwards in time starting from the final time t_f to obtain a trajectory $\mathbf{z}(\mathbf{s}_f, t - t_f)$ shown in the figure. The trajectory $\mathbf{z}(\mathbf{s}_f, t - t_f)$ represents the fact that it is a function of the specified final costate vector, $\mathbf{s}_f$, and the time parameter, $t - t_f$. This trajectory connects all points along its path to the origin in minimum time. Also, as mentioned earlier, this minimum-time trajectory lies on the boundary of the set of reachable states. By storing $\mathbf{z}(\mathbf{s}_f, t - t_f)$ at discrete points

in time, we can obtain a set of points in the state space for which the time-optimal control history is known. If we assume a different initial $\mathbf{s}_f$, another time-optimal trajectory is obtained. By varying $\mathbf{s}_f$ over $\mathcal{R}^2$ and integrating the system for each $\mathbf{s}_f$ backwards in time, all trajectories on the boundary of the set of reachable states can be generated. In this way all the reachable states can be obtained at discrete intervals and the associated control history can be stored as a function of these states.

In the next section we employ the techniques described here to obtain a set of boundary condition points for which the control history is known for the case of minimum-time reorientation of a spinning missile.

6. An Examination of the "y" System

We use here the idea of backwards integration and examine the system behaviour. Similar to the previous section, we conduct the following experiment.

Four of the final state variables are zero, as given in Eq. (29). We assume that $y_5(T_f) = y_{5,f}$, where $y_{5,f}$ is an arbitrary constant. Similarly, we assume an arbitrary value for the final costate vector, $\mathbf{q}(T_f) = \mathbf{q}_f$, where $q_{5,f} = 0$. The expression for the time-optimal control is given in Eq. (34). Once all the final conditions on the state and costate variables are specified, we start the integration at T_f and integrate the state and costate equations backwards in time. At each numerical integration step, we obtain a $\mathbf{y}(T')$ where $T' = T - T_f$. This trajectory connects all points along its path to the specified final state in minimum time. By varying the first four components of the final costate vector $\mathbf{q}_f$ over $\mathcal{R}^4$ (the fifth component is zero as given in Eq. (37)), the entire boundary of the set of reachable states is generated starting from the specified $\mathbf{y}_f$. We note here that the only non-zero component of the state vector is $y_{5,f}$, which corresponds to the roll angle of the missile. This angle can take on values in the range $[-\pi, \pi]$. By varying $y_{5,f}$ in this range, and following the aforementioned procedure of generating the boundary of the set of reachable states, all the time-optimal solutions can be generated. During the integration, the state vector, $\mathbf{y}$, and the corresponding thruster switch times can be stored at discrete points in time. Hence, this procedure gives a set of $\mathbf{y}$ points for which the control history given in terms of the thruster firing times is known.

The control scheme can then be implemented in real-time by using the thruster switch times which are stored at discrete values of the transformed state vector, $\mathbf{y}$. However, in the actual missile the desired attitude is measured with respect to the moving missile frame, whereas the vector $\mathbf{y}$ gives the orientation of the missile with respect to an observer fixed in the target. Hence, it is desirable to store the boundary conditions in terms of the original state vector $\mathbf{x}$. The boundary conditions in terms of the $\mathbf{x}$ vector are given in Eq. (14) and (15). The state vector $\mathbf{y}(T')$ can be transformed back to our original system to obtain the corresponding boundary conditions $x_{1,0}$, $x_{2,0}$, $x_{3,d}$, and $x_{4,d}$.

The procedure to generate the control history as a function of the boundary conditions can be summarized in the following way:

1. Initialize $\mathbf{y}_f$ and $\mathbf{q}_f$.

2. Integrate the $\dot{\mathbf{y}}$ and $\dot{\mathbf{q}}$ equations backwards in time and at each $T'_n = n\Delta T$ obtain $\mathbf{y}(T'_n)$, where ΔT is the time interval chosen to give desired data-point spacing between $\mathbf{y}(T'_n)$ and $\mathbf{y}(T'_{n+1})$, and n is a positive integer. Note that the numerical integration step can be a submultiple of ΔT.

