SATELLITE ATTITUDE CONTROL USING OPTIMAL CONTROL THEORY

by

Arthur Guibord

A thesis submitted to the School of Graduate Studies,
University of Ottawa, in partial fulfilment
of the requirements for the degree of
Master of Applied Science

Department of Electrical Engineering
Faculty of Science and Engineering
University of Ottawa
Ottawa, Canada

August, 1978

Approved for the Department
of Electrical Engineering

_____

Supervisor

_____

Chairman of the Examining
Committee

_____

Chairman of the Department

Abstract

The mathematical models for a zero-momentum three-axis attitude controlled satellite, employing a combination of active controls, such as flywheels, gyrotorquers and reaction jets, are derived. From these models, a set of optimal control problems are formulated.

A short survey of various numerical methods available for the solution of optimal control problems is provided. The conjugate gradient descent (CGD) method is adopted and its application, to optimal control problems having piecewise continuous input controls, is outlined.

Various aspects of satellite attitude control are investigated by solving the related optimal control problems. Finally simple sub-optimal, open-loop and feedback control policies yielding results approaching the optimal values are formulated.

Acknowledgements

The author expresses his deep gratitude to his advisor, Professor N.U. Ahmed, for his generous encouragement, understanding and guidance throughout this work, without which this thesis would not have been possible.

Special thanks are also due to Professor L.G. Birta for his helpful suggestions in the presentation of this thesis.

Special thanks to Mr. H.W. Wong for the encouragements and stimulating discussions.

Special thanks to Mrs. Jane VanDerWoude for the typing of all the equations and the proof reading of this manuscript.

Thanks to the Government of Ontario, to the National Research Council and to the University of Ottawa for financial assistance during the period of this research.

TABLE OF CONTENTS

TABLE OF CONTENTS

List of Figures

List of Figures

List of Figures

INTRODUCTION

With the ever increasing demand for high speed communication links between any two points on earth, the number of satellites in use is steadily increasing. In the field of communications, most satellites operate as microwave repeater stations, thus directional antennas are used to receive and transmit the electromagnetic signals. By using narrow beam high gain antennas, less energy has to be transmitted to maintain a given degree of quality in the communication link. This implies that the receive and transmit antennas should be accurately aimed. Furthermore, with the new advances in optical communications, it can be expected that laser beams will be used as signal carriers within a few years [27]. The extremely narrow laser beam imposes rather stringent requirements on the attitude control of satellites. An accuracy of less than 0.1 degree will be required [1]. This kind of accuracy can best be achieved using zero-momentum three-axis control employing active devices such as jets and flywheels or gyrotorquers.

At present, the attitude control of a satellite is monitored on a regular basis by tracking stations located on the earth. Discrepancies can be corrected by transmitting appropriate control signals to the satellite's attitude control mechanism in order to regain the desired orientation [4].

Since a satellite is essentially a free falling body, its dynamics are characterized by six degrees of freedom. A system of six first order coupled nonlinear ordinary differential equations describes the dynamics. The control problems arising from such dynamics are not trivial. As a simplification the dynamics are either linearized [3] or assumed uncoupled [4]. This makes it possible to apply classical control theory through Laplace transforms. A careful simulation must then be done to insure that the results are satisfactory for the actual nonlinear system.

In this research the attitude dynamics of a satellite are first derived and then used to formulate a set of optimal control problems (no linearization or uncoupling is assumed). Here the primary objective is to apply optimal control theory to study the attitude control of a satellite. Considering the complexity of the dynamics, it is not possible to obtain a closed form solution, thus numerical techniques must be used to compute the optimal controls. The next objective is then to obtain a computational scheme which is both efficient and relatively transparent. The latter implies that the resulting computer programs should necessitate a minimum of modifications to solve a set of similar optimal control problems.

By properly selecting the set of optimal control problems to be solved, it is possible to investigate certain

aspects of satellite attitude control. For example, the system's controllability when jets are used, both with throttled jets and on-off jets. Controllability under flywheel control when some of the flywheels are inoperative can also be investigated. In this study, a system is said to be controllable, if given at the initial time a non-zero state, then using the available controls the system can be driven to the origin by the terminal time. As a result of the solution of some of the above mentioned problems, it is possible to formulate open-loop and feedback control policies, giving sub-optimal results approaching the optimal values.

An outline of the thesis will now be presented. In Chapter One, the dynamics of a satellite are considered and the mathematical models are derived for the case of a flywheel on each body axis and then for a twin gyrotorquer on each body axis. In Chapter Two, some basic theorems of optimal control theory are first reviewed and then the models derived in Chapter One are reformulated into optimal control problems. Chapter Three begins with a survey of various numerical methods available for the solution of optimal control problems. A conjugate gradient descent approach is adopted and expressions are derived for the gradient of the cost function for piecewise continuous input controls. Chapter Four illustrates the computational results obtained by solving a variety of satellite optimal attitude control problems. Special emphasis is placed on non-saturating

flywheel control . In Chapter Five simple open-loop and feedback control policies for flywheel control operating in the non-saturating mode are given. Finally Conclusions are formulated.

Satellite Attitude Control Using Reaction Jets And Zero Momentum Three-Axis Control.

In the light of future satellite communications links, possibly employing optical systems, the requirement for highly accurate attitude control systems is apparent. A three-axis controller must be employed to achieve the required accuracy of less than 0.1 degree [1].

A combination of both momentum expulsion devices (e.g. jets) and momentum storage devices (e.g. reaction wheels or gyros), is often used to furnish control moments for common space vehicle attitude control systems. The jets and momentum storage devices complement one another, the storage devices countering cyclical torques on the vehicle without loss of mass and the jets countering long-term secular torques by periodically expelling momentum from the storage devices as they near spin saturation.

The typical operation of such a system may be described in the following manner: the momentum stored in the storage devices is monitored and the appropriate jets are fired when the stored momentum reaches some percentage of the total storage capacity. The design requirements for such a system are as outlined in Canon [2].

(1) The total stored gas must represent a momentum capacity larger than the total secular impulse anticipated for the life of the vehicle.

(2) The maximum storage torque capacity of the jets must be larger than the anticipated maximum torque on the vehicle, e.g.the misalignment torque during on-orbit firing.

(3) The impulse of the jet system (momentum times time) should be controllable to within a low percent of the total storage capacity of the momentum storage system, so that momentum can be expelled from the system with precision.

The above require that the momentum storage elements have sufficient capacity, such that saturation will not occur for the largest expected cyclical impulse, plus the secular momentum change between momentum expulsions. Furthermore, good precision and speed should characterize the attitude control system.

A good description of the free body dynamics of such a three-axis controlled satellite is necessary in order to present plausible control policies. In this chapter, a complete derivation of a three-axis attitude controlled satellite, employing in one case three flywheels - one on each body axis - and in the other case a twin gyrotorquer on each body axis. Both of the above systems are complemented by a set of reaction jets-one on each body axis.

It is well known that any complete derivation, relating to the dynamics of a satellite and the behaviour of flywheels and gyros within the satellite, will likely involve more than one coordinate transformation. In this derivation, the following four coordinate systems will be used: the reference, the body, the gimbal and the gyrotorquer coordinate system respectively. It is assumed that all centers of mass are located at the origin of the system. This implies that all coordinate transformations are accomplished by a rotation of coordinates without any translation, which implies that the coordinate transformation matrices are all orthogonal.

Clearly, a judicious choice of notation is necessary to give a complete yet short and concise description of the dynamics involved.

1.1 Notation.

As a convenience for the sequel of this chapter, the same notation as employed in Guibord and Ahmed [3] will be used. That is: $\Gamma_\alpha = [\bar{i}_\alpha, \bar{j}_\alpha, \bar{k}_\alpha]$, ($\alpha$=g,G,b,r for gyro, gimbal, body and reference coordinate system respectively) and A' denotes transpose of A. $\Gamma_\nu = c_\nu^\beta \Gamma_\beta : \nu, \beta \in \{g,G,b,r\}$, where c_ν^β is a (3x3) orthogonal transformation matrix and therefore $[c_\nu^\beta] = [c_\nu^\beta]^{-1} = c_\beta^\nu$. The trigonometric functions sine and cosine are abbreviated by the letters s and c respectively. The quantity $\dot{q}$ means

the time derivative of q. The subscripts x,y,z will denote the x,y and z body axes respectively. A square bracket followed by an index indicates that the quantity, within the brackets, is expressed in the indexed coordinate system. The quantities capped by ($\bar{}$) such as $\bar{A}$, are tensors of rank one or simply vectors while quantities capped by ($\bar{\bar{}}$) are tensors of rank two.

The following nomenclature is employed:

$\bar{H}$:total angular momentum of satellite and gyroscope.

$\bar{H}_b$:angular momentum of satellite body.

$\bar{H}_g$:angular momertum of gyroscope.

$\bar{\bar{I}}$:inertia dyadic of satellite body.

$\bar{\bar{J}}$:inertia dyadic of gyroscope rotor.

$\bar{\omega}_g$:angular velocity of gyrotorquer coordinate frame.

$\bar{\omega}_G$:angular velocity of gimbal coordinate frame.

$\bar{\omega}_b$:angular velocity of satellite body coordinate frame.

$\bar{\omega}_r$:angular velocity of reference coordinate frame.

p,q,r:satellite body rates with respect to body coordinate frame.

ϕ , θ, ψ:satellite body angles with respect to reference coordinate frame.

Ω :gyrotorquer flywheel angular velocities.

δ :gyrotorquer flywheel gimballing angle.

ε :gyrotorquer flywheel position angle.

A :nonion form of flywheel inertia tensor.

B :nonion form of satellite body inertia tensor.

To transform a quantity given in one of the four mentioned coordinate systems requires the knowledge of three basic transformation matrices. These will be taken to be c_g^G, c_G^b, and c_b^r which are given by;

$$c_g^G = \begin{bmatrix} c\varepsilon & s\varepsilon & o \\ -s\varepsilon & c\varepsilon & o \\ o & o & 1 \end{bmatrix} \tag{1.1.1}$$

$$c_G^b = \begin{bmatrix} 1 & o & o \\ o & s\delta & -c\delta \\ o & c\delta & s\delta \end{bmatrix} \tag{1.1.2}$$

and:

$$c_b^r = \begin{bmatrix} c\theta c\psi & c\theta s\psi & -s\theta \\ s\phi s\theta c\psi - c\phi s\psi & s\phi s\theta s\psi + c\phi c\psi & s\phi c\theta \\ c\phi s\theta c\psi + s\phi s\psi & c\phi s\theta s\psi - s\phi c\psi & c\phi c\theta \end{bmatrix} \tag{1.1.3}$$

Note that $c_g^r = c_g^G c_G^b c_b^r$ and $c_r^g = c_r^b c_b^G c_G^g$ etc. The matrices c_G^b and c_b^r are derived in Greensite [4]. The matrix c_G^b corresponds to the configuration for a single degree of freedom gyro [5,6] as shown in Fig.1.1.1.

To obtain the c_b^r transformation, the sequence of rotations shown in Fig.1.1.2 should be followed. Note that the Euler angles ϕ, θ and ψ denote the relative orientation of

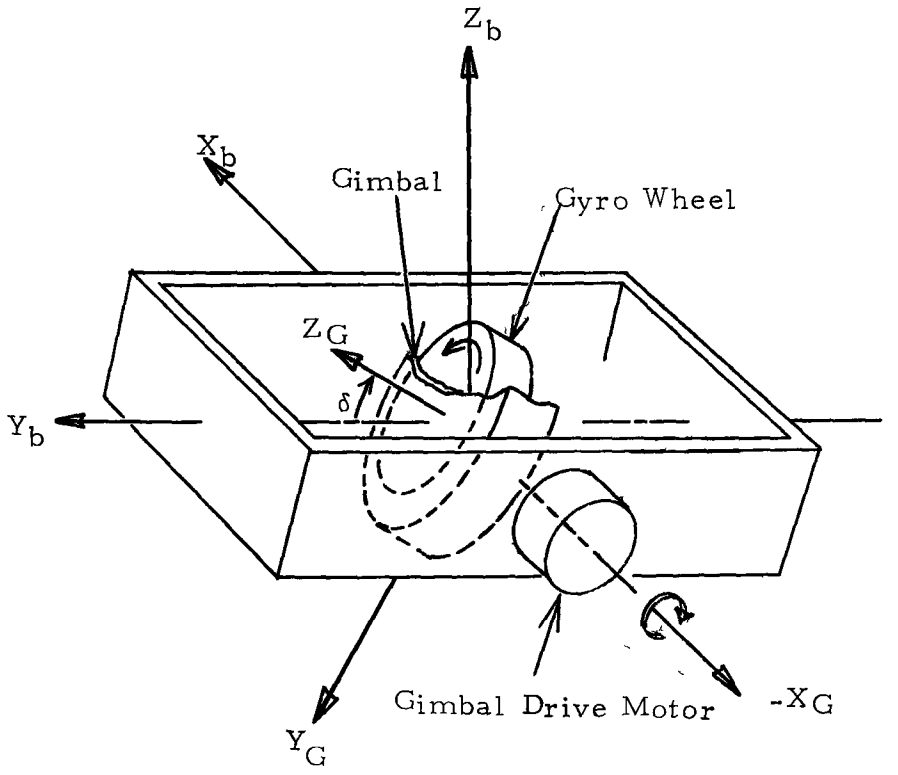


Fig. 1.1.1 Single Flywheel Gyrotorquer Controller.

$$\bar{i}_r \longrightarrow \bar{i}_1 \longrightarrow \bar{i}_b \xrightarrow{\phi} \bar{i}_b$$

$$\bar{j}_r \longrightarrow \bar{j}_1 \xrightarrow{\theta} \bar{j}_1 \longrightarrow \bar{j}_b$$

$$\bar{k}_r \xrightarrow{\psi} \bar{k}_1 \longrightarrow \bar{k}_2 \longrightarrow \bar{k}_b$$

Fig. 1.1.2 Coordinate Rotation Procedure.
the two systems.

In other words, the reference axis system is rotated about the $\bar{k}_r$ axis by a positive* amount, ψ, thereby defining a new axis system $(\bar{i}_1, \bar{j}_1, \bar{k}_r)$. This is followed by a posi-

tive rotation about axis $\bar{j}_1$ of an amount θ and finally by a positive rotation of amount ϕ about axis $\bar{i}_b$. At the end of this rotation sequence, the reference frame is coincident with the body frame. The matrix c_g^G is obtained directly

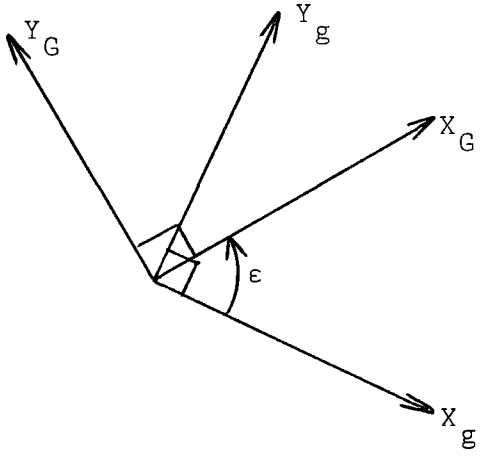


Fig. 1.1.3 Gimbal and Gyro Coordinate Vectors.

from Fig.1.1.3.

The quantities $\bar{\omega}_g$, $\bar{\omega}_G$, $\bar{\omega}_b$ and $\bar{\omega}_r$ are implicitly defined in their respective coordinate systems**.

$$\bar{\omega}_g \equiv \left[\bar{\omega}_g\right]_g = \Omega_y \bar{k}_g = (0,0,\Omega_y)\Gamma_g \tag{1.1.4}$$

$$\bar{\omega}_G \equiv \left[\bar{\omega}_G\right]_G = \dot{\delta}_y \bar{i}_G = (\dot{\delta}_y,0,0)\Gamma_G \tag{1.1.5}$$

$$\bar{\omega}_b \equiv \left[\bar{\omega}_b\right]_b = p\bar{i}_b + q\bar{j}_b + r\bar{k}_b = (p,q,r)\Gamma_b \tag{1.1.6}$$

$$\bar{\omega}_r \equiv \left[\bar{\omega}_r\right]_r = -\omega_0 \bar{j}_r = (0,-\omega_0,0)\Gamma_r \tag{1.1.7}$$

---

*In the sense of the right hand rule.
**This derivation is initiated with the y-body axis.

The same applies to the tensors $\overline{\overline{T}}$ and $\overline{\overline{T}}$. i.e. $\overline{\overline{I}} \equiv [\overline{\overline{I}}]_b = \Gamma_b^{'} B \Gamma_b$

where;

$$B = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \qquad (1.1.8)$$

and $\overline{\overline{J}} \equiv [\overline{\overline{J}}]_G = \Gamma_G^{'} A \Gamma_G$, where:

$$A = \begin{bmatrix} A_y & 0 & 0 \\ 0 & A_y & 0 \\ 0 & 0 & C_y \end{bmatrix} \qquad (1.1.9)$$

1.2 Basic Dynamics.

Proceeding in a manner similar to Greensite [4], the dynamics are derived for a zero-momentum three-axis satellite attitude control system for two special cases. The first case is for one flywheel and the second is for a twin gyrotorquer controller on each of the satellite's body axes respectively.

From Newtonian mechanics, it is known that the rate of change of angular momentum ($\overline{H}$) in a free body is equal to

the applied external torques ($\overline{T}$).

$$\text{i.e.} \quad \frac{d\overline{H}}{dt} = \overline{T} \tag{1.2.1}$$

In the case of a satellite, the external torques can basically be separated into two parts $\overline{T}_d$ disturbance torques and $\overline{T}_c$ controlling torques. The disturbance torques may be characterized by both impulses and sinusoids, with the latter predominating for a vehicle in a circular orbit* about a planet. Impulsive disturbances may occur through meteorite impact. The external controlling torques are applied through a set of reaction jets symmetrically positioned on each body axis.

From (1.2.1) it can be seen that if no external torques act on the satellite, the dynamics must satisfy:

$$\frac{d\overline{H}}{dt} = 0 \tag{1.2.2}$$

which implies that the total angular momentum must be conserved. In such a case, attitude control is readily achieved through momentum exchange devices.

For the purpose of this derivation let:

$$\overline{T} = T_x\overline{i}_b + T_y\overline{j}_b + T_z\overline{k}_b \tag{1.2.3}$$

---

*In this research a circular orbit has been assumed.

and,

$$\frac{d\bar{H}}{dt} = \dot{H}_x \bar{i}_b + \dot{H}_y \bar{j}_b + \dot{H}_z \bar{k}_b \tag{1.2.4}$$

from which it follows that;

$$\dot{H}_x = T_x, \quad \dot{H}_y = T_y, \quad \dot{H}_z = T_z \tag{1.2.5}$$

which is the basic set of equations necessary to formulate and solve satellite attitude control problems.

The computation of $d\bar{H}/dt$ is somewhat complicated by the fact that rotating coordinate frames are involved. The following formula given by Goldstein [4] must be employed:

$$\left(\frac{d.}{dt}\right)_{space} = \left(\frac{d.}{dt}\right)_{body} + \bar{\omega}_{body} \times (.) \tag{1.2.6}$$

It was shown in [3] that in the case of a satellite the coordinate system to be identified as 'body' in (1.2.6) is arbitrary. Let us now proceed with the derivation of $d\bar{H}/dt$, where $\bar{H} = \bar{H}_b + \bar{H}_g$.