3. Transform $\mathbf{y}(T'_n)$ to get the boundary conditions in the original form, $x_{1,0}$, $x_{2,0}$, $x_{3,d}$, and $x_{4,d}$. Store the thruster switching times as functions of these four variables. Note that if $q_1 > 0$, then $T_1 = 0$, and similarly if, $q_1 < 0$ then $T_1 > 0$, where T_1 is the first turn-on time of the thruster.

Each of these steps is discussed in the following sections.

6.1. Initialization of $\mathbf{y}_f$ and $\mathbf{q}_f$

As indicated earlier, five of the variables at T_f are zero.

$$y_{1,f} = y_{2,f} = y_{3,f} = y_{4,f} = q_{5,f} = 0$$

The other five variables at T_f are free. These must be varied over all possible values to obtain the optimal control history as a function of the boundary conditions. The variable $y_{5,f}$ corresponds to the roll angle and, thus, is confined to

$$y_{5,f} \in [-\pi, \pi]$$

The space over which $q_{1,f}$, $q_{2,f}$, $q_{3,f}$, and $q_{4,f}$ must be varied is a subset of the costate space. We refer to this subspace as $\mathcal{Q}$.

Definition: The space $\mathcal{Q}$ is defined as

$$\mathcal{Q} = \{\mathbf{q}(T_f) : q_5(T_f) = 0, q_i(T_f) = \pm 1,$$
$$q_j(T_f) \in [-1, +1], i, j = 1, 2, 3, 4, j \neq i\}$$

An algorithm to vary these variables over the range of their possible values is given below:

do $y_{5,f} = -\pi, \pi, \delta$ where δ is the desired spacing between values of $y_{5,f}$

do $i = 1, 4$
$q_{i,f} = \pm 1$
do $q_{j,f} = -1, +1, \Delta$ where $j \neq i$ and Δ is the desired spacing between values of $q_{j,f}$

During the implementation of our scheme to generate time-optimal solutions, we observe that a constant Δ (uniform spacing in $q_{j,f}$'s) does not result in a uniform span of the entire desired space of time-optimal solutions. We find that when $q_{j,f}$ is close to zero, very small Δ is needed to span the set of desired time-optimal solutions. Conversely, when $q_{j,f}$ is not close to zero, Δ need not be small.

6.2. Integration of State and Costate Equations

A practical issue in the implementation of the scheme given in Section 6 is the choice of a numerical integration algorithm and the handling of discontinuities that occur when the control switches from on to off or *vice versa*.

The RK-4 fixed-step algorithm is used to integrate the state and costate equations. We utilize the fact that analytic solutions for y_1 and y_2 can be obtained from Eq. (11). Thus, the equations for $\dot{y}_1$ and $\dot{y}_2$ do not have to be integrated numerically. The analytical solutions for y_1 and y_2 are also used to obtain the half- and full-frame derivative estimates of the remaining state and costate variables, as required in the RK-4 integration algorithm.

There is a discontinuity in these derivatives when the application of the control u starts or stops. This switching time is a function of q_1, the first component of the costate vector, as given by Eq. (34). If this discontinuity occurs within an integration step, it can cause large errors in the numerical solution. To reduce these errors, the step size must be chosen small enough to meet some integration error criterion, which can result in excessive computational time. Several papers have been written suggesting special methods to circumvent this difficulty. We choose the method proposed by Howe, Ye and Li[9] for its accuracy and ease of implementation. In this scheme, at each successive time step, q_1 is tested to see whether it has switched sign. If it has not, the integration proceeds to the next step. If switching has occurred, the time of its occurrence is computed by combining a fixed-point simplified Hermite interpolation with a continued fraction formula. Hermite interpolation is also used to compute the state and costate variable values at the crossover time. The RK-4 algorithm is then used to integrate through the remainder of the fixed-time step.