For the purpose of derivation, differentiation is done with respect to the satellite body coordinates. Hence the quantities $\bar{H}_b$ and $\bar{H}_g$ should be expressed in the body coordinate system. Proceeding with $\bar{H}_b$.

$$\left[\bar{H}_b\right]_b = \left[\bar{\bar{I}}\right]_b \cdot \left[\bar{\omega}_b + \bar{\omega}_r\right]_b \tag{1.2.7}$$

which becomes;

$$\left[\bar{H}_b\right]_b = \left[(p,q,r) + (0,-\omega_0,0)c_r^b\right]B\Gamma_b \tag{1.2.8}$$

after substitution using (1.1.6), (1.1.7) and (1.1.8). Similarly;

$$\left[\bar{H}_g\right]_b = \left[\bar{\bar{J}}\right]_b \cdot\left[\bar{\omega}_g + \bar{\omega}_G + \bar{\omega}_b + \bar{\omega}_r\right]_b \tag{1.2.9}$$

becomes;

$$\left[\bar{H}_g\right]_b = \left[(0,0,\Omega_y)c_g^b + (\delta_y,0,0)c_G^b + (p,q,r) + (0,-\omega_0,0)c_r^b\right]c_b^G Ac_G^b\Gamma_b \tag{1.2.10}$$

after substitution using (1.1.4), (1.1.5), (1.1.6), (1.1.7) and (1.1.9). Hence, adding (1.2.8) and (1.2.10) gives:

$$\left[\bar{H}\right]_b = \left[(0,0,\Omega_y)c_g^G Ac_G^g + (\dot{\delta}_y,0,0)Ac_G^b + (p,q,r)(B+c_b^G Ac_G^b) + (0,-\omega_0,0)(c_r^b B+c_r^b c_b^G Ac_G^b)\right]\Gamma_b \tag{1.2.11}$$

The above can be simplified considerably since $B>>A$ which implies $B>>c_b^G Ac_G^b$. Therefore;

$$\left[\bar{H}\right]_b = \left[(0,0,\Omega_y)c_g^G Ac_G^b + (\dot{\delta}_y,0,0)Ac_G^b + (p,q,r)B + (o,-\omega_0,0)c_r^b B\right]\Gamma_b \tag{1.2.12}$$

or more simply:

$$\left[\bar{H}\right]_b = \left[\bar{\bar{J}}\right]_b \cdot\left[\bar{\omega}_g+\bar{\omega}_G\right]_b + \left[\bar{\bar{I}}\right]_b \cdot\left[\bar{\omega}_b+\bar{\omega}_r\right]_b \tag{1.2.13}$$

The time derivative of the first term in (1.2.13) was completely derived in [3] and was found to be;

$$\frac{d}{dt}\left(\left[\bar{\bar{J}}\right]_b \cdot\left[\bar{\omega}_g+\bar{\omega}_G\right]_b\right)= \left[A_y\ddot{\delta}_y+C_y\Omega_y(q-\omega_0 m_2)s\delta_y - C_y\Omega_y (r-\omega_0 n_2)c\delta_y\right]\bar{i}_b + \left[A_y\dot{\delta}_y(r-\omega_0 n_2) - C_y\Omega_y(\dot{\delta}_y+p-\omega_0 l_2)s\delta_y + C_y\dot{\Omega}_y c\delta_y\right]\bar{j}_b + \left[C_y\Omega_y(\dot{\delta}_y+p-\omega_0 l_2)c\delta_y - A_y(q-\omega_0 m_2)\dot{\delta}_y + C_y\Omega_y s\delta_y\right]\bar{k}_b \tag{1.2.14}$$

where:

$$l_2 = c\theta s\psi$$
$$m_2 = s\phi s\theta s\psi + c\phi c\psi \tag{1.2.15}$$
$$n_2 = c\phi s\theta s\psi - s\phi c\psi$$

To compute the second term in (1.2.13), let:

$$\left[\frac{d\bar{H}_b}{dt}\right]_b = \frac{d\left[\bar{H}_b\right]_b}{dt} + \left[\bar{\omega}_b + \bar{\omega}_r\right]_b \times \left[\bar{H}_b\right]_b \tag{1.2.16}$$

Thus from (1.2.8);

$$\frac{d\left[\bar{H}_b\right]_b}{dt} = \left[(\dot{p},\dot{q},\dot{r}) + (0,-\omega_0,0)\dot{c}^b_r\right]B\Gamma_b \tag{1.2.17}$$

and:

$$\left[\bar{\omega}_b+\bar{\omega}_r\right]_b \times \left[\bar{H}_b\right]_b = \left[(q-\omega_0 m_2)(r-\omega_0 n_2)(I_z-I_y)\right]\bar{i}_b + \left[(p-\omega_0 l_2)(q-\omega_0 m_2)(I_x-I_z)\right]\bar{j}_b + \left[(p-\omega_0 l_2)(q-\omega_0 m_2)(I_y-I_x)\right]\bar{k}_b \tag{1.2.18}$$

Adding (1.2.14), (1.2.17) and (1,2,18) yields:

$$\left[\frac{d\bar{H}}{dt}\right]_b = \left[(\dot{p}-\omega_0\dot{l}_2)I_x + (q-\omega_0 m_2)(r-\omega_0 n_2)(I_z-I_y) + A_y\ddot{\delta}_y + C_y\Omega_y(q-\omega_0 m_2)s\delta_y - C_y\Omega_y(r-\omega_0 n_2)c\delta_y\right]\bar{i}_b + \left[(\dot{q}-\omega_0\dot{m}_2)I_y + (p-\omega_0 l_2)(r-\omega_0 n_2)(I_x-I_z) + A_y\dot{\delta}_y(r-\omega_0 n_2) - C_y\Omega_y(\dot{\delta}_y+p-\omega_0 l_2)s\delta_y + C_y\dot{\Omega}_y c\delta_y\right]\bar{j}_b + \left[C_y\Omega_y(\dot{\delta}_y+p-\omega_0 l_2)c\delta_y - A_y(q-\omega_0 m_2)\dot{\delta}_y +\right.$$

$$+ C_y\dot{\Omega}_y s\delta_y + (\dot{r}-\omega_0\dot{n}_2)I_y + (p-\omega_0 l_2)(q-\omega_0 m_2)(I_y-I_x)\Big]\bar{k}_b$$

(1.2.19)

Now (1.2.19) can be simplified considerably by assuming that ω_0, ϕ, θ, and ψ are small. From (1.2.15) it follows that:

$$\begin{aligned} \omega_0 l_2 &\longrightarrow 0 \\ \omega_0 m_2 &\longrightarrow \omega_0 \\ \omega_0 n_2 &\longrightarrow 0 \end{aligned} \qquad (1.2.20)$$

Furthermore, it was shown in Greensite [4] that for the transformation matrix (1.1.3), the Euler angles ϕ, θ and ψ are related by:

$$\begin{aligned} \dot{\phi} &= p + (qs\phi + rc\phi)\tan\theta \\ \dot{\theta} &= qc\phi - rs\phi \\ \dot{\psi} &= (qs\phi + rc\phi)\sec\theta \end{aligned} \qquad (1.2.21)$$

It then follows that for small ω_0, ϕ, θ and ψ that:

$$\begin{aligned} \omega_0\dot{l}_2 &\longrightarrow \omega_0 r \\ \omega_0\dot{m}_2 &\longrightarrow 0 \\ \omega_0\dot{n}_2 &\longrightarrow -\omega_0 p \end{aligned} \qquad (1.2.22)$$

Hence (1.2.19) becomes:

$$\left[\frac{d\bar{H}}{dt}\right]_b = \Big[(\dot{p}-\omega_0 r)I_x + (q-\omega_0)r(I_z-I_y) + A_y\ddot{\delta}_y + C_y\Omega_y(q-\omega_0)s\delta_y$$

$$- C_y\Omega_y rc\delta_y\Big]\bar{i}_b + \Big[\dot{q}I_y + pr(I_x-I_z) + A_y\dot{\delta}_y r - C_y\Omega_y(\dot{\delta}_y+p)s\delta_y$$

$$+ C_y\dot{\Omega}_y c\delta_y\Big]\bar{j}_b + \Big[(\dot{r}+\omega_0 p)I_z + (q-\omega_0)p(I_y-I_x) - A_y(q-\omega_0)\dot{\delta}_y$$

$$+ C_y\Omega_y(\dot{\delta}_y+p)c\delta_y + C_y\dot{\Omega}_y s\delta_y\Big]\bar{k}_b \qquad (1.2.23)$$

Expressing the above as outlined in (1.2.5) gives;

$$(\dot{p}-\omega_0 r)I_x + (q-\omega_0)r(I_z-I_y) + A_y\ddot{\delta}_y + C_y\Omega_y(q-\omega_0)s\delta_y - C_y\Omega_y rc\delta_y = T_x$$

$$\dot{q}I_y + pr(I_x-I_z) + A_y\dot{\delta}_y r + C_y\dot{\Omega}_y c\delta_y - C_y\Omega_y(\dot{\delta}_y+p)s\delta_y = T_y$$

$$(\dot{r}+\omega_0 p)I_z + (q-\omega_0)p(I_y-I_x) - A_y(q-\omega_0)\dot{\delta}_y + C_y\dot{\Omega}_y s\delta_y + C_y\Omega_y(\dot{\delta}_y+p)c\delta_y = T_z \quad (1.2.24)$$

which corresponds to the dynamics for one flywheel on the y-body axis. To include the dynamics for one flywheel on each body axis, the principle of superposition can easily be applied because of the symmetry involved. Furthermore, (1.2.24) can be specialized to one or the other of the special cases of attitude control mentioned earlier, which leads to simpler expressions.

1.3 Flywheel Control.

Let us consider the case of flywheel control, with one flywheel controller on each body axis, for which it is assumed that δ_y, $\dot{\delta}_y$ and $\ddot{\delta}_y$ are all zero. Hence all expressions containing δ_y, $\dot{\delta}_y$, $\ddot{\delta}_y$ or $s\delta_y$ in (1.2.24) vanish. The extra terms introduced by having one flywheel on the x and

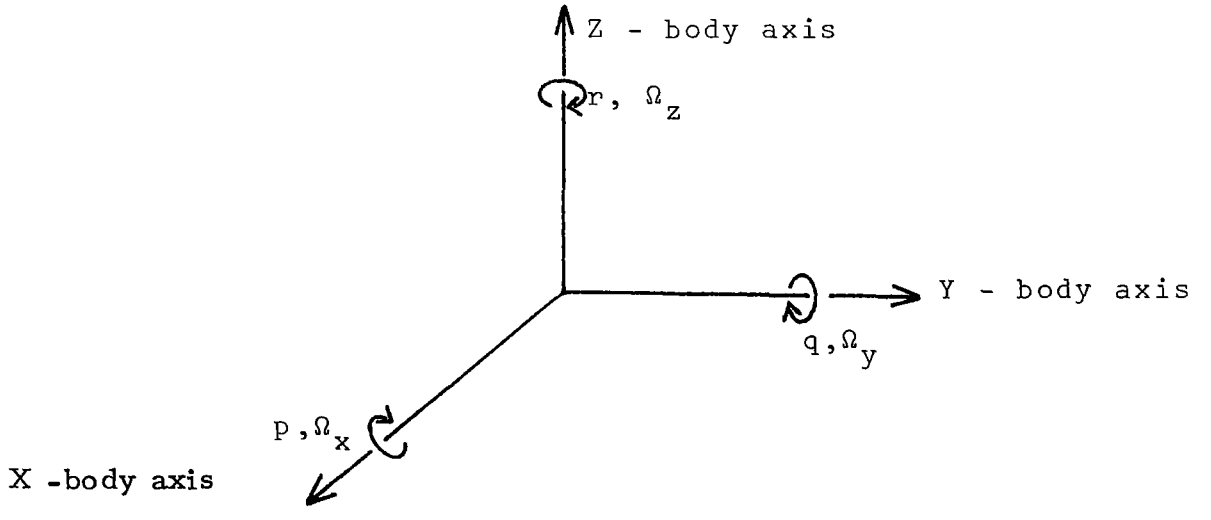


Fig. 1.3.1 Flywheel Angular Velocities with respect to Body Coordinates.

z body axes respectively may be visualized through Fig.1.3.1. Thus for one flywheel controller on each body axis (1.2.24) becomes:

$$
\begin{aligned}
(\dot{p}-\omega_0 r)I_x + (q-\omega_0)r(I_z-I_y) &= -C_x\dot{\Omega}_x + C_y\Omega_y r - C_z\Omega_z q + T_x \\
\dot{q}I_y + pr(I_x - I_z) &= -C_x\Omega_x r - C_y\dot{\Omega}_y + C_z\Omega_z p + T_y \qquad (1.3.1) \\
(\dot{r}+\omega_0 p)I_z + (q-\omega_0)p(I_y-I_x) &= C_x\Omega_x q - C_y\Omega_y p - C_z\dot{\Omega}_z + T_z
\end{aligned}
$$

1.4 Gyrotorquer Control.

For the case of a twin gyrotorquer controller on each body axis, symmetry and superposition may again be used

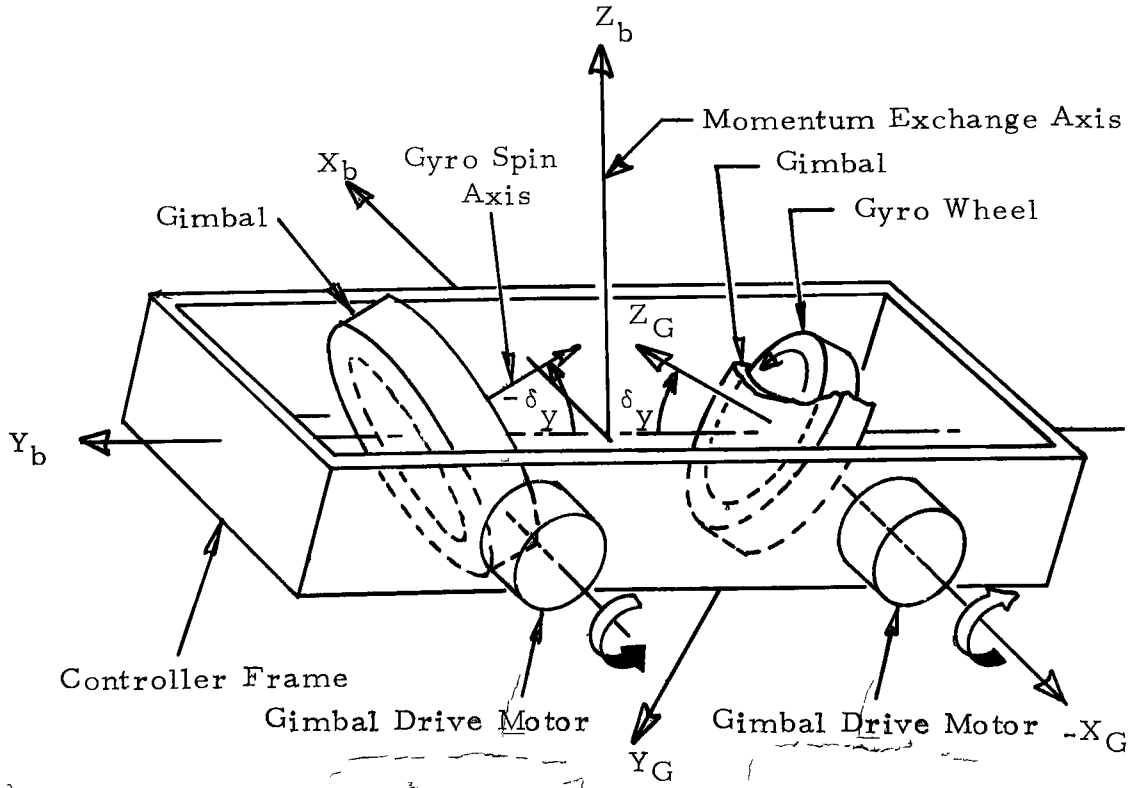


Fig. 1.4.1 Twin Flywheel Gyrotorquer Controller.

Fig.1.4.1 illustrates a twin gyrotorquer controller. In the

twin gyrotorquer, the angular velocity Ω of both flywheels is held constant, at the same value, but in the opposite directions. Furthermore, because of the geometry involved, all terms describing the rate of change of angular momentum for the upper flywheel have the quantities δ, $\dot{\delta}$, $\ddot{\delta}$, Ω and $\dot{\Omega}$ of opposite sign of those for the same quantities in the lower flywheel. Hence, for a twin gyrotorquer controller on each body axis, the dynamics are expressed by:

$$(\dot{p}-\omega_0 r)I_x + (q-\omega_0)r(I_z-I_y) = 2C_x\Omega_x rs\delta_x - 2C_y\Omega_y(q-\omega_0)s\delta_y - 2C_z\Omega_z\dot{\delta}_z c\delta_z + T_x$$

$$\dot{q}I_y + pr(I_x-I_z) = -2C_x\Omega_x\dot{\delta}_y + 2C_y\Omega_y ps\delta_y - 2C_z\Omega_z(r-\omega_0)s\delta_z + T_y$$

$$(\dot{r}+\omega_0 p)I_z + (q-\omega_0)p\,(I_y-I_x) = -2C_x\Omega_x(p-\omega_0)s\delta_x - 2C_y\Omega_y\dot{\delta}_y c\delta_y + 2C_z\Omega_z qs\delta_z + T_z \qquad (1.4.1)$$

Summary.

In this chapter, the dynamics for two special cases of three-axis zero- momentum attitude control for a satellite have been derived. Equations (1.2.21), (1.3.1) and (1.4.1) will serve as the basis in the formulation of satellite optimal attitude control problems in the next chapter.

CHAPTER TWO

Optimal Control And Formulation Of Optimal Attitude Control Problems.

In the past, satellite attitude control problems have been solved using classical control techniques and computer simulations [7,8]. The basic approach was to assume that no coupling existed between the different axes. The uncoupled equations were then linearized for analysis by Laplace transforms. This method yields sub-optimal results. In this study no assumption is made on the coupling of the dynamics. An appropriate cost function subject to the complete dynamics will then be introduced. The minimization of this cost function for some set of admissible controls should yield improved control policies over the previous technique.

The solution of an optimal control problem presents certain difficulties such as the necessity to solve a large dimensional two-point boundary value problem (TPBVP) or its equivalent. Furthermore, the solution of the optimal control problems, by such methods as the well known Pontryagin Maximum Principle (PMP) [9], do not immediately lead to a simple optimal controller design.

The intent of this research is to study certain aspects of satellite attitude control through the solution of a set

of related optimal control problems. As a result of this study, it may be possible to formulate simple open-loop and feedback control policies that give results approaching the optimal solution.

Before proceeding to the formulation of satellite optimal attitude control (SOAC) problems, some of the basic theorems of optimal control theory will first be outlined. This will provide an idea of the type of problems for which solutions exist and perhaps some apriori knowledge of the nature of the solution that can be anticipated.

The theorems stated in the sequel of this chapter were taken from Pontryagin et al. [9]. Proofs of the theorems are also given in [9].

2.1 Pontryagin's Maximum Principle.

Consider the system S;

$$\left.\begin{aligned} &\dot{x} = f(x,u), \quad x(t_0) = c_0 \\ &\text{where:} \\ &x \equiv (x_1(t),\ x_2(t),\ \ldots,\ x_n(t))' \\ &u \equiv (u_1(t),\ u_2(t),\ \ldots,\ u_m(t))' \\ &f(x,u) \equiv (f_1(x,u), f_2(x,u),\ \ldots,\ f_n(x,u))' \end{aligned}\right\} \quad (s)$$

The vectors x and u represent the states and controls respectively. The functions $f_i(x,u)$ are defined by;

$$f_i(x,u): X \times U \longrightarrow Y, \quad i=1,2,\ \ldots,n \qquad (2.1.1)$$

where $x \in X \subset R^n$, $u \in U \subset E^r$ and $Y \subset R^n$ have continuous partial derivatives in XxU. U is the class of admissible controls defined by:

$$U = \left\{ \Omega : [t_0, T] \longrightarrow E^r \;\middle|\; \Omega \text{ is bounded and measurable} \right\} \tag{2.1.2}$$

Let the functional to be minimized for some $u^* \in U$ have the form:

$$J(u) = \int_{t_0}^{T} f_0(x,u)dt \tag{2.1.3}$$

By defining a new state variable;

$$x_0(t) = \int_{t_0}^{t} f_0(x(s),u(s))ds \tag{2.1.4}$$

and adjoining it to the state vector x, the dimension of the state space X is increased from n to (n+1) and is denoted by $\hat{X}$. Hence, the new system S' is represented by;

$$\left.\begin{aligned} \dot{\hat{x}} &= \hat{f}(x,u),\ \hat{x}(t_0) = \hat{c}_0 \\ \hat{x} &\equiv (x_0(t), x_1(t), \ldots, x_n(t))' \\ u &\equiv (u_1(t), u_2(t), \ldots, u_m(t))' \\ \hat{f}(\hat{x},u) &\equiv (f_0(x,u), f_1(x,u), \ldots, f_n(x,u))' \end{aligned}\right\} \quad (s')$$

where:

Concise Problem Statement: The problem described herein may be stated more concisely as:

Given $x = (0, c_0) \in \hat{X}$ and the line l parallel to the x_0 axis passing through the point $(0, c_1)$; find the controls $u = u(t) \in U$ such that the solution of S' intersects l at the coordinate of least value for $x_0(T)$. This problem is referred to as the fundamental problem.

Hamiltonian Notation: To state the theorems of Pontryagin et al. [9] in a manner as concise as possible, the Hamiltonian function H is introduced;

$$H = \left\langle \xi, \hat{f}(x,u) \right\rangle = \sum_{i=0}^{n} \xi_i f_i(x,u) \tag{2.1.5}$$

where $\xi_i, i=0, 1, \ldots, n$ are referred to as the co-state variables and satisfy the differential equations:

$$\dot{\xi}_i = - \sum_{j=0}^{n} \frac{\partial f_j(x,u)\xi_j}{\partial x_i}, \quad i=0,1, \ldots,n \tag{2.1.6}$$

Using the Hamiltonian function, the state and co-state differential equations may be expressed by the Hamiltonian system:

$$\dot{x}_i = \frac{\partial H}{\partial \xi_i}, \quad i=0,1, \ldots,n \tag{2.1.7}$$

$$\dot{\xi}_i = \frac{-\partial H}{\partial x_i}, \quad i=0,1, \ldots,n \tag{2.1.8}$$

Theorem 2.1.1 (Pontryagin's Maximum Principle): Let $u(t)$, $t \in [t_0, T]$, be an admissible control such that the corresponding trajectory $\hat{x}(t)$ (see S), which begins at the point $\hat{c}_0$ at the time $t=t_0$, is defined on the interval $t \in [t_0, T]$ and passes at the time T through a point on the line l. In order that $u(t)$ and $\hat{x}(t)$ be optimal, it is necessary that there exist a non-zero absolutely continuous vector function $\xi = (\xi_0(t), \xi_1(t), \ldots, \xi_n(t))'$, corresponding to the functions $u(t)$

and $\hat{x}(t)$ such that:

(1) The function $H(\xi,x,u)$ of the variable $u\varepsilon U$ attains its maximum at the point $u=u(t)$, almost everywhere for $t\varepsilon[t_0,T]$;

$$\text{i.e.}H(\xi,x,u) = \max_{v\varepsilon U} H(\xi,x,v) = M(\xi,x) \tag{2.1.9}$$

(2) At the time T, the relations;

$$\xi(T)\leq 0, \quad M(\xi(T),x(T)) = 0 \tag{2.1.10}$$

are satisfied. Furthermore, it turns out that if ξ, $\hat{x}$ and u satisfy (2.1.7), (2.1.8), and condition (1), the time functions $\xi_0(t)$ and $M(x,\xi)$ are constant. Thus (2.1.10) holds at any time $t\varepsilon[t_0,T]$ and not just at T.

For the time optimal case, $f_0(x,u)=1$ making the Hamiltonian function;

$$H = \xi_0 + \sum_{i=1}^{n} f_i(x,u)\xi_i \tag{2.1.11}$$

where ξ becomes an n dimensional vector because $\xi_0(t)$ is a constant and the Hamiltonian system is given by:

$$\dot{x}_i = \frac{\partial H}{\partial \xi_i}, \quad i=1,2, \ldots,n \tag{2.1.12}$$

$$\dot{\xi}_i = \frac{-\partial H}{\partial x_i}, \quad i=1,2, \ldots,n \tag{2.1.13}$$

Theorem 2.1.2 PMP for the Time Optimal Case [9]: Let u(t),

$t\in[t_0,T]$, be an admissible control which transfers the state point from c_0 to c_1 and let $x(t)$ be the corresponding trajectory (see (2.1.12)) so that $x(t_0)=c_0$, $x(T)=c_1$. In order that there exist a non-zero continuous function $\xi=(\xi_1(t),\xi_2(t),\ldots,\xi_n(t))'$ corresponding to $u(t)$ and $x(t)$ (see (2.1.13)) such that:

(1) For all $t\in[t_0,T]$, the function $H(\xi,x,u)$ of the variable $u\in U$ attains its maximum at the point $u=u(t)$;

$$\text{i.e. } H(\xi,x,u) = M(\xi,x) \tag{2.1.14}$$

(2) At the terminal time T, the relation;

$$M(\xi(T),\ x(T)) = 0 \tag{2.1.15}$$

is satisfied. Furthermore, it turns out that if ξ, x and u satisfy (2.1.12) and (2.1.13), the time function $M(\xi(t),x(t))$ is constant. Thus (2.1.15) may be verified for any time $t\in[t_0,T]$ and not just at t=T.

Since the optimal control problems to be formulated in this chapter arise from the attitude control of a satellite, it is generally too stringent to prescribe a fixed point to which the state vector is to be driven. Rather, it is more practical to require that certain states be driven to a given neighborhood, perhaps a small sphere around the origin. Such a requirement may be reflected in the cost functional to be minimized, by a terminal cost;

$$\text{i.e. } J(u) = \Theta(x(T)) + \int_{t_0}^{T} f_0(x,u)dt \tag{2.1.16}$$

with typically*;

$$\Theta(x(T)) = \tfrac{1}{2}x'R\mathbf{x}\Big|_{t=T} \qquad (2.1.17)$$

where R is a positive semi-definite matrix.

The two theorems of Pontryagin et al. stated earlier are equally valid in this case. The difficulty here is that the Hamiltonian generated by this problem has (n+1) initial conditions and no terminal conditions for a system of 2(n+1) first order differential equations. Another set of (n+1) end-point conditions must be specified to permit solution of the Hamiltonian system. At this point the notion of the transversality conditions is introduced.

Transversality Conditions: Let Ψ be a smooth manifold in X of arbitrary (but less than n) dimension m but sufficiently large so that the optimal trajectory $x(t)\varepsilon\Psi$. The optimal control problem can now be stated as:
Find a $u(t)\varepsilon U$ that minimizes (2.1.16) subject to S.

The transversality conditions are formulated as follows: Let $x(T)\varepsilon\Psi$ and T_1, the plane tangent to Ψ passing through $x(T)$, have dimension m. Furthermore, let $u(t), \hat{x}(t)$, $t\varepsilon[t_0,T]$ be the solution of the optimal control problem with

---

*The 1/2 is a mathematical convenience in later considerations.

fixed end-points c_0 and c_1. Finally, let $\xi(t)$ be a vector satisfying theorem 2.1.1. The vector $\xi(t)$ satisfies the transversality conditions at the right-hand end-point of the trajectory $\hat{x}(t)$ (i.e. at $\hat{x}(T)$) if the vector is orthogonal to Ψ. The following theorem by Pontryagin et al. results.

Theorem 2.1.3 PMP with Free End-Point: Let $u(t)$, $t\varepsilon[t_0,T]$, be an admissible control which transfers the state from some point c_0 and let $\hat{x}(t)$ be the corresponding trajectory starting at the point $\hat{c}_0=(0,c_0)'$. In order that $u(t)$ and $\hat{x}(t)$ yield the solution of the optimal control problem with a free end-point, it is necessary that there exist a non-zero continuous vector $\xi(t)$ which satisfies the conditions of theorem 2.1.1, and in addition the transversality conditions at the end-point.

In the time-optimal case, replace $\hat{x}(t)$ by $x(t)$ in theorem 2.1.3 and the reference to theorem 2.1.1 is replaced by a reference to theorem 2.1.2.

2.2 Optimal Attitude Control Problems.

A set of optimal attitude control problems that can be solved by applying Pontryagin's theorems will now be formulated.

Many approaches are available for the numerical solution

of the resulting TPBVP's. The details of various numerical techniques for the solution of optimal control problems are discussed in the next chapter. Pros and cons for each method are assessed and justification is given to the computational method adopted in this thesis.

Optimal Flywheel Attitude Control (OFAC): Equations (1.2.21) and (1.3.1) will serve as a basis in the formulation of OFAC problems. They are:

$$\begin{aligned} \dot{\phi} &= p + (qs\phi + rc\phi)\tan\theta \\ \dot{\theta} &= qc\phi - rs\phi \\ \dot{\psi} &= (qs\phi + rc\phi)\sec\theta \end{aligned} \tag{2.2.1}$$

and:

$$(\dot{p}-\omega_0 r)I_x + (q-\omega_0)r(I_z-I_y) = -C_x\dot{\Omega}_x + C_y\Omega_y r - C_z\Omega_z q + T_x$$

$$\dot{q}I_y + pr(I_x-I_z) = C_x\Omega_x r - C_y\dot{\Omega}_y + C_z\Omega_z p + T_y \tag{2.2.2}$$

$$(\dot{r}+\omega_0 p)I_z + (q-\omega_0)r(I_y-I_x) = C_x\Omega_x q - C_y\Omega_y p - C_z\dot{\Omega}_z + T_z$$

However, because the angles ϕ, θ, and ψ are assumed small, the equations of (2.2.1) are approximated by:

$$\dot{\phi} = p, \qquad \dot{\theta} = q, \qquad \dot{\psi} = r \tag{2.2.3}$$

Without loss of generality, it may be assumed that the torques T_x, T_y and T_z constitute control torques.

Let:

$$u_1 = T_x/I_x,\ u_2 = T_y/I_y,\ u_3 = T_z/I_z \qquad (2.2.4)$$

Practically, the controls u_1, u_2 and u_3 would be applied through the use of reaction jets and thus they are of a discontinuous nature. With these controls it may be quite difficult to achieve a high degree of resolution. A much finer control is achievable using the flywheel controls.

Flywheel control may be considerd to be analogous to controlling the speed of a car. As is known, the car's speed is not controlled directly but rather the acceleration is the direct control applied in a prescribed fashion until the desired speed is achieved. Applying this logic to the satellite attitude control problem, the following definitions result:

$$\dot{\Omega}_x = u_4,\ \dot{\Omega}_y = u_5,\ \dot{\Omega}_z = u_6 \qquad (2.2.5)$$

The variables $\Omega_x(t)$, $\Omega_y(t)$, and $\Omega_z(t)$ may be considered as three more state variables thus increasing the dimension of the state space by three. Adjoining equations (2.2.2)-(2.2.5) and expressing in the form:

$$\dot{x} = f(x,u)$$

where;

$$x \equiv (p,q,r,\phi,\theta,\psi,\Omega_x,\Omega_y,\Omega_z)'$$
$$u \equiv (u_1,u_2,u_3,u_4,u_5,u_6)' \qquad (2.2.6)$$

then;

$$\begin{aligned}
\dot{p} &= A_1 r + A_2 qr + A_3 u_4 + A_4 \Omega_y r + A_5 \Omega_z q + A_6 u_1 \\
\dot{q} &= B_1 pr + B_2 \Omega_x r + B_3 u_5 + B_4 \Omega_z p + B_5 u_2 \\
\dot{r} &= C_1 p + C_2 qp + C_3 \Omega_x q + C_4 \Omega_y p + C_5 u_6 + C_6 u_3 \\
\dot{\phi} &= p \\
\dot{\theta} &= q \\
\dot{\psi} &= r \\
\dot{\Omega}_x &= u_4 \\
\dot{\Omega}_y &= u_5 \\
\dot{\Omega}_z &= u_6
\end{aligned} \qquad (2.2.7)$$

with:

$A_1=\omega_0(I_z-I_y-I_x)/I_x$, $A_2=(I_z-I_y)/I_x$, $A_3=-C_x/I_x$, $A_4=C_y/I_x$,
$A_5=-C_z/I_x$, $A_6=1$, $B_1=(I_z-I_x)/I_y$, $B_2=C_x/I_y$,
$B_3=-C_y/I_y$, $B_4=C_z/I_y$, $B_5=1$, $C_2=(I_x-I_y)/I_z$,
$C_1=\omega_0(I_y-I_x-I_z)/I_z$, $C_3=C_x/I_z$, $C_4=-C_y/I_z$, $C_5=-C_y/I_z$,
$C_6=1$. (2.2.8)

A typical cost functional to be minimized subject to (2.2.7) could be;

$$J(u) = \tfrac{1}{2}\left[x' R_1 x\right]\Big|_{t=T} + \int_{t_0}^{T} \tfrac{1}{2}\left[x' R_2 x + u' Q u\right] dt \qquad (2.2.9)$$

where R_1 and R_2 are positive semi-definite matrices and Q is a positive definite matrix. The form of the cost functional to be minimized need not be limited to that shown in

(2.2.9). Indeed more general expressions are acceptable.

With (2.2.7) and (2.2.9) a variety of optimal attitude control problems may be investigated. The dynamics described by (2.2.7) can easily be simplified to represent a number of special situations. As an example, consider the following problem: The body rates p, q and r have to be controlled using only reaction jets. The angles ϕ, θ and ψ are assumed small but their actual values are irrelevant. Since only reaction jets are used, the variables $\Omega_x, \Omega_y, \Omega_z$, $\dot{\Omega}_x, \dot{\Omega}_y$, and $\dot{\Omega}_z$ are assumed zero. With the initial conditions relatively small in magnitude and appropriate weighing in the terminal cost function, the body rates should be in a small neighbourhood of zero at the final time. This problem may be stated as the following optimal cortrol problem;

$$\underset{u\varepsilon U}{\text{minimize}}\ J(u) = \tfrac{1}{2}(p^2+q^2+r^2)\Big|_{t=T} + \tfrac{1}{2}\int_{t_0}^{T}(p^2+q^2+r^2)dt$$

subject to;

$$\begin{aligned}
\dot{p} &= A_1 r + A_2 qr + A_6 u_1 \\
\dot{q} &= B_1 pr + B_5 u_2 \\
\dot{r} &= C_1 p + C_2 qp + C_6 u_3 \\
U &= \left\{-u_{max},\ u_{max}\right\}
\end{aligned} \qquad (2.2.10)$$

with initial conditions:

$$p(t_0) = p_0,\quad q(t_0) = q_0\quad r(o) = r_0 \qquad (2.2.11)$$

The Hamiltonian for this problem is:

$$H = \tfrac{1}{2}(p^2+q^2+r^2)\xi_0 + (A_1 r + A_2 qr + A_6 u_1)\xi_1$$
$$+ (B_1 pr + B_5 u_2)\xi_2 + (C_1 p + C_2 qp + C_6 u_3)\xi_3 \qquad (2.2.12)$$

From H, the Hamiltonian system is given by:

$$\begin{aligned}
\dot{x}_0 &= \tfrac{1}{2}(p^2+q^2+r^2) \\
\dot{p} &= A_1 r + A_2 qr + A_6 u^*_1 \\
\dot{q} &= B_1 pr + B_5 u^*_2 \\
\dot{r} &= C_1 p + C_2 qp + C_6 u^*_3 \\
\dot{\xi}_0 &= 0 \\
\dot{\xi}_1 &= -(\xi_0 p + B_1 \xi_2 + C_1 \xi_3 + C_2 q \xi_3) \\
\dot{\xi}_2 &= -(\xi_0 q + A_2 \xi_1 r + C_2 \xi_3 p) \\
\dot{\xi}_3 &= -(\xi_0 r + A_1 \xi_1 + A_2 q \xi_1 + B_1 p \xi_2)
\end{aligned} \qquad (2.2.13)$$

The initial conditions are;

$$x_0(t_0)=0,\ p(t_0)=p_0,\ q(t_0)=q_0,\ r(t_0)=r_0 \qquad (2.2.14)$$

and the terminal conditions, determined from the transversality conditions, are:

$$\xi_1(T)=p(T),\ \xi_2(T)=q(T),\ \xi_3(T)=r(T) \qquad (2.2.15)$$

To satisfy theorem 2.1.1, set $\xi_0(t)=-1$. The optimal controls u^*_1, u^*_2, and u^*_3 are obtained by maximizing the Hamiltonian in (2.2.12). Hence:

$$\begin{aligned}
u^*_1 &= u_{max}\ \text{sgn}\ (A_6 \xi_1) \\
u^*_2 &= u_{max}\ \text{sgn}\ (B_5 \xi_2) \\
u^*_3 &= u_{max}\ \text{sgn}\ (C_6 \xi_3)
\end{aligned} \qquad (2.2.16)$$

By solving the TPBVP described by (2.2.13) the solution to the optimal control problem stated in (2.2.10) is obtained.

It may be noted that even for one of the simpler cases resulting from (2.2.7) and (2.2.9), the resulting TPBVP to be solved is by no means trivial. The right hand side of the differential equations in (2.2.13) is discontinuous and even with $\xi_0(t)$, a constant, the system is still of order seven.

This is the major difficulty encountered when applying PMP and transforming an optimal control problem to a TPBVP. However, without solving the TPBVP it may be possible to derive the shape of the optimal controls, a valuable feature when applying gradient techniques in Chapter Four.

Other practical attitude control problems could be to investigate controllability when one or more flywheels are defective and without using reaction jets. Or what are the limitations of a non-saturating reaction wheel system? These problems and several variations on them are investigated at a later stage in this thesis.

Optimal Gyrotorquer Attitude Control (OGAC): The formulation of OGAC problems proceeds in a manner similar to that for OFAC problems. Equations (1.2.21) and (1.4.1) describe the dynamics involved. They are;

$$\dot{\phi} = p + (qs\phi+rc\phi)\tan\theta$$

$$\dot{\theta} = qc\phi - rs\phi \qquad (2.2.17)$$

$$\dot{\psi} = (qs\phi+rc\phi)\sec\theta$$

and

$$(\dot{p}-\omega_0 r)I_x + (q-\omega_0)r(I_z-I_y) = 2C_x\Omega_x rs\delta_x - 2C_y\Omega_y(q-\omega_0)s\delta_y - 2C_z\Omega_z\dot{\delta}_z c\delta_z + T_x$$

$$\dot{q}I_y + pr(I_x-I_z) = -2C_x\Omega_x\dot{\delta}_x c\delta_x + 2C_y\Omega_y ps\delta_y - 2C_z\Omega_z(r-\omega_0)s\delta_z + T_y \qquad (2.2.18)$$

$$(\dot{r}+\omega_0 p)I_z + (q-\omega_0)p(I_y-I_x) = -2C_x\Omega_x(p-\omega_0)s\delta_x - 2C_y\Omega_y\dot{\delta}_y c\delta_y + 2C_z\Omega_z qs\delta_z + T_z$$

The gyrotorquers are assumed to be controllable by changing the gyroscopic angular rate. Once more this is in line with practical considerations, as it is not possible to physically change the gyroscopic angle in infinitesimal time. Rather, the angular rate is increased in the direction of the prescribed angle. In practice this amounts to letting:

$$\dot{\delta}_x = u_4, \qquad \dot{\delta}_y = u_5, \qquad \dot{\delta}_z = u_6 \qquad (2.2.19)$$

The quantities δ_x, δ_y, and δ_z can be considered as three state variables. As in the case of flywheel attitude control, equation (2.2.17) is approximated by (2.2.3) and the controls u_1, u_2 and u_3 are defined by (2.2.4). Hence,

expressing the gyrotorquer attitude control dynamics in the form;

$$\dot{x} = f(x,u)$$

where;

$$x \equiv (p,q,r,\phi,\theta,\psi,\delta_x,\delta_y,\delta_z)'$$
$$u \equiv (u_1,u_2,u_3,u_4,u_5,u_6)' \qquad (2.2.20)$$

then;

$$\begin{aligned}
\dot{p} &= A_1 r + A_2 qr + A_3\Omega_x rs\delta_x + A_4\Omega_y qs\delta_y + A_5\Omega_y s\delta_y \\
&\quad + A_6 u_6\Omega_z c\delta_z + A_7 u_1 \\
\dot{q} &= B_1 pr + B_2 u_4\Omega_x c\delta_x + B_3\Omega_y ps\delta_y + B_4\Omega_z rs\delta_z \\
&\quad + B_5\Omega_z s\delta_z + B_6 u_2 \\
\dot{r} &= C_1 p + C_2 qp + C_3 p\Omega_x s\delta_x + C_4\Omega_x s\delta_x + C_5 u_5\Omega_y c\delta_y \\
&\quad + C_6\Omega_z qs\delta_z + C_7 u_3 \\
\dot{\phi} &= p \qquad (2.2.21) \\
\dot{\theta} &= q \\
\dot{\psi} &= r \\
\dot{\delta}_x &= u_4 \\
\dot{\delta}_y &= u_5 \\
\dot{\delta}_z &= u_6
\end{aligned}$$

where:

$A_2=(I_y-I_z)/I_x$, $A_1=\omega_0(I_z-I_y+I_x)I_x$, $A_3=2C_x/I_x$, $A_4=-2C_y/I_x$,
$A_5=2C_y\omega_0/I_x$, $A_6=-2C_z/I_x$ $A_7=1$, $B_1=(I_z-I_x)/I_y$,
$B_2=-2C_x/I_y$, $B_3=2C_y/I_y$, $B_4=-2C_z/I_y$ $B_5=2C_z\omega_0/I_y$,
$B_6=1$, $C_1=-\omega_0(I_x-I_y+I_z)/I_z$, $C_2=(I_x-I_y)/I_z$, $C_3=-2C_x/I_z$,
$C_4=2C_x\omega_0/I_z$, $C_5=-2C_y/I_z$, $C_6=2C_z/I_z$, $C_7=1$.

(2.2.22)

By defining a cost functional to be minimmized subject to the dynamics defined by (2.2.21) for some set of admissible controls an OGAC problem is described. In general, the initial conditions are given and a set of terminal conditions are desired. Using a cost functional similar to that of (2.2.9) a variety of optimal control problems can be formulated. As for OFAC, various cases such as when one or more of the gyro-torquers are defective can be investigated.

However, in this thesis more emphasis is placed on the study of OFAC problems because being less nonlinear, they are less computationally time consuming. Hence more experiments can be performed.

Summary.

In this chapter some basic theorems of optimal control theory have been reviewed. Then from the attitude dynamics, a set of optimal control problems falling within the range of the above theorems was formulated. Some of the basic difficulties involved in the solution of optimal control problems by PMP were pointed out. For example, the solution of a complicated TPBVP. However, PMP may lead to apriori knowledge of the types of control required.

In the following chapter, computational methods for the solution of optimal control problems are investigated. Some

methods that solve TPBVP's and some totally different approaches for solution of the optimal control problems are presented.

Computational Methods In Optimal Control.

In this chapter, a short survey of various techniques available for the solution of optimal control problems that are reducible to a TPBVP, is presented. A brief discussion on the limitations of the techniques commonly used to solve the resulting TPBVP's follows. Then consideration is given to methods of solving the optimal control problem by iteratively updating the controls. Of particular interest, is the conjugate gradient descent (CGD) method employed in the computational exercises for this thesis. This method is discussed in considerable detail and a derivation is given for computing the gradient - of the cost functional - in a variety of cases encountered in optimal control. Finally, the special problems encountered with having a free terminal time, and state constraints are considered.

3.1 Techniques for Solving TPBVP's.

In this section a few techniques for solving a TPBVP are presented. They range from a discretization approach to a linearization approach, in cases when the TPBVP is nonlinear.

Finite Difference Methods: The finite difference method of solving the TPBVP, converts the set of ordinary differential

equations, into a finite set of algebraic or transcendental equations [10]. The solution of the set of algebraic or transcendental equations, yields approximations to the solution of the original differential equations.

Conceptually, the finite difference methods may be quite appealing, however, for systems of nonlinear ordinary differential equations, the formulation of the finite difference equations can be a difficult and time consuming task. Furthermore, convergence of the algebraic solution to the solution of the differential equations, may depend critically on the numerical integration step size used.

Because of the complications involved in transforming a TPBVP from its differential form to a set of algebraic equations and the convergence uncertainty, it was decided to forego this approach.

Quasilinearization: Quasilinearization is a method applicable only to nonlinear TPBVP's [11]. It proceeds by linearizing the nonlinear ordinary differential equations around a nominal solution, satisfying the specified boundary conditions. In a linear TPBVP sequence, the solution of the k'th linear TPBVP satisfies the specified boundary conditions and is taken as the nominal solution for the (k+1)st generated linear TPBVP.

In Roberts and Shipman [12], it is shown that quasilinearization is a realization of the Newton-Raphson method. Therefore, the initial state trajectory profile assumed for the quasilinearization is crucial to the success of the method. This is the major difficulty of this method especially since the TPBVP's resulting from SOAC problems have fairly high dimensions. Guessing the initial trajectories could very well prove to be a futile effort. Based on this observation, it was decided to reject the quasilinearization approach .

Shooting Methods: These methods derive their name from the problem of finding the angle, at which a projectile must be shot in order to hit a target at a given distance. By varying the shooting angle, the miss distantce will be changed. Hence, by analyzing the miss distance, a corrective procedure for the shooting angle is generated.

The classical and best known of the shooting methods is the Goodman-Lance method [13] or simply the method of adjoints [12]. In the case of a linear TPBVP, the method of adjoints computes the missing initial conditions in one pass through the process. However, for a nonlinear TPBVP a sequence of corrections to the trial values of the missing initial conditions is generated.

This procedure requires that a guess be made for the

missing initial conditions. For nonlinear problems an unlucky guess may lead to instability problems. In such a case one can use a continuation approach, which consists of starting with a small portion of the time interval, over which the TPBVP is to be solved. The missing initial conditions, obtained from the solution of this TPBVP, are used as the nominal conditions for a larger portion of the solution interval. This procedure is repeated until the solution is obtained for the desired time interval. This is a good procedure for sensitive problems but it is computationally very time consuming.

Function Minimization Approach: In this approach, the boundary conditions specified at the right-hand end of the solution trajectory of the Hamiltonian system are molded into an appropriate error function, whose minimization coincides with the solution of the TPBVP [14]. Minimization of this function is achieved by changing the missing initial conditions.

Many multidimensional minimization techniques are available for the solution of such problems. A software package incorporating a number of such methods has been developped by Birta [15] and provides such techniques as the Davidon-Fletcher-Powell method [16], and the Simplex method [17] etc.