We define the following notation:

$$T_i = ih, \qquad i = 0, 1, 2, \ldots$$

$$u_i = u(T_i); \qquad \mathbf{y}_i \approx \mathbf{y}(T_i); \qquad \mathbf{q}_i \approx \mathbf{q}(T_i)$$

where h is the integration step size and $\mathbf{y}_i$ and $\mathbf{q}_i$ are the numerical approximations to the exact solution $\mathbf{y}(T_i)$ and $\mathbf{q}(T_i)$. The following computational procedure is used at the ith step.

1. Integrate from T_i to T_{i+1} with the RK-4 algorithm. In this integration u_i is used for all derivative evaluations over the interval.

2. Determine if $q_{1,i+1}$ has changed sign with respect to $q_{1,i}$. If not, repeat 1 starting at T_{i+1}. If q_1 has changed sign ($q_{1,i+1}q_{1,i} < 0$), it means that q_1 has a zero over the interval $T_i \leq T' \leq T_{i+1}$ and the discontinuity has been crossed. In that case, proceed to step 3.

3. Using a fixed-point simplified Hermite interpolation between T_i and T_{i+1} followed by the continued fraction zero-finder described by Howe, Ye and Li[9], determine the zero T_z of q_1. Again, using Hermite interpolation, determine $\mathbf{y}(T_z)$ and $\mathbf{q}(T_z)$.

4. Change the control u from 0 to u_{max} (or from u_{max} to 0, as appropriate) and integrate from T_z to T_{i+1} to recompute $\mathbf{y}_{i+1}$ and $\mathbf{q}_{i+1}$.

5. Return to step 1 and repeat the steps starting at T_{i+1}.

6.3. Transformation Relations Between x and y

Let us consider a vector $\mathbf{r}$ originating at the missile c.g. We let $\{r_m\}$ represent the components of the vector $\mathbf{r}$ in the missile body axis frame; $\{r_t\}$ represent the components of the same vector in the frame $\mathcal{T}$; and $\{r_f\}$ represent the components of the same vector in the frame $\mathcal{F}$. The frame of reference $\mathcal{T}$ is an arbitrary axis system whose x-axis coincides with the desired direction of the missile x_b-axis. The frame of reference $\mathcal{F}$ is defined to be the missile axis sytem at the end of the maneuver. Previously, we have defined $x_{3,d}$ and $x_{4,d}$ to be the yaw and pitch angles describing the orientation of the desired direction of the missile x_b-axis with respect to the missile body axis system. In addition, we define $x_{5,d}$ to be the roll angle of the frame $\mathcal{F}$ with respect to the missile body axis system. It should be borne in mind that $x_{5,d}$ is free in the formulation of the optimal control problem in Section 3. Thus, we can write.

$$\{r_f\} = [\mathrm{C}\,(x_{3,d}, x_{4,d}, x_{5,d})]\,\{r_m\} \qquad (38)$$

where [C] is the direction cosine matrix and is defined in the following way:

$$[\mathbf{C}(\psi,\theta,\phi)] = \begin{bmatrix} \cos\psi\cos\theta & \sin\psi\cos\theta & -\sin\theta \\ -\sin\psi\cos\phi+ & \cos\psi\cos\phi+ & \cos\theta\sin\phi \\ \cos\psi\sin\theta\sin\phi & \sin\psi\sin\theta\sin\phi & \\ \sin\psi\sin\phi+ & -\cos\psi\sin\phi+ & \cos\theta\cos\phi \\ \cos\psi\sin\theta\cos\phi & \sin\psi\sin\theta\cos\phi & \end{bmatrix} \quad (39)$$