The function minimization approach is easily implemented

since a package already exists. However, the major difficulty is in making a good guess for the missing initial conditions, i.e. one which will not create stability problems. For the satellite optimal attitude control problems considered in this thesis, this is a major difficulty and is the reason for rejecting this approach.

3.2 Alternatives to Solving the TPBVP.

In this section, techniques for which the solution of optimal control problems is not achieved through the direct solution of a TPBVP, are discussed. For example, using invariant imbedding, the optimal control problem can be reformulated as an initial value problem [18] or using dynamic programming, it can be reformulated as the solution of a partial differential equation [19]. Finally, with the gradient techniques the control is iteratively updated.

Invariant Imbedding: Invariant imbedding consists of imbedding a new variable in the TPBVP. This imbedding results in the generation of a partial differential equation with initial conditions given. The TPBVP is thus reduced to an initial value problem. However, the solution of this partial differential equation is difficult and very demanding in computer memory. This approach has been applied successfully to low dimensional problems [18]. Considering the large dimensions of our optimal control problems, this is

not a viable approach.

Dynamic Programming: The dynamic programming approach is based on interchanging the minimization and integration procedures which leads to a partial differential equation called the Hamilton-Jacobi-Bellman equation [19,20]. This equation is generally quite difficult to solve because one of its terms requires a minimization over the set of admissible controls. However, when it is possible to solve it, the resulting optimal control is given directly in a feedback form. As in the case of invariant imbedding, the dynamic programming approach requires a large amount of computer memory and for this reason it was not selected for the solution of SOAC problems in this thesis.

Gradient Techniques: With the gradient techniques, the controls are iterated in such a way as to directly minimize the cost functional for the optimal control problem. These methods offer the advantage that even with a very crude guess of the optimal control, convergence to the optimal control can still be achieved. The convergence rate is generally not outstanding, although for the CGD there is quadratic convergence [21] as the control sequence approaches the optimal value. The gradient techniques can easily be applied in optimal control problems having discontinuous controls [21,22].

A salient feature of the gradient techniques is that for each iteration the state differential equations in the Hamiltonian system are first integrated forward and then the co-state differential equations in the Hamiltonian system are integrated backwards. This forward then backward integration order eliminates instability problems and is the principle reason for choosing the CGD method in this thesis.

3.3 Solution of Optimal Control Problems by the Conjugate Gradient Descent (CGD) Method.

Consider the following optimal control problem;

$$\text{minimize} \quad J = \Theta(x(t_f))$$

subject to:

$$\dot{x} = f(x,u), \quad x(t_0) = c \tag{3.3.1}$$

for some $u \in U$, the set of admissible controls, where $f(.,.)$ is a function whose range is in R^n, and $\Theta(.)$ is a real-valued function on R^n with $x(t_f)$ the state of the system at the final time t_f. The time interval of interest is therefore $[t_0, t_f]$. The basic assumptions made here are as follows:

(1) f is such that unique, bounded solutions (starting at $x(t_0)=c$) exist for bounded $||u||$ and can be found by numerical integration of the differential equations in (3.3.1);

(2) f has partial derivatives with respect to all components of x and u for $||x||$ and $||u||$ bounded;

(3) Θ has partial derivatives with respect to all components

of x for $\|x\|$ bounded.

The CGD process and algorithm can be described as follows [21]: Consider the real-valued functional J(.). The u that minimizes J(u) is sought. A sequence u_0, u_1, ... is constructed such that $J(u_{i+1}) < J(u_i)$. With g_i as the gradient of J evaluated at u_i, then, the CGD algorithm used is [21]:

$$\left.\begin{aligned} s_0 &= g_0 \\ s_{i+1} &= -g_{i+1} + \beta_i s_i \\ \beta_i &= \langle g_{i+1}, g_{i+1} \rangle / \langle g_i, g_i \rangle \\ u_{i+1} &= u_i + \alpha_i s_i \end{aligned}\right\} \qquad (3.3.2)$$

where α_i is chosen to minimize $J(u_i + \alpha_i s_i)$ and $\langle .,. \rangle$ is a scalar product. It can be shown that the resulting sequence of u_i's converges to u^*, the minimizing argument of J, if the initial guess u_0 is sufficiently close to u^*.

Linear Search: It is desired to minimize $f(\alpha) = J(u_i + \alpha s_i)$. The following assumptions are made:

(1) J is a continuous function of u;

(2) J(u) and g(u) can be evaluated for any $u \in U$.

Then proceed as follows:

(1) Choose α_0, the estimate of the minimizing argument of f;

(2) Compute $\alpha_k = 2^s \alpha_{k-1}$ and $f(\alpha_k)$, k=1, 2, ... until $f(\alpha_k) > f(\alpha_{k-1})$ where s=+1 if $f(\alpha_0) < f(0)$ and s=-1 otherwise. Note if $k \leq 2$ when $f(\alpha_k) > f(\alpha_{k-1})$ occurs, another

estimate of α_0 must be taken and the procedure repeated;

(3) Fit a third order polynominal through;

$$f(\alpha_{k-2}),\ f(\alpha_{k-1}),\ f(\frac{\alpha_{k-1} + \alpha_k}{2}),\ f(\alpha_k)$$

and find α_i (the end result of the minimizing process) by finding the value of α that minimizes this third order polynomial between α_{k-2} and α_k.

3.4 The Gradient Of The Cost Functional In Selected Cases Of Optimal Control Problems.

In order to apply the CGD algorithm described in the previous section, it is necessary to compute the gradient of the cost functional that is to be minimized. From PMP (Chapter Two) it was shown that quite often the optimal controls are of a bang-bang nature and therefore piecewise continuous. Hence, a derivation of the gradient for an optimal control problem, with piecewise continuous input controls, will be a great asset in the solution of satellite attitude control problems to be tackled in the next chapter.

The following development is a generalization of the derivation presented in Chapter Five, section two of Hasdorff [21]. The theory presented by Hasdorff is essentially that required for the solution of typical satellite optimal attitude control problems. The basic limitation of Hasdorff's presentation is that it is given for only one control. In

this section the theory is generalized to the case of m controls.

Consider the optimal control problem given by;

$$\text{minimize } J = \Theta(x(t_f)) \tag{3.4.1}$$

subject to:

$$\dot{x} = f(x,u), \quad x(t_0) = c \tag{3.4.2}$$

The system is assumed to have n state variables;

$$\text{i.e. } x \equiv (x_1, x_2, \ldots, x_n)' \tag{3.4.3}$$

and m controls:

$$\text{i.e. } u \equiv (u_1, u_2, \ldots, u_m)' \tag{3.4.4}$$

Thus $x \in P^n$ and $u \in U^m$, where U^m, the set of admissible controls, is an appropriate Hilbert space. The assumptions taken here are as follows:

(1) the controls are piecewise continuous;

(2) all the controls have the same number of discontinuities.

Graphically each control may be represented by Fig.3.4.1 ;

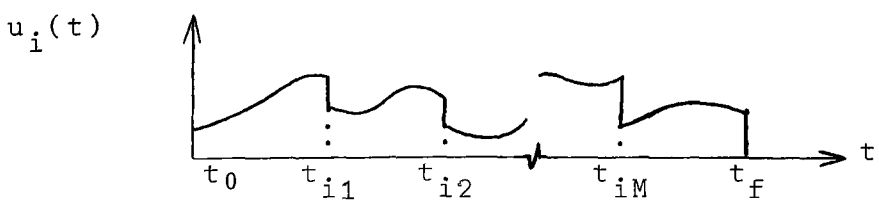


Fig. 3.4.1 Typical Piecewise Continuous Control.

which may be expressed mathematically by:

$$u_i(t) = c_i(t) + \sum_{j=0}^{M} h_{ij}\left[1(t-t_{ij}) - 1(t-t_{ij+1})\right] \tag{3.4.5}$$

Consider equation (3.4.5) as defining a mapping from $C_{[t_0,t_f]}xR^M xR^{M+1}$ to $PC^M_{[t_0,t_f]}$ where $C_{[t_0,t_f]}$ is the space of continuous functions on $[t_0,t_f]$ and $PC^M_{[t_0,t_f]}$ is the space of piecewise continuous functions having M discontinuities on $[t_0,t_f]$. The function 1(.) is the unit step:

$$\text{i.e.} \quad 1(t) = \begin{cases} 1 & t \geq 0 \\ 0 & t < 0 \end{cases} \tag{3.4.6}$$

Now define the following vector spaces;

$$A \equiv C_{[t_0,t_f]} xR^M xR^{M+1} \tag{3.4.7}$$

and;

$$U \equiv PC^M_{[t_0,t_f]} \tag{3.4.8}$$

where $c_i(t) \in C_{[t_0,t_f]}$, $\bar{t}_i \in R^M$, and $\bar{h}_i \in R^{M+1}$ for i=1, 2, ...,m. Let:

$$T : A \longrightarrow U \tag{3.4.9}$$

Then:

$$T(c_i,\bar{t}_i,\bar{h}_i) = u_i(t) \in U \tag{3.4.10}$$

That is to say for every $w \in A$, $T(w)=u \in U$. By definition the operation of T is as shown in (3.4.5). However, in this case there are m controls. Hence, from (3.4.10) a mapping T can be defined such that;

$$T^m : A^m \longrightarrow U^m \tag{3.4.11}$$

where;

$$A^m \equiv \left[C_{[t_0,t_f]} xR^M xR^{M+1}\right]^m \tag{3.4.12}$$

$$U^m \equiv \left[PC^M_{\left[t_0,t_f\right]}\right]^m \tag{3.4.13}$$

Then, for $W_I=(w_1,w_2,\ldots,w_m)'\epsilon A^m$ and $w_i=(c_i,\bar{t}_i,\bar{h}_i)\epsilon A$,

$$T^m(w_1,w_2,\ldots,w_m) = (u_1,u_2,\ldots,u_m)'\epsilon U^m \tag{3.4.14}$$

Inner Products: Let $c_1(t)$, $c_2(t)\epsilon C_{\left[t_0,t_f\right]}$, then:

$$\langle c_1(t),\ c_2(t)\rangle_{C_{\left[t_0,t_f\right]}} \equiv \int_{t_0}^{t_f} c_1(t)c_2(t)dt \tag{3.4.15}$$

Let $\bar{t}_1,t_2\epsilon R^M$, then:

$$\langle \bar{t}_1,\bar{t}_2\rangle_{R^M} \equiv \sum_{i=1}^{M} t_{1i}t_{2i} \tag{3.4.16}$$

Let $\bar{h}_1$, $\bar{h}_2\epsilon R^{M+1}$, then:

$$\langle \bar{h}_1,\bar{h}_2\rangle_{R^{M+1}} \equiv \sum_{i=0}^{M} h_{1i}h_{2i} \tag{3.4.17}$$

Therefore:

$$\langle w_1,w_2\rangle_A = \int_{t_0}^{t_f} c_1(t)c_2(t)dt + \sum_{i=1}^{M} t_{1i}t_{2i} + \sum_{i=0}^{M} h_{1i}h_{2i} \tag{3.4.18}$$

Consequently if $w_i=(c_i,\bar{t}_i,\bar{h}_i)\epsilon A$ and $W=(w_1,w_2,\ldots\ w_m)'\epsilon A^m$, then for W_1, W_2 the inner product is defined as follows:

$$\langle W_1,W_2\rangle_{A^m} = \langle (w_{11},w_{12},\ldots,w_{1m}),(w_{21},w_{22},\ldots,w_{2m})\rangle = \sum_{i=1}^{m} \langle w_{1i},w_{2i}\rangle_A \tag{3.4.19}$$

then;

$$\langle W_1,W_2\rangle_{A^m} = \sum_{i=1}^{m}\int_{t_0}^{t_f} c_{1i}(t)c_{2i}(t)dt + \sum_{i=1}^{m}\sum_{j=1}^{m} t_{1ij}t_{2ij} + \sum_{i=1}^{m}\sum_{j=0}^{m} h_{1ij}h_{2ij} \tag{3.4.20}$$

therefore:

$$\left\langle W_1, W_2 \right\rangle_{A^m} = \sum_{i=1}^{m} \left\langle c_{1i}, c_{2i} \right\rangle_{C\left[t_0, t_f\right]} + \sum_{i=1}^{m} \left\langle \bar{t}_{1i}, \bar{t}_{2i} \right\rangle_{R^M} + \sum_{i=1}^{m} \left\langle \bar{h}_{1i}, \bar{h}_{2i} \right\rangle_{R^{M+1}} \tag{3.4.21}$$

It is important to have the gradient or A^m properly defined since it will be required when computing the gradient of the cost functional. To do this, it is necessary to introduce the notion of the derivative of a functional.

Derivative of a Functional: Consider $J(.)$, a continuous functional from a normed space A^m to the real numbers R. $J(.)$ is said to be differentiable at $x_0 \in A^m$, if a linear mapping, let it be called $J'(x_0)$, can be shown to exist which satisfies;

$$\lim_{z \to 0} \frac{\| J(x_0+z) - J(x_0) - J'(x_0).z \|}{\| z \|} = 0 \tag{3.4.22}$$

for all $z \in A^m$. $J'(x_0)$ is called the derivative of $J(.)$ at x_0.

A practical approach for determining the derivative of a given differentiable functional $J(.)$ at some x_0 will now be shown. Consider vectors z of the form $z=\varepsilon W$ where W is a fixed vector of unit norm and ε is small number, (a parameter) substitution in (3.4.22) gives:

$$\lim_{\varepsilon \to 0} \frac{\| J(x_0+\varepsilon W) - J(x_0) - J'(x_0).\varepsilon W \|}{\varepsilon} = 0 \tag{3.4.23}$$

But since the norm is a linear operation, the limit can be

moved inside the norm operation. This implies that:

$$J'(x_0)\cdot W = \lim_{\varepsilon \to 0} \frac{J(x_0+\varepsilon W) - J(x_0)}{\varepsilon} \tag{3.4.24}$$

$$= \frac{d}{d\varepsilon} J(x_0+\varepsilon W)\Big|_{\varepsilon=0} \tag{3.4.25}$$

The expression for $J'(x_0)\cdot W$ given by (3.4.25) is known as the Gateaux variation [23]. Assuming that $J(.)$ is differentiable at x_0, it may be shown that $J'(x_0)$ is a continuous linear functional on the space A^m. Then according to the Riesz representation theorem [24], it follows that there exists an element $g(x_0)\in A^m$ such that for $W\in A^m$,

$$J'(x_0)\cdot W = \left\langle g(x_0),\ W \right\rangle_{A^m} \tag{3.4.26}$$

Hence from (3.4.25) and (3.4.26) it follows that;

$$\frac{d}{d\varepsilon} J(x_0+\varepsilon W)\Big|_{\varepsilon=0} = \left\langle g(x_0), W \right\rangle_{A^m} \tag{3.4.27}$$

where $g(x_0)$ is the gradient of the function $J(.)$ at x_0.

Application of the Gateaux Variation to the Optimal Control Problem: Equation (3.4.1) subject to (3.4.2) defines a cost functional on the space of input control functions $u(t)$ for $t\in[t_0,t_f]$, denoted by $u_{[t_0,t_f]}$. Hence, for a given input $u_{[t_0,t_f]}$, (3.4.2) can be integrated and a corresponding value for $x(t_f)$ determined. The cost functional could then be expressed by;

$$J(u) = \theta(x(t_f,u)) \tag{3.4.28}$$

where the time interval over which the solution is sought is $I=[t_0, t_f]$. Thus applying (3.4.27) gives;

$$\frac{d}{d\varepsilon} J(u+\varepsilon W)\Big|_{\varepsilon=0} = \frac{d}{d\varepsilon} \theta(x(t_f+\varepsilon W))\Big|_{\varepsilon=0}$$

$$= \left\langle g(u), W \right\rangle_{A^m} \tag{3.4.29}$$

where g(u) is the gradient of the cost functional. But (3.4.28) can be simplified further since;

$$\frac{d}{d\varepsilon} \theta(x(t_f,u+\varepsilon W))\Big|_{\varepsilon=0} = \left[\nabla_x \theta(x(t_f,u+\varepsilon W)) \cdot \frac{d}{d\varepsilon} x(t_f,u+\varepsilon W)\right]\Big|_{\varepsilon=0}$$

$$= \left\langle \nabla_x \theta(x(t_f,u+\varepsilon W)), \frac{d}{d\varepsilon} x(t_f,u+\varepsilon W) \right\rangle_{A^m}\Big|_{\varepsilon=0} \tag{3.4.30}$$

or simply;

$$\frac{d}{d\varepsilon} \theta(x(t_f,u)) = \left\langle \nabla_x \theta(x(t_f,u)), \frac{d}{d\varepsilon} x(t_f,u) \right\rangle_{A^m} \tag{3.4.31}$$

where the evaluation at $\varepsilon=0$ will be implied and where:

$$\nabla_x \theta(\cdot) \equiv \left[\frac{\partial \theta(\cdot)}{\partial x_1}, \ldots, \frac{\partial \theta(\cdot)}{\partial x_n}\right]' \tag{3.4.32}$$

The goal is to obtain an expression for g(u) from the relationship:

$$\left\langle g(u), W \right\rangle_{A^m} = \left\langle \nabla_x \theta(x(t_f),u), \frac{dx}{d\varepsilon}(t_f,u) \right\rangle_{A^m} \tag{3.4.33}$$

An expression for $\frac{dx}{d\varepsilon}(t_f,u)$ is first obtained. Integrating

(3.4.2) gives:

$$x(t_f,u) = x(t_0) + \int_{t_0}^{t_f} f(x,u)dt \tag{3.4.34}$$

Since there are m controls each having M discontinuities, the integral from t_0 to t_f may be broken down into a summation of integrals with each integral performed over a time interval such that all the controls are continuous. Thus:

$$x(t_f,u) = x(t_0) + \sum_{i=0}^{m.M} \int_{\tau_i}^{\tau_{i+1}} f(x,u_{1i},u_{2i},\ldots,u_{mi})dt \tag{3.4.35}$$

Let $\varepsilon W \varepsilon U^m$ be a perturbation in the controls, then from the definition (3.4.5), it follows that (3.4.35) becomes:

$$x(t_f,u+\varepsilon W) = x(t_0) + \sum_{i=0}^{m.M} \int_{\tau_i+\varepsilon s_i}^{\tau_{i+1}+\varepsilon s_{i+1}} f(x,c_1+\varepsilon v_1+h_{1i}+\varepsilon z_{1i}, \ldots,c_m+\varepsilon v_m+h_{mi}+\varepsilon z_{mi})dt \tag{3.4.36}$$

The salient feature of the set of vectors $\bar{h}_0$, $\bar{h}_1$, ..., $\bar{h}_{m.M}$ and $\bar{z}_0$, $\bar{z}_1$, ..., $\bar{z}_{m.M}$ with dimension m is that $\bar{h}_i$ and $\bar{z}_i$ are constant over the time interval $[\tau_i+\varepsilon s_i, \tau_{i+1}+\varepsilon s_{i+1}]$ for i=0, 1, ... ,(m.M) respectively (see for example Appendix A). Now differentiating (3.4.36) w.r.t. ε and setting ε=0 gives:

$$\frac{d}{d\varepsilon} x(t_f,u+\varepsilon W)\Big|_{\varepsilon=0} = \int_{t_0}^{t_f} [f_x(x,u)\cdot\frac{d}{d\varepsilon} x(t_f,u) + f_u(x,u)\cdot v]dt + \sum_{i=0}^{m.M} \int_{\tau_i}^{\tau_{i+1}} f_u(x,u)\bar{z}_i dt + \sum_{i=1}^{m.M} [f(x,u)\Big|_{t_i^-} - f(x,u)\Big|_{t_i^+}]s_i \tag{3.4.37}$$

By using the unit step function $1(t)$ and the unit impulse $\delta(t)$, equation (3.4.37) may be written in the following form (the variable of integration has beer charged to σ to avoid

later confusion with t):

$$\frac{d}{d\varepsilon}x(t_f,u) = \int_{t_0}^{t_f}\Big[f_x(x,u)\,\frac{d}{d\varepsilon}x(t_f,u) + f_u(x,u)\cdot v + \sum_{i=0}^{m.M} f_u(x,u)\cdot\bar{z}_i\cdot\big[1(\sigma-\tau_i)-1(\sigma-\tau_{i+1})\big] + \sum_{i=1}^{m.M}\Big[f(x,u)\Big|_{\tau_i^-} - f(x,u)\Big|_{\tau_i^+}\Big]s_i\,\delta(\sigma-\tau_i)\Big]d\sigma \qquad (3.4.38)$$

Equation (3.4.38) is an integral equation in $\frac{dx}{d\varepsilon}(t_f,u)$. Let $t_f=t$ in (3.4.38) and differentiate w.r.t. t to obtain an equation of the form $\dot{x}=A(t)x(t)+B(t)$, $x(t_0)=0$ for which the solution is well known. Using $\Phi(t,\sigma)$ as the transition operator, the solution of (3.4.38) takes the following form:

$$\frac{d}{d\varepsilon}x(t_f,u) = \int_{t_0}^{t_f}\Phi(t_f,\sigma)f_u(x,u)v\,d\sigma + \sum_{i=1}^{m.M}\Phi(t_f,\tau_i)\cdot\Big[f(x,u)\Big|_{\tau_i^-} - f(x,u)\Big|_{\tau_i^+}\Big]\cdot s_i + \sum_{i=0}^{m.M}\int_{\tau_i}^{\tau_{i+1}}\Phi(t_f,\sigma)f_u(x,u)\bar{z}_i\,d\sigma \qquad (3.4.39)$$

Substituting in (3.4.33) it follows that:

$$\Big\langle g(u),W\Big\rangle_{A^m} = \Big\langle \nabla_x\theta(x(t_f,u)),\ \int_{t_0}^{t_f}\Phi(t_f,\sigma)f_u(x,u)\cdot v\,d\sigma\Big\rangle + \Big\langle \nabla_x\theta(x(t_f,u)),\ \sum_{i=1}^{m.M}\Phi(t_f,\tau_i)\Big[f(x,u)\Big|_{\tau_i^-} - f(x,u)\Big|_{\tau_i^+}\Big]s_i\Big\rangle + \Big\langle \nabla_x\theta(x(t_f,u)),\ \sum_{i=0}^{m.M}\int_{\tau_i}^{\tau_{i+1}}\Phi(t_f,\sigma)f_u(x,u)\cdot\bar{z}_i\,d\sigma\Big\rangle \qquad (3.4.40)$$

$$\text{or}\quad \Big\langle g(u),W\Big\rangle_{A^m} = G_1 + G_2 + G_3 \qquad (3.4.41)$$