We have already defined $y_3 = y_3(T - T_f)$, $y_4 = y_4(T - T_f)$, and $y_5 = y_5(T - T_f)$ to be the Euler angles describing the missile orientation relative to an observer fixed in the frame $\mathcal{T}$. Therefore, we can write

$$\{r_m\} = [\mathbf{C}(y_3, y_4, y_5)]\{r_t\} \quad (40)$$

Finally, the frame $\mathcal{F}$ is obtained by rotating the frame $\mathcal{T}$ about its x-axis. This rotation is given by the angle $y_{5,f} = y_5(T_f)$ and we get

$$\{r_f\} = [\Phi]\{r_t\} \quad (41)$$

where

$$[\Phi] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos y_{5,f} & \sin y_{5,f} \\ 0 & -\sin y_{5,f} & \cos y_{5,f} \end{bmatrix} \quad (42)$$

The matrices [C] and [Φ] are both direction cosine matrices; therefore, each is an orthogonal matrix. Hence, the inverse of these matrices is obtained by merely transposing them.

Eqs. (40) and (41) can be combined to write

$$\{r_f\} = [\Phi][\mathbf{C}(y_3, y_4, y_5)]^T \{r_m\} \quad (43)$$

Comparing Eqs. (43) and (38), we finally get

$$[\mathbf{C}(x_{3,d}, x_{4,d}, x_{5,d})] = [\Phi][\mathbf{C}(y_3, y_4, y_5)]^T \quad (44)$$

We let the entries in $[\mathbf{C}(x_{3,d}, x_{4,d}, x_{5,d})]$ be denoted by ζ_{ij} and the entries in $[\Phi][\mathbf{C}(y_3, y_4, y_5)]^T$ by η_{ij}, where i and j are the row and column indices respectively. Equating $\zeta_{13} = \eta_{13}$, we obtain

$$-\sin x_{4,d} = \sin y_3 \sin y_5 + \cos y_3 \sin y_4 \cos y_5 \quad (45)$$

Since $-\pi/2 < x_{4,d} < \pi/2$, the above equation gives a unique value for $x_{4,d}$.

$$x_{4,d} = -\sin^{-1}(\sin y_3 \sin y_5 + \cos y_3 \sin y_4 \cos y_5) \quad (46)$$

To get an expression for $x_{3,d}$, we equate $\zeta_{11} = \eta_{11}$ and $\zeta_{12} = \eta_{12}$ and obtain

$$\cos x_{3,d} \cos x_{4,d} = \cos y_3 \cos y_4 \quad (47)$$

$$\sin x_{3,d} \cos x_{4,d} = -\sin y_3 \cos y_5 + \cos y_3 \sin y_4 \sin y_5 \quad (48)$$

$\mathbf{x}$	$\varphi(\mathbf{y})$
$x_{1,0}$	y_1
$x_{2,0}$	y_2
$x_{3,d}$	$\tan^{-1}\left(\frac{-\sin y_3 \cos y_5 + \cos y_3 \sin y_4 \sin y_5}{\cos y_3 \cos y_4}\right)$
$x_{4,d}$	$-\sin^{-1}(\sin y_3 \sin y_5 + \cos y_3 \sin y_4 \cos y_5)$

Table 1: Transformation relations between $\mathbf{x}$ and $\mathbf{y}$.

Dividing Eq. (48) by Eq. (47), we get the following expression for $x_{3,d}$ without an ambiguity in the quadrant:

$$x_{3,d} = \tan^{-1}\left(\frac{-\sin y_3 \cos y_5 + \cos y_3 \sin y_4 \sin y_5}{\cos y_3 \cos y_4}\right) \quad (49)$$

Similarly, by equating $\zeta_{23} = \eta_{23}$ and $\zeta_{33} = \eta_{33}$, we obtain

$$\cos x_{4,d} \sin x_{5,d} = \cos y_{5,f}(-\cos y_3 \sin y_5 + \sin y_3 \sin y_4 \cos y_5) + \sin y_{5,f} \cos y_4 \cos y_5 \quad (50)$$

$$\cos x_{4,d} \cos x_{5,d} = -\sin y_{5,f}(-\cos y_3 \sin y_5 + \sin y_3 \sin y_4 \cos y_5) + \cos y_{5,f} \cos y_4 \cos y_5 \quad (51)$$