The quantities G_1, G_2, and G_3 can be written in a form which is more convenient for identifying the gradient terms. To

simplify (3.4.40) the co-state vector $\xi(t)$ defined by;

$$\xi(t) = \Phi'(t_f,t)\nabla_x\theta(x(t_f)) \tag{3.4.42}$$

is introduced. It may be verified that $\xi(t)$ also satisfies the Hamiltonian system described earlier in Chapter Two by (2.1.7)- (2.1.8). Therefore;

$$\begin{aligned} G_1 &= \int_{t_0}^{t_f} \left\langle f_u(x,u)\ \Phi'(t_f,\sigma)\nabla_x\theta(x(t_f)), v(\sigma)\right\rangle d\sigma \\ &= \int_{t_0}^{t_f} \left\langle f_u'(x,u)\xi(\sigma), v(\sigma)\right\rangle d\sigma \\ &= \int_{t_0}^{t_f} \sum_{i=1}^{m} \left[f_u'(x,u)\xi(\sigma)\right]_i \cdot v_i(\sigma)d\sigma \\ &= \sum_{i=1}^{m} \int_{t_0}^{t_f} \left[f_u'(x,u)\xi(\sigma)\right]_i \cdot v_i(\sigma)d\sigma \\ &= \sum_{i=1}^{m} \left\langle g_i^{G_1}(u), v_i\right\rangle_{C\left[t_o,t_f\right]} \end{aligned} \tag{3.4.43}$$

where;

$$g_i^{G_1}(u) = \left[f_u'(x,u)\xi(t)\right]_i \tag{3.4.44}$$

Proceeding in a similar fashion for G_2:

$$\begin{aligned} G_2 &= \sum_{i=1}^{m.M} \left\langle \nabla_x\theta(x(t_f,u),\ \Phi(t_f,\tau_i)\left[f(x,u)\Big|_{\tau_i^-} - f(x,u)\Big|_{\tau_i^+}\right]\cdot s_i\right\rangle \\ &= \sum_{i=1}^{m.M} \left\langle \left[f(x,u)\Big|_{\tau_i^-} - f(x,u)\Big|_{\tau_i^+}\right]' \xi(\tau_i), s_i\right\rangle \\ &= \sum_{i=1}^{m.M} \left[f(x,u)\Big|_{\tau_i^-} - f(x,u)\Big|_{\tau_i^+}\right]' \xi(\tau_i)s_i \end{aligned} \tag{3.4.45}$$

Equation (3.4.45) may be expressed as m summations of the following form;

$$G_2 = \sum_{i=1}^{m} \sum_{j=1}^{M} \left[f(x,u)\Big|_{t_{\bar{i}j}} - f(x,u)\Big|_{t^{+}_{ij}} \right]' \xi(t_{ij}) r_{ij} \qquad (3.4.46)$$

where t_{ij} is the j'th time of discontinuity for the control $u_i(t)$ and r_{ij} is the perturbation associated with the switching time t_{ij}. Using inner product notation it follows that;

$$G_2 = \sum_{i=1}^{m} \left\langle \left[f(x,u)\Big|_{\bar{t}_{\bar{i}}} - f(x,u)\Big|_{\bar{t}^{+}_{i}} \right]' \xi(\tau_i), \bar{r}_i \right\rangle$$

$$= \sum_{i=1}^{m} \left\langle g_i^{G_2}(u), \bar{r}_i \right\rangle_{R^M} \qquad (3.4.47)$$

where:

$$g_i^{G_2}(u) = \left[f(x,u)\Big|_{\bar{t}_{\bar{i}}} - f(x,u)\Big|_{\bar{t}^{+}_{i}} \right]' \xi(t_i) \qquad (3.4.48)$$

Proceeding similarly for G_3:

$$G_3 = \sum_{i=0}^{m.M} \left\langle \nabla_x \Theta(x(t_f,u)), \int_{\tau_i}^{\tau_{i+1}} \Phi(t_f,\sigma) f_u(x,u) \cdot \bar{z}_i d\sigma \right\rangle$$

$$= \sum_{i=0}^{m.M} \int_{\tau_i}^{\tau_{i+1}} \left\langle f_u'(x,u) \Phi'(t_f,\sigma) \nabla_x \Theta(x(t_f,u)), \bar{z}_i \right\rangle d\sigma$$

$$= \sum_{i=0}^{m.M} \int^{\tau_{i+1}}_{\tau_i} \left\langle f_u(x,u)\xi(\sigma), \bar{z}_i \right\rangle d\sigma \qquad (3.4.49)$$

$$= \sum_{i=0}^{m.M} \int^{\tau_{i+1}}_{\tau_i} \xi'(\sigma)\, f_u(x,u) \bar{z}_i d\sigma$$

At this point it is convenient to define a new set of vectors $\bar{y}_{ij}$ i =1, 2, ..., m; j=0, 1, ..., M; where each vector in the set has dimension m. Some details on the definition of these vectors is given in Appendix A. By definition each

of the vectors $\bar{y}_{ij}$ has only one non-zero element. Thus it is possible to replace the single sum in (3.4.49) by an equivalent double sum. Hence replacing $\bar{y}$ for $\bar{z}$ in (3.4.49) it follows that;

$$G_3 = \sum_{i=1}^{m} \sum_{j=0}^{M} \int_{t_{ij}}^{t_{ij+1}} \xi'(\sigma) f_u(x,u) \bar{y}_{ij} d\sigma$$

$$= \sum_{i=1}^{m} \left\langle g_i^{G_3}(u), \bar{y}_i \right\rangle_{R^{M+1}} \tag{3.4.50}$$

where,

$$g_i^{G_3}(u) = [\int_{t_{i0}}^{t_{i1}} \xi'(\sigma) f_u(x,u) d\sigma, \ldots, \int_{t_{iM}}^{t_{if}} \xi'(\sigma) f_u(x,u) d\sigma]' \tag{3.4.51}$$

Thus a complete expression for the gradient of the cost functional in the space of input controls can be written as follows;

$$g(u) = \left[g^{G_1}(u) \vdots g^{G_2}(u) \vdots g^{G_3}(u) \right]' \tag{3.4.52}$$

where;

$$g^{G_1}(u) = \left[g_1^{G_1}(u), g_2^{G_1}(u), \ldots, g_m^{G_1}(u) \right]' \tag{3.4.53}$$

$$g^{G_2}(u) = \left[g_1^{G_2}(u), g_2^{G_2}(u), \ldots, g_m^{G_2}(u) \right]' \tag{3.4.54}$$

$$g^{G_3}(u) = \left[g_1^{G_3}(u), g_2^{G_3}(u), \ldots, g_m^{G_3}(u) \right]' \tag{3.4.55}$$

With equations (3.4.53), (3.4.54), and (3.4.55) it is straight forward to give an algorithm for the solution of optimal control problems having piecewise continuous input controls.

3.5 Computing Optimal Controls by the CGD Method.

With the gradient of the cost functional properly defined in the space of input controls, an algorithm for the numerical solution of optimal control problems by the CGD method can be readily formulated as follows:

(1) guess an initial set of controls;

(2) using the present controls integrate (3.4.2) over the time interval $[t_0, t_f]$ and store the state trajectories;

(3) compute the terminal conditions for the co-state differential equations using (3.4.42);

(4) integrate the co-state differential equations as defined using Hamiltonian notation backwards in time from t_f to t_0 and store the trajectories;

(5) using the stored state and co-state trajectories, evaluate the gradients as defined by (3.4.53)-(3.4.55);

(6) stop the process if all the gradients are zero (The present controls are optimal). Otherwise continue to step 7;

(7) using the CGD algorithm described by (3.3.2) compute the optimal search directions for the controls;

(8) perform a linear search by following the steps described in the subsection 'Linear Search';

(9) compute the new controls according to the results of the linear search;

(10) stop if the updated controls are within a prescribed

distance of the previous controls or if a maximum number of iterations has been exceeded. Otherwise return to step 2.

A flow chart illustrating this process is given in Appendix B.

3.6 State-Variable Constraints.

In physical systems it is very common to impose constraints on the state trajectories as well as the controls. This requirement can easily be imbedded in the original problem [25].

Let $x(t)$ be the state trajectory satisfying (3.4.2). Consider the s-dimensional inequality constraint h given by;

$$h(x(t),t) \geq 0 \tag{3.6.1}$$

where each component of h is assumed to be continuously differentiable in state space. (3.6.1) can be converted to an equality constraint by the introduction of a new state variable;

$$\text{i.e.} \quad \dot{x}_{n+1} = f_{n+1} = h_1^2(x,t)\cdot H(h_1) + \ldots + h_s^2(x,t)\cdot H(h_s) \tag{3.6.2}$$

The function $H(h_j(x,t))$ is defined as follows;

$$H(h_j(x,t)) = \begin{cases} 0 & \text{if} \quad h_j(x,t) \geq 0 \\ k_j & \text{if} \quad h_j(x,t) < 0 \end{cases} \tag{3.6.3}$$

with $k_j>0$. The initial condition is $x_{n+1}(t_0)=0$. Hence, the requirement that $x_{n+1}(t_f)=0$ imposes the restriction that the inequality constraints not be violated.

This approach to state variable constraints blends into the previous developments without forcing any changes.

3.7 Free Terminal Time.

The procedure taken here is an extension of section 3.4. Again the controls are assumed to be piecewise continuous. Consider the optimal control problem described by (3.4.1)-(3.4.2). With the terminal time free, the optimal control is specified by a pair $(u^*,t_f^*)\in U^m\times R$. Let $T:A\times R\longrightarrow U$, where A and U are defined by (3.4.7) and (3.4.8) respectively. From this defirition an extension can easily be made to give a mapping $T^m:A^m\times R\longrightarrow U^m$. Let $G\equiv A^m\times R$, then for $v,z\in G$, the quantity g(v) in;

$$\frac{d}{d\varepsilon}J(v+\varepsilon z)\Big|_{\varepsilon=0}=\left\langle g(v),z\right\rangle_G \tag{3.7.1}$$

must be evaluated. Let $z=(\delta u,\delta t_f)'$. Then, from (3.4.21):

$$\left\langle g(v),z\right\rangle_G=\left\langle g(u),\delta u\right\rangle_{A^m}+\left\langle g(t_f),\delta t_f\right\rangle_R \tag{3.7.2}$$

The first term on the right hand side of (3.7.2) was completely derived in section 3.4. The term $\left\langle g(t_f),t_f\right\rangle_R$ in (3.7.2) is contributed by the perturbation δt_f about the terminal time t_f. Hence integration of (3.4.2) from t_0 to

$t_f + \varepsilon \delta t_f$ and application of (3.4.30) yields;

$$\left\langle g(t_f), \delta t_f \right\rangle_R = \left\langle \nabla_x \theta(x(t_f,u)),\ \delta t_f f(x(t_f),u(t_f)) \right\rangle_R \qquad (3.7.3)$$

from which:

$$g(t_f) = f'(x(t_f),u(t_f))\nabla_x \theta(x(t_f,u)) \qquad (3.7.4)$$

With this result, time optimal control problems can be solved using the same procedure as with a fixed terminal time.

Summary.

In this chapter various techniques available for the solution of a TPBVP were briefly discussed. Considering the inherent difficulties associated with solving the TPBVP's resulting from optimal control problems, it was decided to use a scheme which computes a sequence of controls converging to the optimal controls under appropriate conditions, namely the CGD method. The gradient of the cost functional for general piecewise continuous input controls was derived and an algorithm outlining the steps involved in computing optimal controls by the CGD was given. With these results it is now possible to proceed with the solution of specific satellite optimal attitude control problems.

Satellite Optimal Attitude Control.

In this chapter, a variety of satellite attitude control problems are investigated in an effort to study certain controllability aspects. For example, in instances where one or more of the flywheels are defective, are the satellite body dynamics controllable to the origin? Or, it may be interesting to know how much coupling there is amongst the controllers on the three axes. This may lead to a characterization of the optimal controls in certain instances and possibly to the generation of simple feedback controls, giving results approaching those obtained with the optimal controls.

Whecon Satellite. For numerical values, the physical parameters of the Whecon system [26] have been used in this research. Throughout the sequel these quantities remain unchanged. They are the moments of inertia;

$$I_x = 645 \text{ slug-ft}^2$$

$$I_y = 100 \text{ slug-ft}^2$$

$$I_z = 669 \text{ slug-ft}^2$$

and the orbital rate:

$$\omega_0 = 7.29 \times 10^{-5} \text{ rad/s}$$

Satellite Attitude Controllability: The determination of the controllability of a given nonlinear system is generally

a difficult task. No attempt is made here to give a theoretical demonstration of controllability for the problems solved. Conclusions are based wholly on computational experiments.

4.1 Reaction Jet Control.

Two problems using only reaction jet control are investigated first. In practice, reaction jets are considered to be on-off devices. However, for the first problem considered the assumption that the jets have a "variable throttle," implies the capability of continuous variations. This may not be a totally unrealistic approach depending on the operation principle of the jets used.

The sole use of reaction jets is not recommended because of the jets' limited capacity. A prolonged use would inevitably result in a short lifetime expectancy, unless prohibitively large fuel storage tanks were used.

Problem 4.1.A (Fixed Terminal Time): Consider the problem of driving the satellite body rates p, q, r and Euler angles ϕ, θ, ψ to the origin after an initial disturbance, using only reaction jets. From equation (2.2.7) an optimal control problem can be formulated as follows:

$$\text{minimize } J = \left[I + \tfrac{1}{2} R(p^2+q^2+r^2+\phi^2+\theta^2+\psi^2) \right]\Big|_{t=t_f}$$
$$\text{with } u \varepsilon U^3$$

subject to (4.1.1) for $t\varepsilon[0,t_f]$, where:

$$\begin{aligned}
\dot{p} &= A_1 r + A_2 qr + u_1 \\
\dot{q} &= B_1 pr + u_2 \\
\dot{r} &= C_1 p + C_2 qp + u_3 \\
\dot{\phi} &= p \\
\dot{\theta} &= q \\
\dot{\psi} &= r \\
\dot{I} &= \tfrac{1}{2}\lambda_1(p^2+q^2+r^2+\phi^2+\theta^2+\psi^2) + \tfrac{1}{2}\lambda_2(u_1^2+u_2^2+u_3^2)
\end{aligned} \qquad (4.1.1)$$

At this point, it is important to emphasize that the value of R should be much larger than either of λ_1 or λ_2. This is to insure that at the terminal time, the states are near the origin. The system constants are as defined in (2.2.7). U, the set of admissible controls is the space of continuous functions, and each control is given by $u_1 = T_x/I_x$, $u_2 = T_y/I_y$ and $u_3 = T_z/I_z$. All the initial conditions are assumed known for (4.1.1). This problem is stated in the form outlined by (3.4.1) - (3.4.2), which is required for the application of the CGD procedure outlined in Chapter Three. The co-state differential equations satisfying the Hamiltonian system are described by;

$$\dot{\xi} = -f_x'(x,u)\xi \qquad (4.1.2)$$

with terminal conditions*:

$$\xi(t_f) = \nabla_x \Theta(x(t_f)) \qquad (4.1.3)$$

(*)Note this is also the traversality condition mentioned in Chapter Two.

In a more explicit form, these are;

$$\begin{aligned}
\dot{\xi}_1 &= -(B_1 r\xi_2 + C_1\xi_3 + C_2 q\xi_3 + \xi_4 + \lambda_1 p\xi_7) \\
\dot{\xi}_2 &= -(A_2 r\xi_1 + C_2 p\xi_3 + \xi_5 + \lambda_1 q\xi_7) \\
\dot{\xi}_3 &= -(A_1\xi_1 + A_2 q\xi_1 + B_2 p\xi_2 + \xi_6 + \lambda_1 r\xi_7) \\
\dot{\xi}_4 &= -\lambda_1\phi\xi_7 \\
\dot{\xi}_5 &= -\lambda_1\theta\xi_7 \\
\dot{\xi}_6 &= -\lambda_1\psi\xi_7 \\
\dot{\xi}_7 &= 0
\end{aligned} \tag{4.1.4}$$

with the following terminal conditions:

$$\begin{aligned}
&\xi_1(t_f) = Rp(t_f) \quad \xi_4(t_f) = R\phi(t_f) \quad \xi_7(t_f) = 1 \\
&\xi_2(t_f) = Rq(t_f) \quad \xi_5(t_f) = R\theta(t_f) \\
&\xi_3(t_f) = Rr(t_f) \quad \xi_6(t_f) = R\psi(t_f)
\end{aligned} \tag{4.1.5}$$

Since the co-states are continuous functions of time, it can easily be verified using PMP that the optimal controls are indeed continuous funtions of time. Hence, it follows that the gradient of the cost function in the space of contiruous input functions is obtained by applying (3.4.53) and is given by:

$$\begin{aligned}
g_1(u) &= \xi_1 + \lambda_2\xi_7 u_1 \\
g_2(u) &= \xi_2 + \lambda_2\xi_7 u_2 \\
g_3(u) &= \xi_3 + \lambda_2\xi_7 u_3
\end{aligned} \tag{4.1.6}$$

The Fortran code used to solve this problem is listed in Appendix C. Figures 4.1.1-4.1.12 illustrate the rate of convergence to the minimum cost as well as the optimal controls

and state trajectories for three different cases:

Case 1;

$p(0) = .02$ rad/s, $q(0) = .04$ rad/s, $r(0) = .06$ rad/s,

$\phi(0) = .00$ rad, $\theta(0) = .00$ rad, $\psi(0) = .00$ rad,

$I(0) = 0.$, $R = 4x10^4$, $\lambda_1 = 10$, (4.1.7)

$\lambda_2 = 1$, $t_f = 10s.$

Case 2;

$p(0) = .00$ rad/s, $q(0) = .00$ rad/s, $r(0) = .00$ rad/s,

$\phi(0) = .02$ rad, $\theta(0) = .04$ rad, $\psi(0) = .06$ rad,

$I(0) = 0.$, $R = 4x10^4$, $\lambda_1 = 10$, (4.1.8)

$\lambda_2 = 1$, $t_f = 10s.$

Case 3:

$p(0) = .01$ rad/s, $q(0) = .03$ rad/s, $r(0) = .05$ rad/s

$\phi(0) = .01$ rad, $\theta(0) = .03$ rad, $\psi(0) = .05$ rad,

$I(0) = 0.$, $R = 4x10^4$, $\lambda_1 = 10$, (4.1.9)

$\lambda_2 = 1$, $t_f = 10s.$

From the graphical results, it appears that the system is completely controllable within a small distance of the origin using only variable reaction jets.