Dividing Eq. (50) by Eq. (51), we get the following expression for $x_{5,d}$ without an ambiguity in the quadrant:

$$x_{5,d} = \tan^{-1}\left(\frac{(a+b)\cos y_{5,f} + c \sin y_{5,f}}{(-a+b)\sin y_{5,f} + c \cos y_{5,f}}\right) \quad (52)$$

where

$$a = \cos y_3 \sin y_5$$

$$b = \sin y_3 \sin y_4 \cos y_5$$

$$c = \cos y_4 \cos y_5$$

Table 1 summarizes the transformation relations between the boundary conditions, given in terms of the state vector $\mathbf{x}$, and the new state vector $\mathbf{y}$ at time $T - T_f$.

7. Example Trajectories

Three sets of final conditions on the state and costate variables are given in Table 2. We only vary the final roll angle $y_{5,f}$ in these three examples. Starting with these final values, the state and costate equations are integrated backwards in time. Figure 3 shows the plot of y_3 vs. y_4, where y_3 and y_4 are the yaw and pitch angles of the missile with respect to a frame whose x-axis coincides with the desired direction of the missile x_b-axis. At each integration step the state vector $\mathbf{y}$ is transformed to obtain the boundary conditions $x_{3,d}$ and $x_{4,d}$. The total angle α between the initial and the desired direction of the missile is given

	Example 1	Example 2	Example 3
$q_{1,f}$	0.25	0.25	0.25
$q_{2,f}$	−0.50	−0.50	−0.50
$q_{3,f}$	0.75	0.75	0.75
$q_{4,f}$	1.00	1.00	1.00
$y_{5,f}$	0.00	$2\pi/3$	$4\pi/3$

Table 2: Final conditions to generate example control histories.

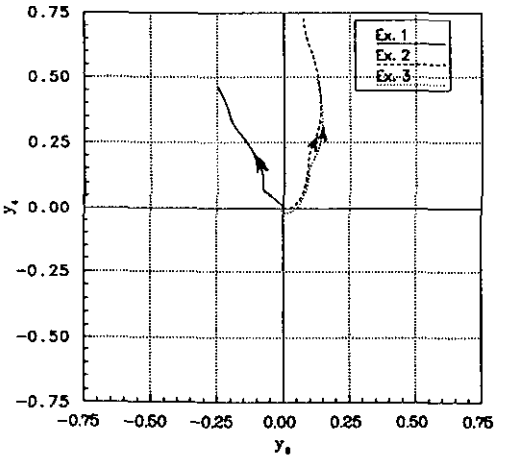

Figure 3: Path of the missile x_b-axis in the y_3-y_4 plane, where y_3 and y_4 are the yaw and pitch angles, respectively, of the x_b-axis with respect to the desired pointing direction.

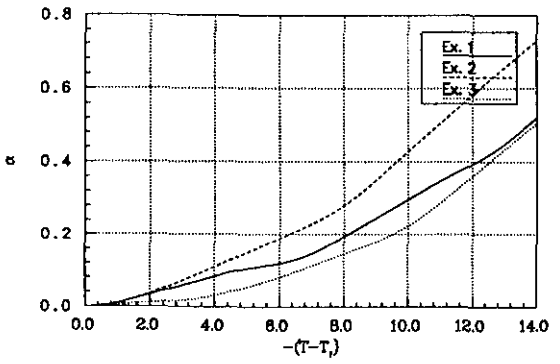

Figure 4: Total angle α vs. $-(T - T_f)$.

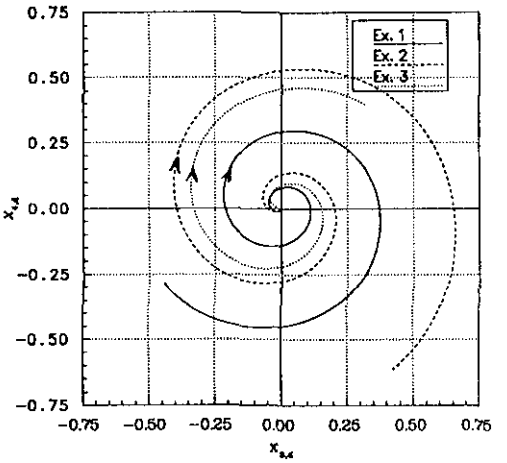

Figure 5: Path of the target in the $x_{3,d}$-$x_{4,d}$ plane, where $x_{3,d}$ and $x_{4,d}$ are the yaw and pitch angles, respectively, of the target direction with respect to the moving missile body-axis frame.

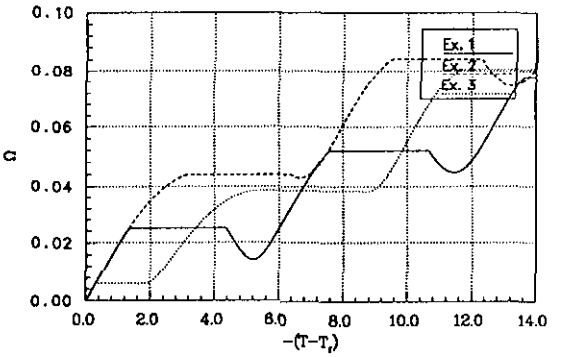

Figure 6: Total transverse angular velocity Ω vs. $-(T - T_f)$.

by $\alpha = \cos^{-1}(\cos x_{3,d} \cos x_{4,d})$. Figure 4 shows the angle α as a function of the dimensionless time T. The angles $x_{3,d}$ and $x_{4,d}$, which represent the target yaw and pitch angles, respectively, relative to the moving missile frame, are plotted in Figure 5.

The total transverse angular velocity $\Omega = \sqrt{y_1^2 + y_2^2}$ is plotted as a function of the dimensionless time T in Figure 6. We start at $\Omega = 0$ and, integrating backwards, obtain the time history of y_1 and y_2. Figure 7 shows the trajectory of transverse angular velocity components y_1-y_2 as the integration proceeds.

8. Mechanization of the Control Scheme

The thruster switch times $T_i, i = 1, 2, \ldots, n$ can be obtained from the approach given in Section 6 for the desired set of boundary conditions, where T_i is

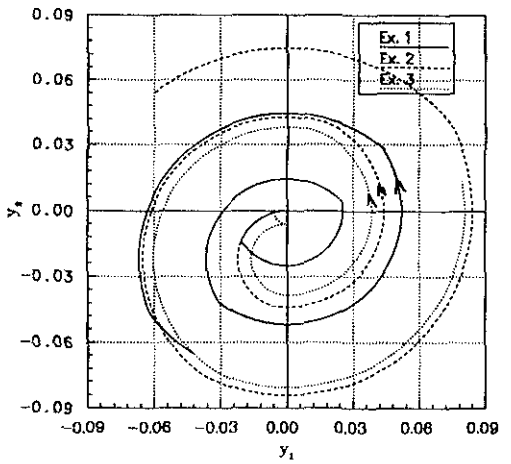

Figure 7: Trajectories of transverse angular velocity components y_1 and y_2.

the time when control switches and n is the total number of switches required to complete the attitude change maneuver. In order to implement this scheme in real-time, the switch times $T_i, i = 1, 2, \ldots, n$ must be stored in an on-board computer. We propose a control scheme which only needs to store T_1 and T_2, the first turn-on and turn-off times, respectively.

In this scheme table look-up followed by interpolation is used to compute T_1 and T_2. The thruster is then turned on from T_1 to T_2. After this first thruster firing has been completed, we can measure the state variables at T_2. The switch times T_1 and T_2 can now be recomputed based on this measured state. These new T_1 and T_2 correspond to T_3 and T_4, respectively, for the previous T_1 and T_2. Thus for the two-pulse case, the new T_3 and $T_4(= T_f)$ are such that $T_4 - T_3 = 0$. In the presence of interpolation, numerical, or measurement errors this will not be quite true. Nevertheless, in reality this scheme would probably be superior because it can correct for system and measurement errors by introducing feedback based on the latest state information.