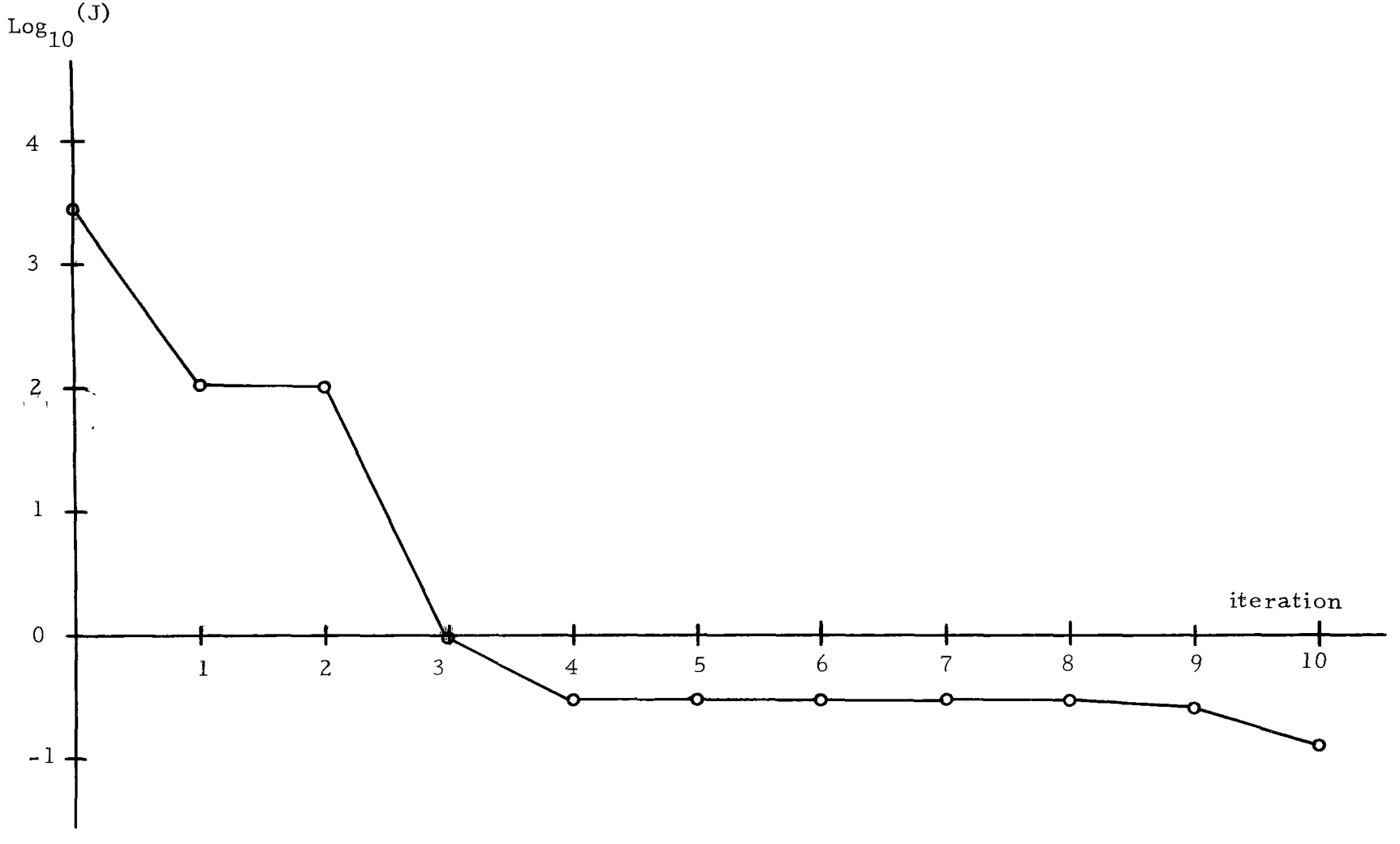


Fig. 4.1.1 $\text{Log}_{10}^{(J)}$ for Problem 4.1.A Case 1

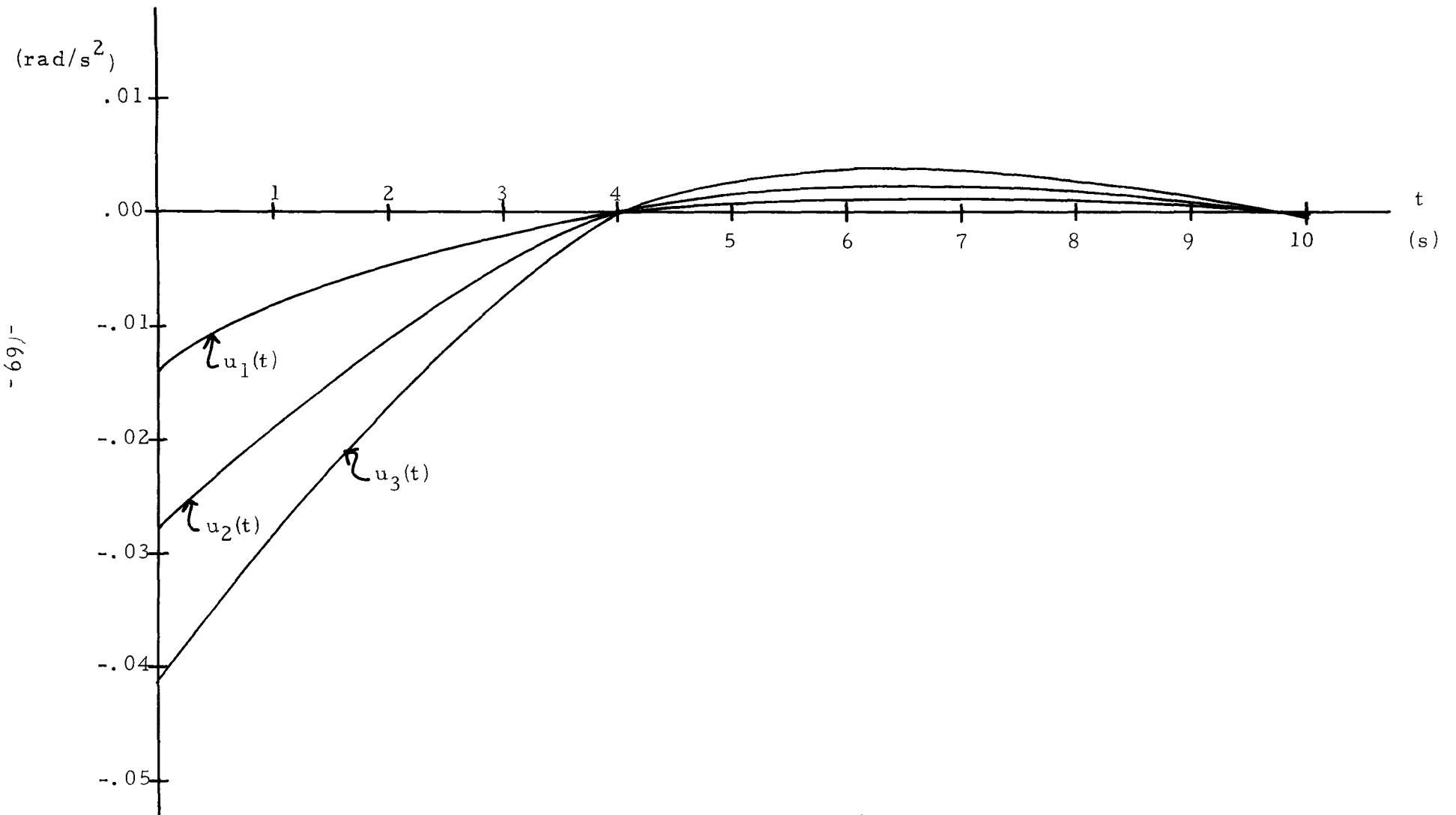


Fig. 4.1.2 Controls for Problem 4.1.A Case 1

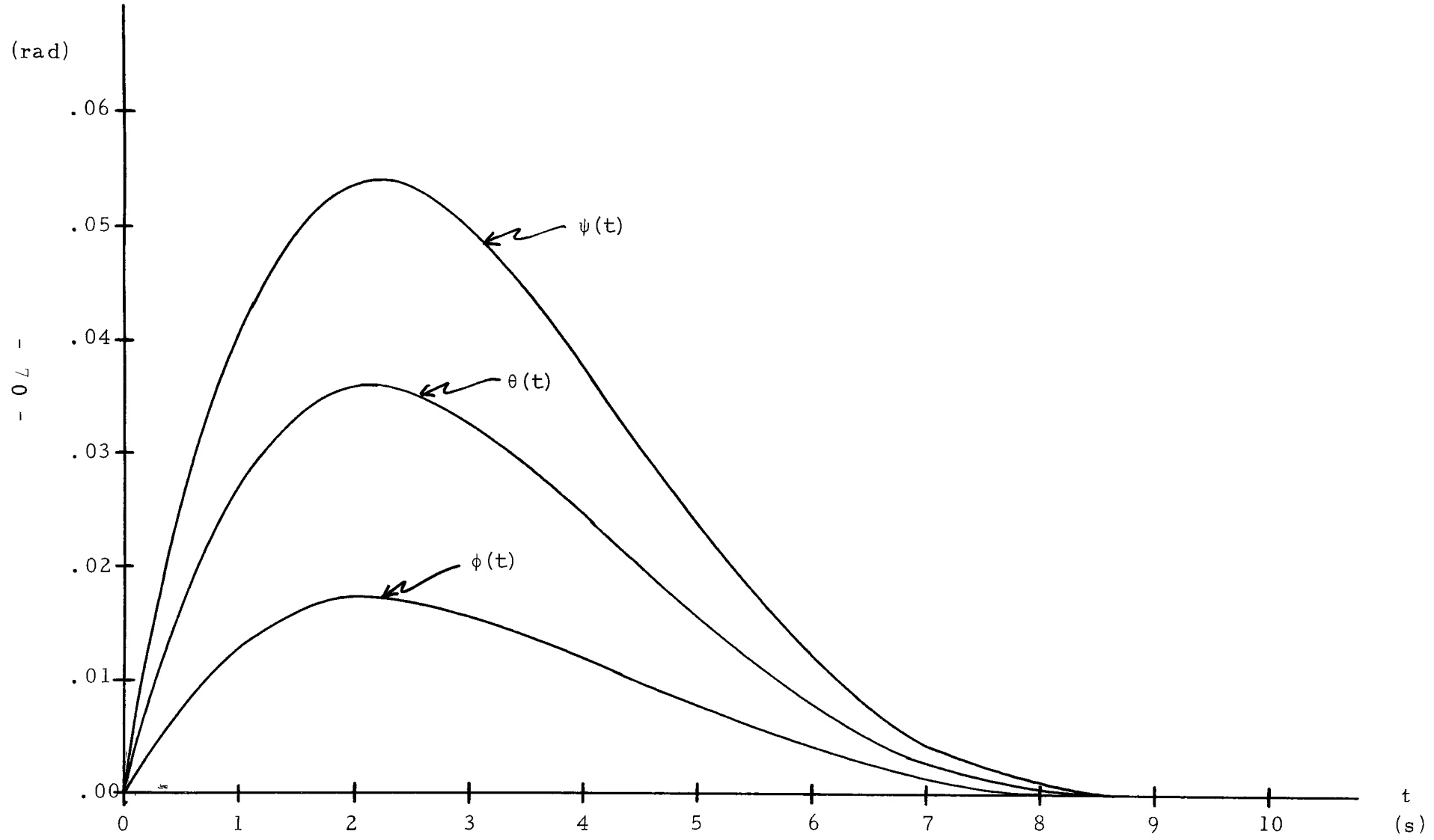


Fig. 4.1.3 Euler Angles for Problem 4.1.A Case 1

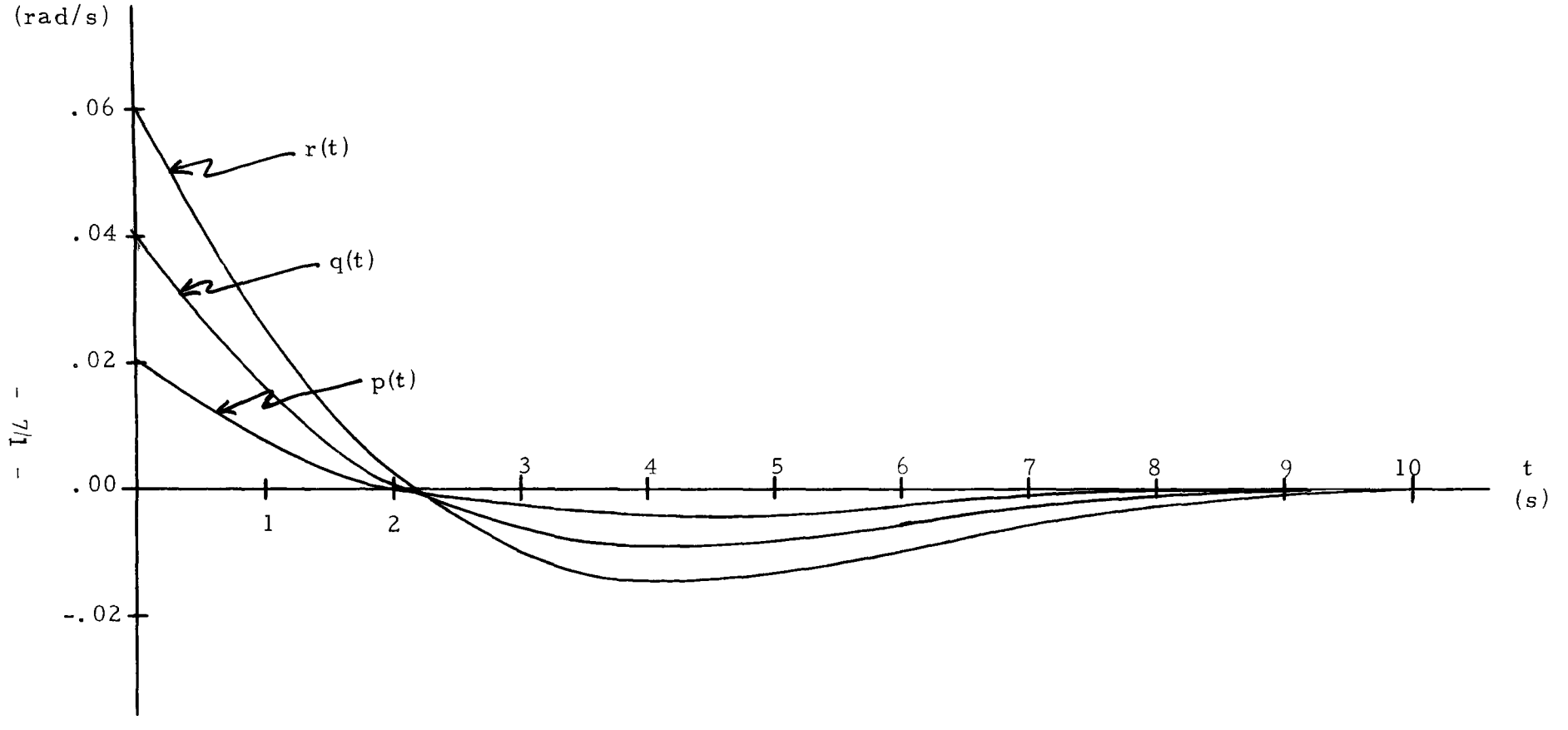


Fig. 4.1.4 Body Rates for Problem 4.1.A Case 1

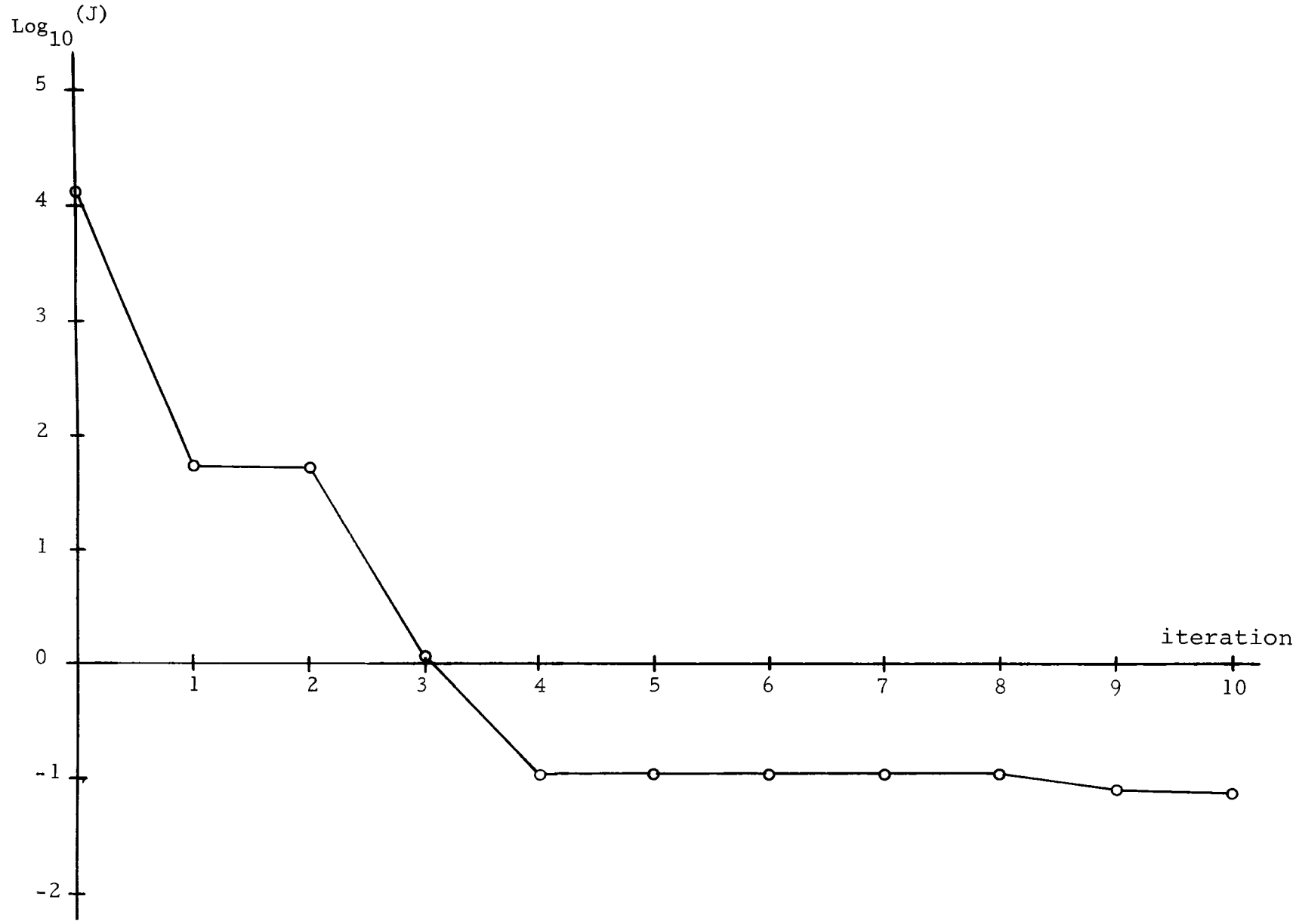


Fig. 4.1.5 $\text{Log}_{10}^{(J)}$ for Problem 4.1.A Case 2

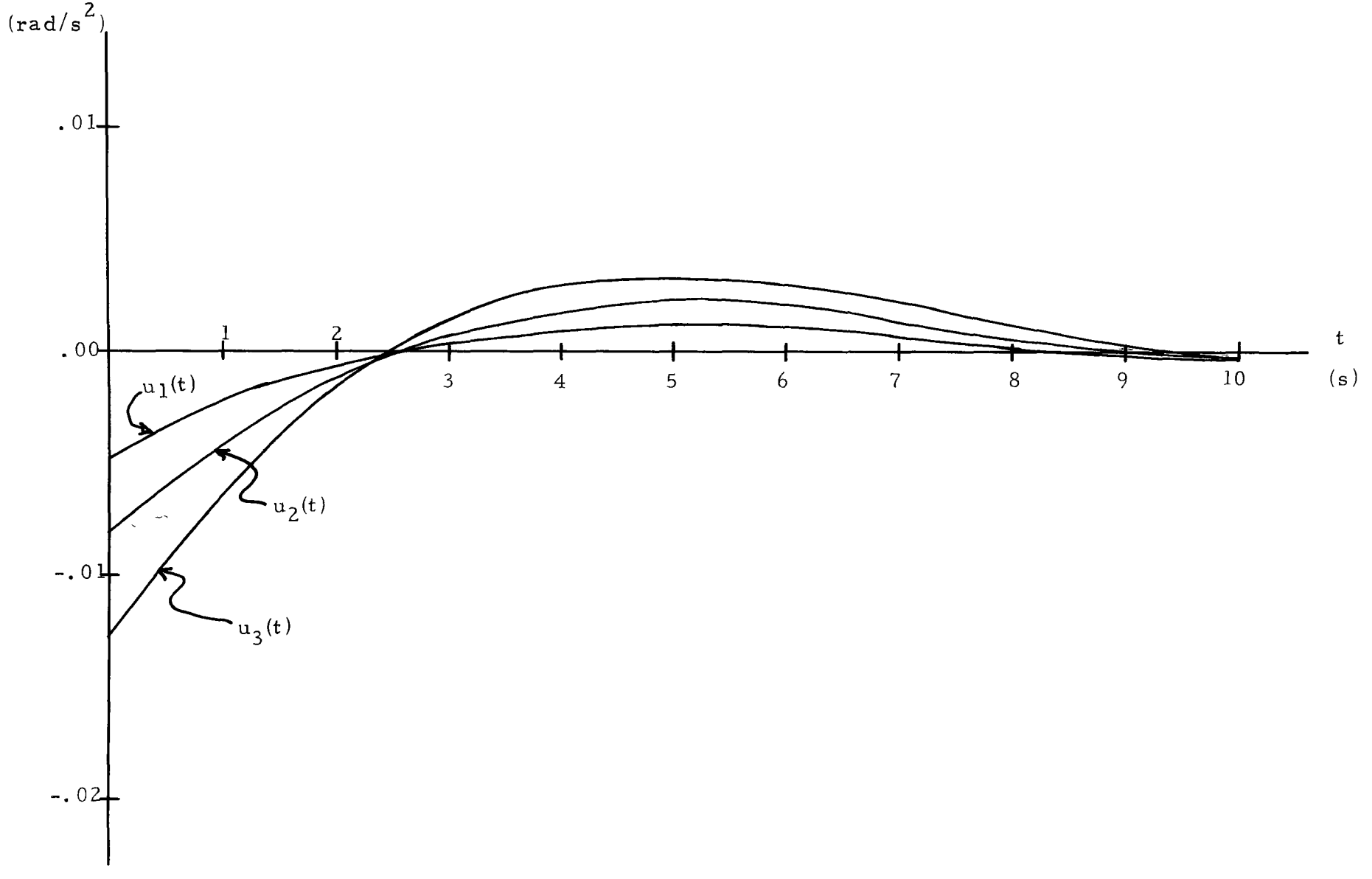


Fig. 4.1.6 Controls for Problem 4.1.A Case 2

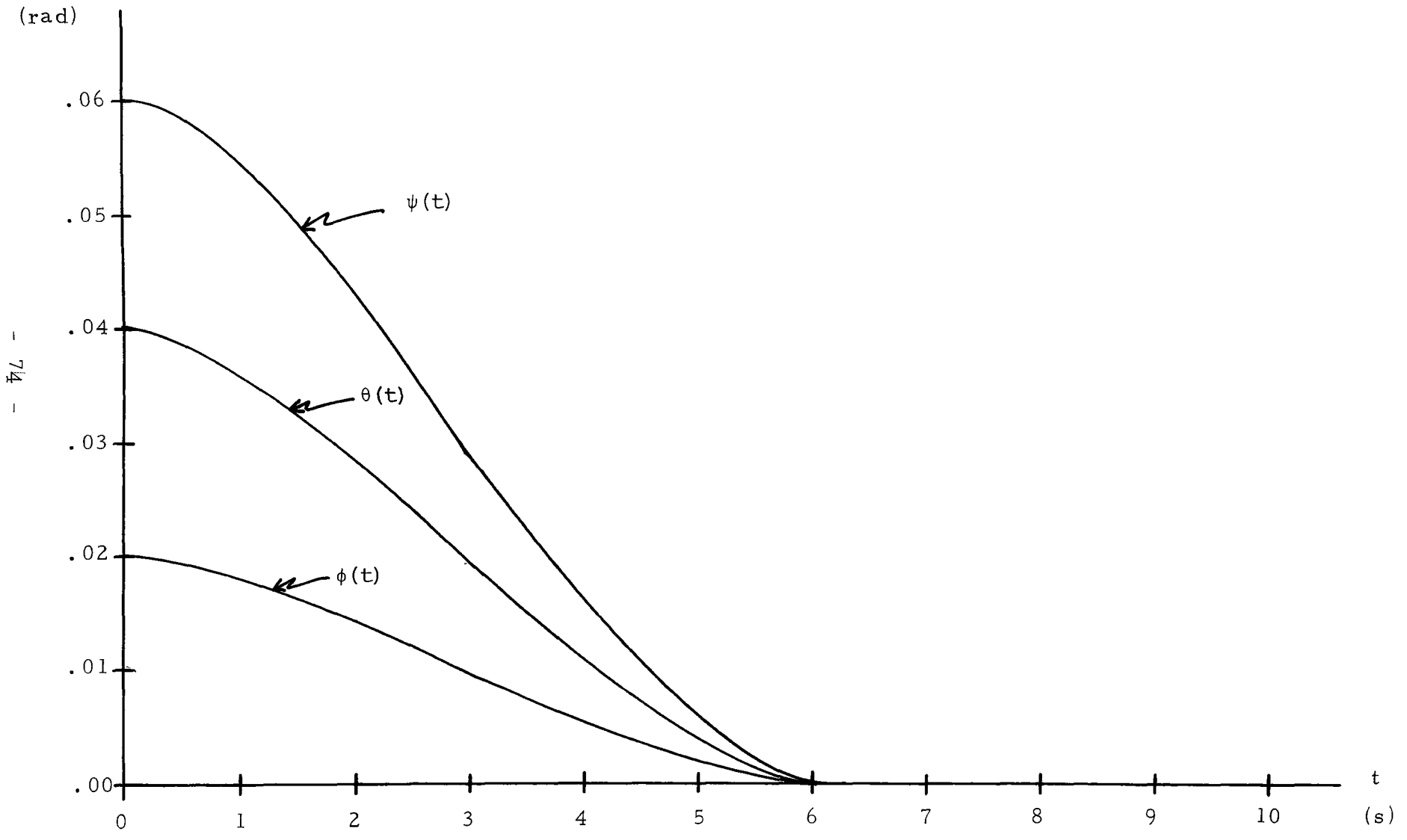


Fig. 4.1.7 Euler Angles for Problem 4.1.A Case 2

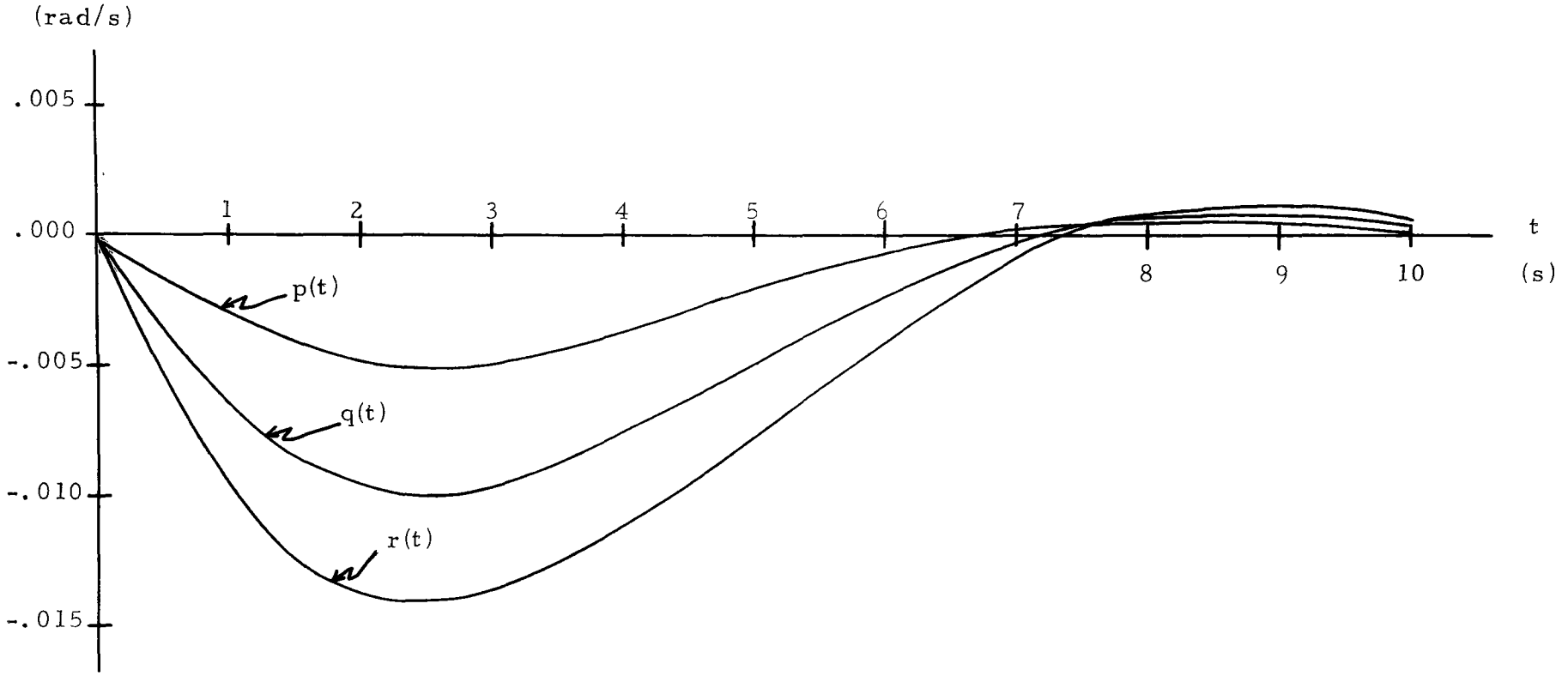


Fig. 4.1.8 Body Rates for Problem 4.1.A Case 2

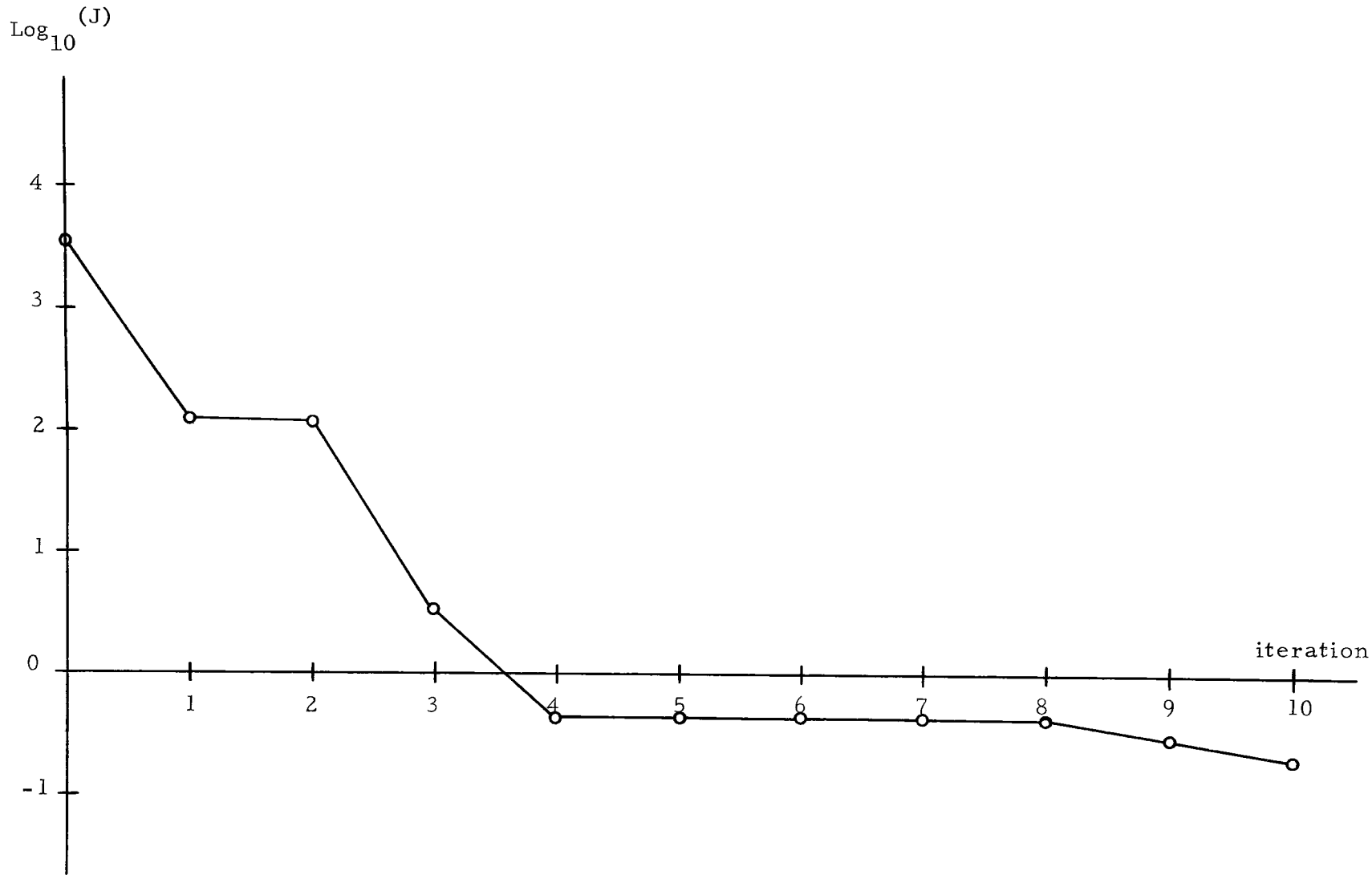


Fig. 4.1.9 $\text{Log}_{10}{}^{(J)}$ for Problem 4.1.A Case 3

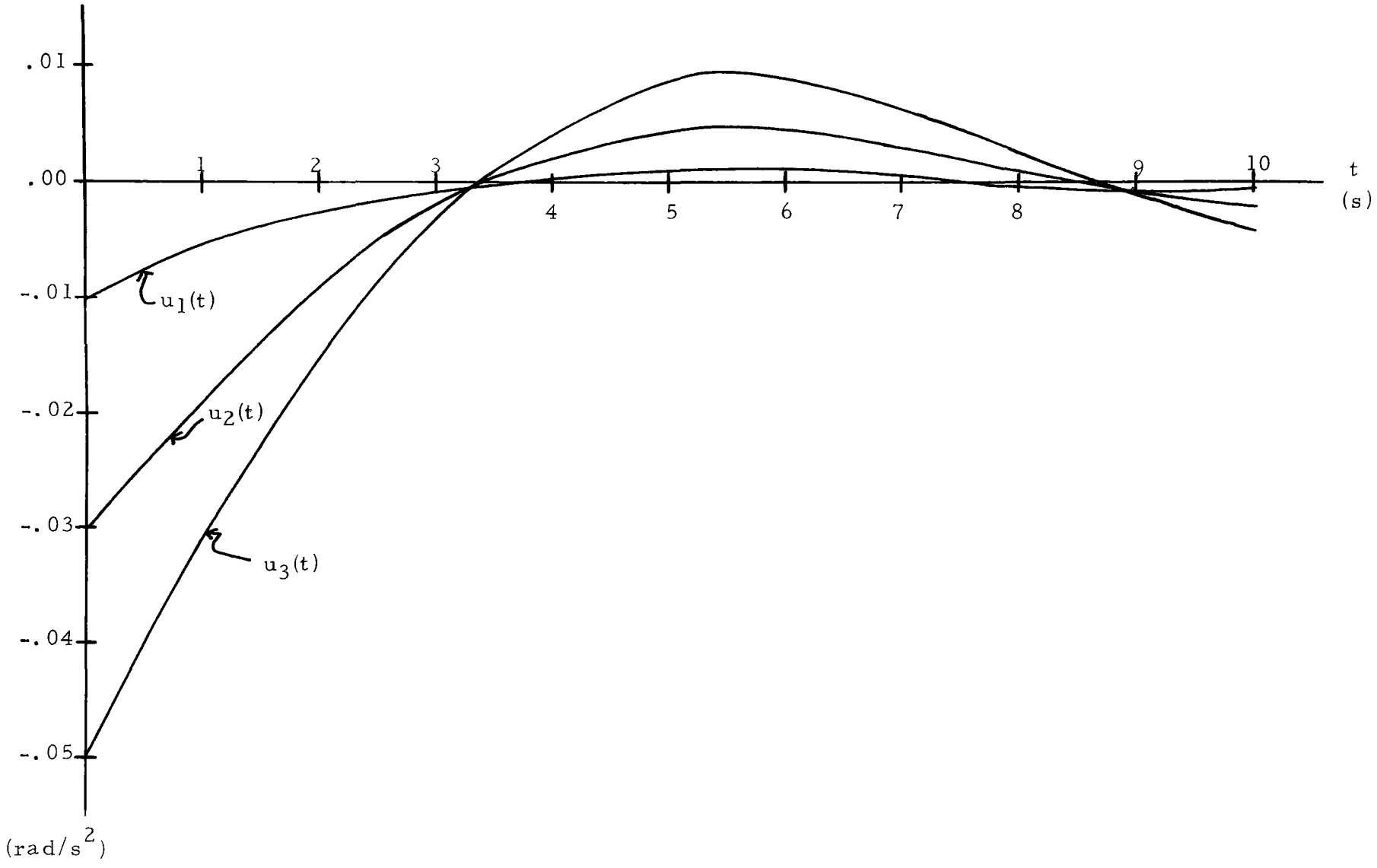


Fig. 4.1.10 Controls for Problem 4.1.A Case 3

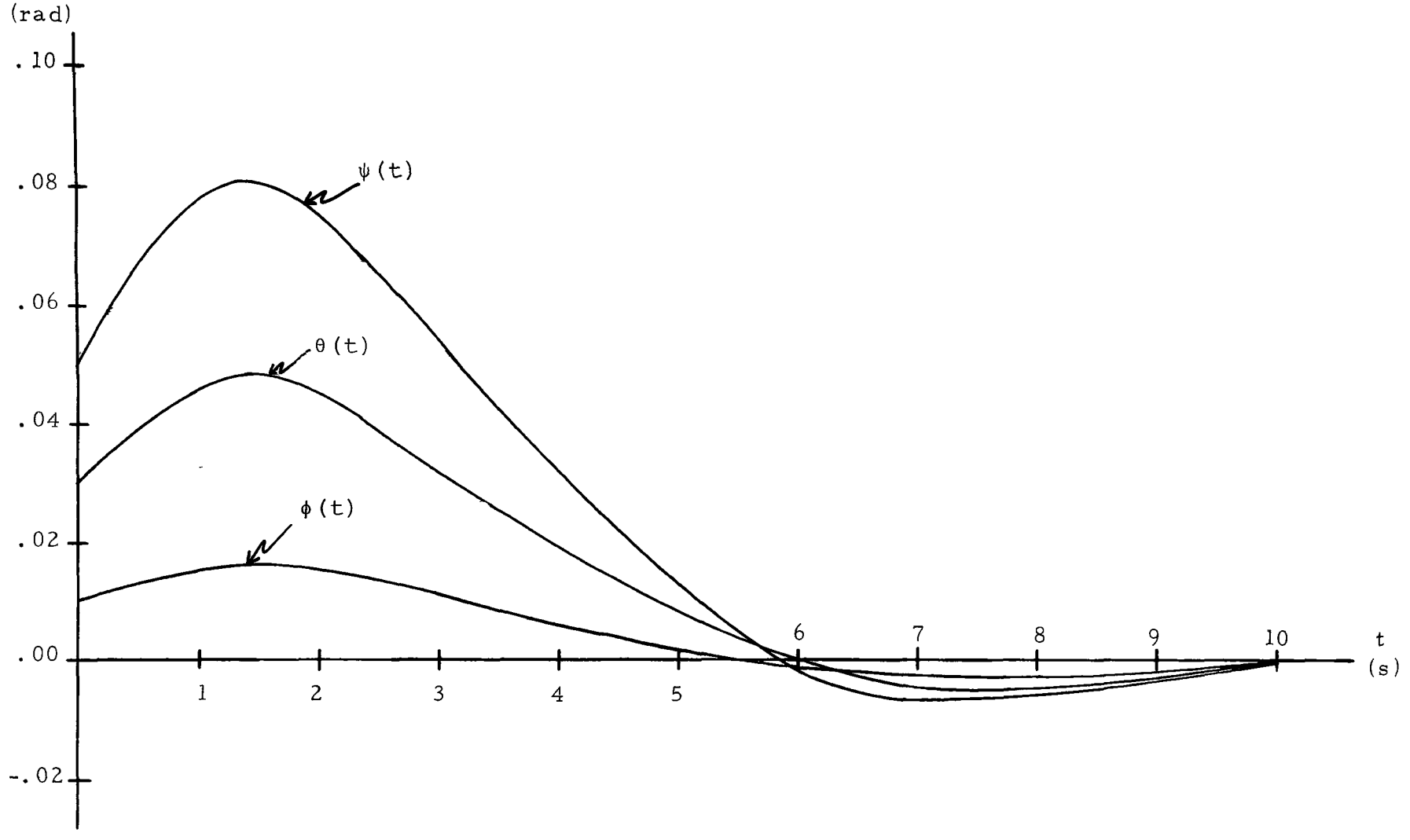


Fig. 4.1.11 Euler Angles for Problem 4.1.A Case 3

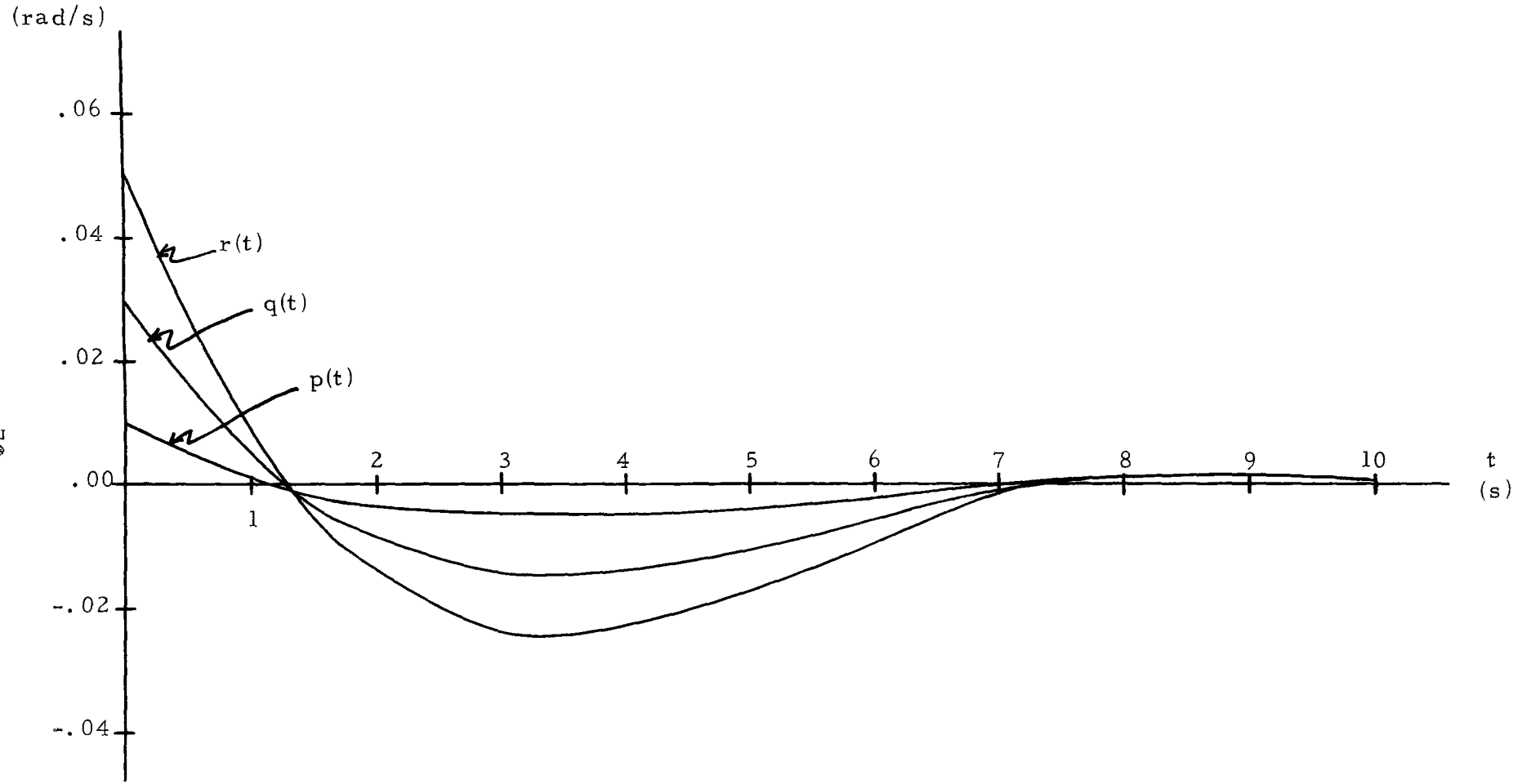