When we store the thruster firing times at specified time intervals while integrating the state and costate equations backwards in time, a randomly spaced function is obtained. However, it is desirable for real-time function generation to have an equally spaced function. This table of equally spaced function values can be obtained by using linear interpolation across the randomly spaced function values. Once the equally spaced function to be used in real-time function generation is created, table search and interpolation can be carried out easily. Our recent research has focused on creating this table and implementing the control law using function generation. The results of this approach appear in a Ph.D. dissertation[10] and will also be published in future research papers.

Acknowledgements

The research reported in this paper has been supported by the U.S. Army Strategic Defense Command under contract number DASG60-88-C-0037 and by AFOSR under contract number F49620-86-C-0138.

References

[1] Adams, J. J., "Study of an Active Control System for a Spinning Body", NASA TN D-905, Langley Research Center, Langley Field, VA, June 1961.

[2] Athans, M. and Falb, P. L., *Optimal Control*, McGraw-Hill Inc., 1966.

[3] Cole, R. D., Ekstrand, M. E. and O'Neil, M. R., "Attitude Control of Rotating Satellites", *ARS Journal*, Vol. 31, 1961, pp. 1446-1447.

[4] Filippov, A. F., "On Certain Qustions in the Theory of Optimal Control", *SIAM J. Control*, 1962, pp. 76-84.

[5] Freed, L. E., "Attitude Control System for a Spinning Body", *National IAS-ARS Joint Meeting, Los Angeles*, Paper 61-207-1901, June 13-16, 1961.

[6] Grasshoff, L. H., "A Method for Controlling the Attitude of a Spin-Stabilized Satellite", *ARS Journal*, Vol. 31, 1961, pp. 646-649.

[7] Grubin, C., "Generalized Two-Impulse Scheme for Reorienting a spin stabilized Vehicle", *ARS Guidance, Control, and Navigation Conference, Stanford, CA*, Aug 7-9, 1961.

[8] Howe, R. M., "Attitude Control of Rockets Using a Single Axis Control Jet", *XIth International Astronautical Congress, Stockholm*, 1960.

[9] Howe, R. M., Ye, X. A. and Li, B. H., "An Improved Method for Simulation of Dynamic Systems with Discontinuous Nonlinearities", *Transactions of the Society for Computer Simulation*, Vol. 1, No. 1, Jan. 1984, pp.33-47.

[10] Jahangir, E., *Time-Optimal Attitude Control of a Spinning Missile*. Ph.D. Dissertation, University of Michigan, 1990.

[11] Jahangir, E. and Howe, R. M., "A Two-Pulse Scheme for the Time-Optimal Attitude Control of a Spinning Missile", *AIAA Guidance, Navigation and Control Conference, Portland, OR*, Aug 20-22, 1990.

[12] Patapoff, H., "Bank Angle Control System for a Spinning Satellite", AIAA Paper No. 63-339, August 1963.

[13] Porcelli, G. and Connolly, A., "Optimal Attitude Control of a Spinning Space-Body – A Graphical Approach", *IEEE Transactions on Automatic Control*, Vol. AC-12, No. 3, June 1967, pp. 241-249.

[14] Roberson, R. E., "Attitude Control of a Satellite Vehicle - An Outline of the Problems", *Proc. VIII International Astronautical Congress, Barcelona*, 1957, pp. 317-339.

[15] Wheeler, P. C., "Two-Pulse Attitude Control of an Asymmetric Spinning Satellite", *Progress in Aeronautics and Astronautics*, Vol. 13, 1964.

[16] Windenknecht, T. G., "A Simple System for Sun Orientation of a Spinning Satellite", *National IAS-ARS Joint Meeting, Los Angeles*, Paper 61-204-1898, June 13-16, 1961.