Fig. 4.1.12 Body Rates for Problem 4.1.A Case 3

Problem 4.1.B (Free Terminal Time): As noted earlier reaction jets seldom have a throttle, rather they are on-off devices. This type of control is optimal for a class of time optimal control problems.

Consider the following time optimal problem.

$$\underset{u\in U^3}{\text{minimize}}\ J = \left[I+\tfrac{1}{2}R(p^2+q^2+r^2+\phi^2+\theta^2+\psi^2)\right]\Big|_{t=t_f}$$

subject to (4.1.10) for $t\in[0,t_f]$, where;

$$\begin{aligned}
\dot{p} &= A_1 r + A_2 qr + u_1\\
\dot{q} &= B_1 pr + u_2\\
\dot{r} &= C_1 p + C_2 qp + u_3\\
\dot{\phi} &= p\\
\dot{\theta} &= q\\
\dot{\psi} &= r\\
\dot{I} &= \tfrac{1}{2}\lambda_1(p^2+q^2+r^2+\phi^2+\theta^2+\psi^2)
\end{aligned} \qquad (4.1.10)$$

with $U=\left\{-u_{max},u_{max}\right\}$. As in 4.1.A, the controls are given by $u_1=T_x/I_x$, $u_2=T_y/I_y$ and $u_3=T_z/I_z$, with $T_{\alpha max}=u_{max}I_\alpha$, $\alpha=x, y, z$. The co-state differential equations with their terminal conditions are the same as for problem 4.1.A and are given by (4.1.4) and (4.1.5).

Using PMP it can easily be verified that the optimal controls for this problem are of a bang-bang nature. To apply the CGD method it is necessary to know the number of discontinuities in each control. To determine this number, the

discontinuous controls were approximated by continuous functions such as arctangents and then the problem was solved by the method of adjoints [12]. It was found that one switching occurred for each control. Hence, the gradient in the space of switching times is given by applying (3.4.54) with the following result in our case;

$$\begin{aligned} g_1(u) &= -2\, u_{max}\, \xi_1(t_1) \\ g_2(u) &= -2\, u_{max}\, \xi_2(t_2) \\ g_3(u) &= -2\, u_{max}\, \xi_3(t_3) \end{aligned} \tag{4.1.11}$$

where t_i, is the switching time corresponding to the control u_i, i=1, 2, 3 respectively. Furthermore, since this is a free terminal time problem, it is necessary to compute the gradient of the terminal time. From equation (3.7.4) it follows that:

$$g(t_f) = R\Big[\phi p+\theta q+\psi r+(A_1+C_1)pr + (A_2+B_1+C_2)pqr\Big]\Big|_{t=t_f} + \tfrac{1}{2}\lambda_1(p^2+q^2+r^2+\phi^2+\theta^2+\psi^2)\Big|_{t=t_f} \tag{4.1.12}$$

The CGD algorithm described in section 3.5 was applied for two different choices of terminal time; namely 10 and 11 seconds. In both cases, after convergence, the optimal terminal time was 9.45 seconds. The initial conditions were taken to be;

$$\begin{aligned} &p(0) = .1 \text{ rad/s}, \quad q(0) = .1 \text{ rad/s}, \quad r(0) = .1 \text{ rad/s}, \\ &\phi(0) = .1 \text{ rad}, \quad \theta(0) = .1 \text{ rad}, \quad \psi(0) = .1 \text{ rad}, \\ &I(0) = 0 \end{aligned} \tag{4.1.13}$$

and the initial guess at the switching times was:

$$t_1 = 1.0 \text{ s}, \quad t_2 = 1.5 \text{ s}, \quad t_3 = 2.0 \text{ s} \qquad (4.1.14)$$

From Figures 4.1.13-4.1.16, it appears that even with a free terminal time it is rather difficult to control the system to the origin using only on-off jets. The proximity to the origin to which the system can be driven is rather sensitive to the switching time. Practically though, it is meaningless to determine a switching time to five decimal places. Thus it can be concluded that jets alone are not a very good choice to achieve a high degree of resolution, although they may be used to correct large perturbations in the angular body rates; it is best to use them for flywheel desaturation.

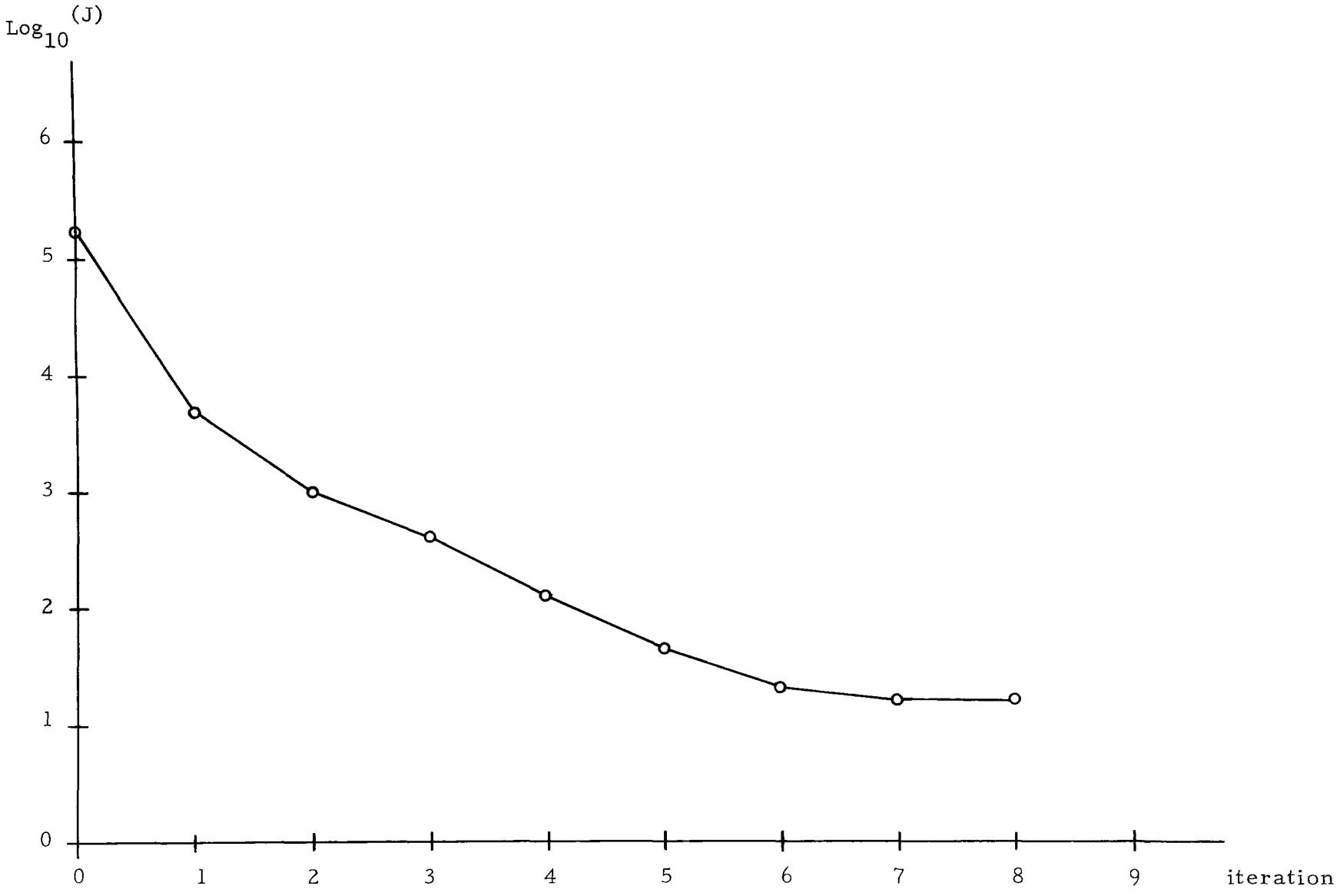


Fig. 4.1.13 $\text{Log}_{10}(J)$ for Problem 4.1.B

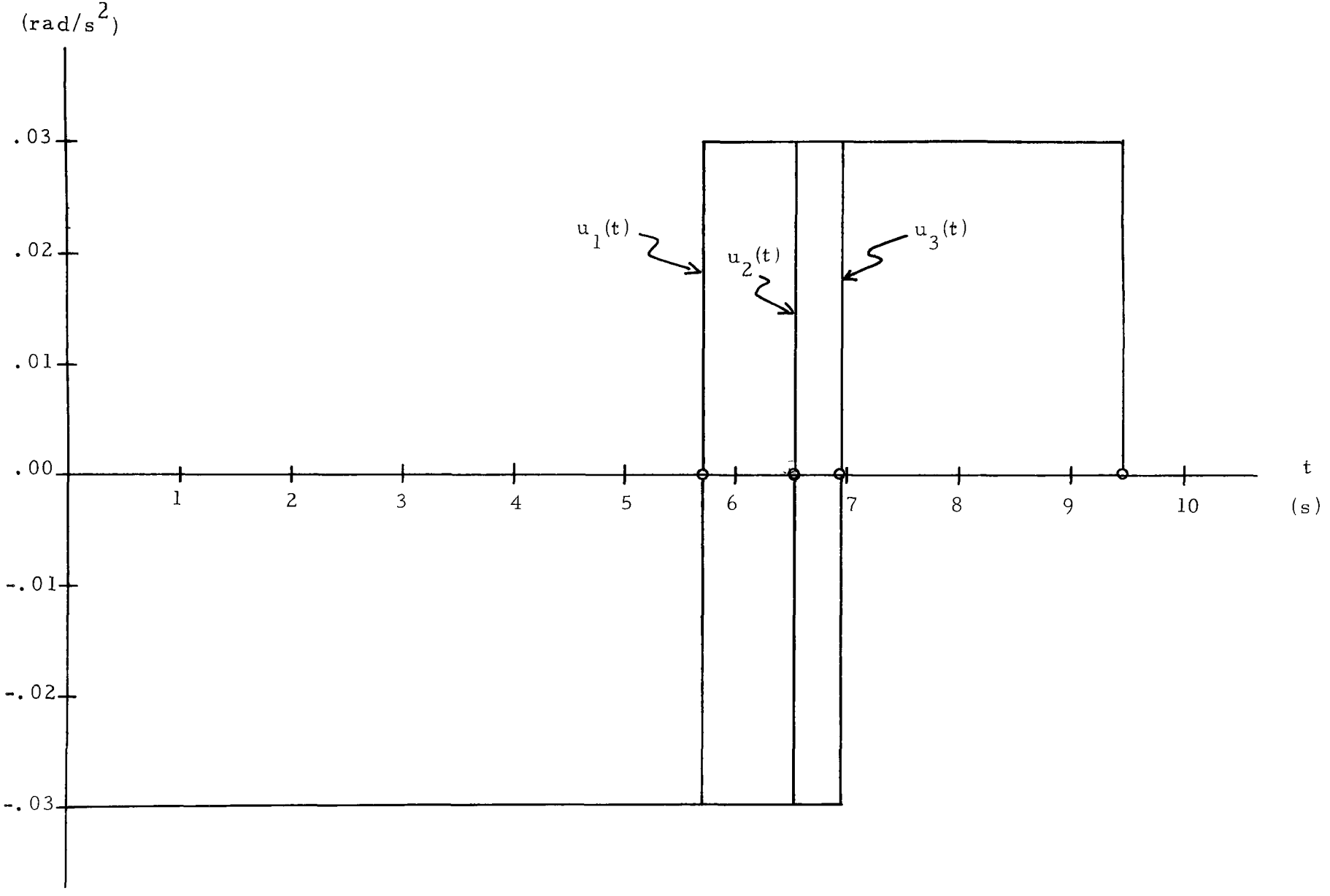


Fig. 4.1.14 Controls for Problem 4.1.B

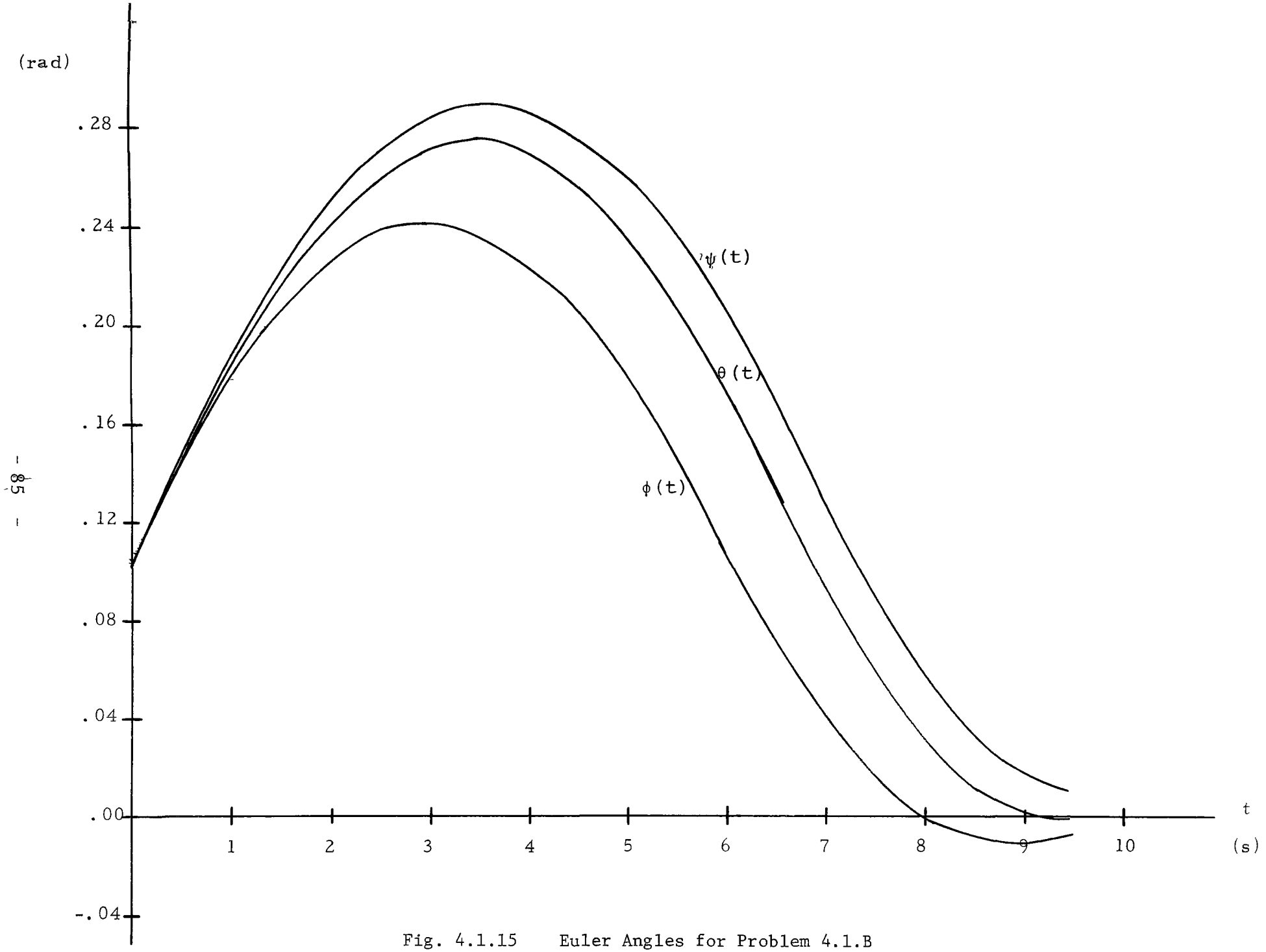


Fig. 4.1.15 Euler Angles for Problem 4.1.B

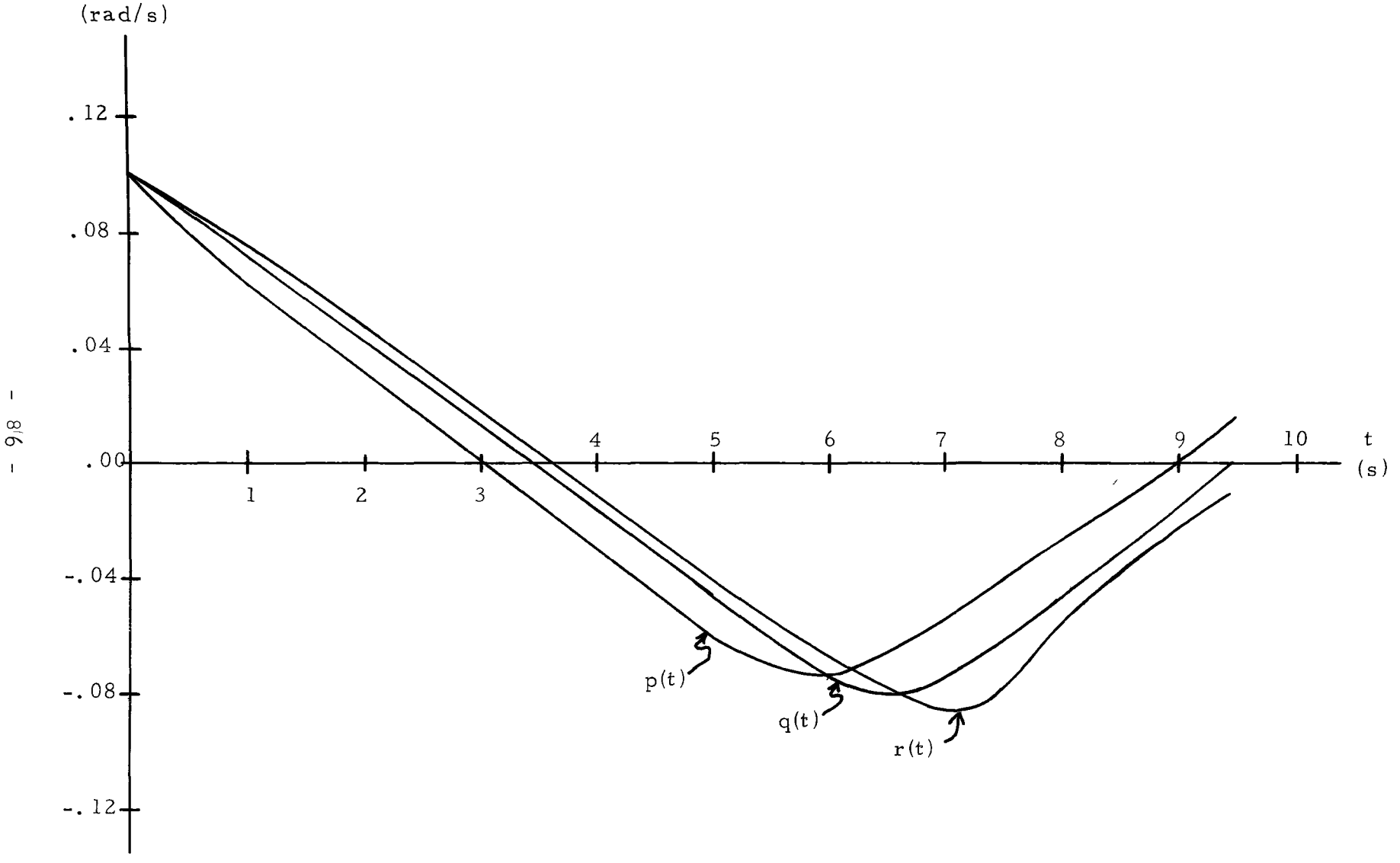


Fig. 4.1.16 Body Rates for Problem 4.1.B

4.2 Flywheel Attitude Control.

In the previous section it was shown that reaction jets alone will not, in general, produce the required resolution. They should be complemerted by some type of momentum transfer devices such as flywheels or gyrotorquers. Flywheels will be considered in this section under two modes of operation -saturating and non-saturating.

In the non-saturating mode, the flywheels are stationary at the initial time. They are then accelerated in a prescribed fashion to correct the disturbance in such a way, that when the disturbance has been removed the flywheels are again stationary. This type of control obviously cannot correct any initial disturbance in the angular body rates, because there is no net change in angular momentum for the overall system. However, as will be shown this type of control has interesting features. For example, a disturbance in one of the Euler angles can be corrected by one flywheel on a particular body axis. Also, fixed terminal time feedback controls are easily synthesized, as demonstrated in the next chapter. For the sequel of this chapter, a fixed terminal time is assumed.

Problem 4.2.A Flywheel Control (Non-Saturating): In this problem, non-saturating flywheel control is used to correct initial conditions in the Euler angles ϕ, θ and ψ. The

optimal control problem is formulated as follows;

$$\underset{u\varepsilon U^3}{\text{minimize}}\ J = \left[I+\tfrac{1}{2}R_1(p^2+q^2+r^2+\phi^2+\theta^2+\psi^2) + \tfrac{1}{2}R_2(\Omega_x^2+\Omega_y^2+\Omega_z^2) \right]\Big|_{t=t_f}$$

subject to (4.2.1) for $t\varepsilon[0,t_f]$, where;

$$\begin{aligned}
\dot{p} &= A_1 r + A_2 qr + A_3 u_1 + A_4 \Omega_y r + A_5 \Omega_z q \\
\dot{q} &= B_1 pr + B_2 \Omega_x r + B_3 u_2 + B_4 \Omega_z p \\
\dot{r} &= C_1 p + C_2 qp + C_3 \Omega_x q + C_4 \Omega_y p + C_5 u_3 \\
\dot{\phi} &= p \\
\dot{\theta} &= q \\
\dot{\psi} &= r \\
\dot{\Omega}_x &= u_1 \\
\dot{\Omega}_y &= u_2 \\
\dot{\Omega}_z &= u_3 \\
\dot{I} &= \tfrac{1}{2}\lambda_1(p^2+q^2+r^2+\phi^2+\theta^2+\psi^2) + \tfrac{1}{2}\lambda_2(u_1^2+u_2^2+u_3^2)
\end{aligned} \qquad (4.2.1)$$

with initial conditions of the form;

$$\begin{array}{llll}
p(0) = 0. & \phi(0) = \phi_0 & \Omega_x(0) = 0. & I(0) = 0 \\
q(0) = 0. & \theta(0) = \theta_0 & \Omega_y(0) = 0. & \\
r(0) = 0. & \psi(0) = \psi_0 & \Omega_z(0) = 0. &
\end{array} \qquad (4.2.2)$$

U, the set of admissible controls, consists of the continuous functions of time on $[0,t_f]$. The co-state differential equations for the above are:

$$\begin{aligned}
\dot{\xi}_1 &= -(B_1 r\xi_2+B_4\Omega_z\xi_2+C_1\xi_3+C_2 q\xi_3+C_4\Omega_y\xi_3+\xi_4+\lambda_1 p\xi_{10}) \\
\dot{\xi}_2 &= -(A_2 r\xi_1+A_5\Omega_z\xi_1+C_2 p\xi_3+C_3\Omega_x\xi_3+\xi_5+\lambda_1 q\xi_{10}) \\
\dot{\xi}_3 &= -(A_1\xi_1+A_2 q\xi_1+A_4\Omega_y\xi_1+B_1 p\xi_2+B_2\Omega_x\xi_2+\xi_6+\lambda_1 r\xi_{10}) \\
\dot{\xi}_4 &= -\lambda_1\phi\xi_{10} \\
\dot{\xi}_5 &= -\lambda_1\theta\xi_{10}
\end{aligned}$$

$$\dot{\xi}_6 = -\lambda_1\psi\xi_{10}$$

$$\dot{\xi}_7 = -(B_2 r\xi_2 + C_2 q\xi_3) \qquad (4.2.3)$$

$$\dot{\xi}_8 = -(A_4 r\xi_1 + C_4 p\xi_3)$$

$$\dot{\xi}_9 = -(A_5 q\xi_1 + B_4 p\xi_2)$$

$$\dot{\xi}_{10} = 0$$

and their terminal conditions are:

$$\xi_1(t_f) = R_1 p(t_f),\quad \xi_2(t_f) = R_1 q(t_f),\quad \xi_3(t_f) = R_1 r(t_f),\quad \xi_{10}(t_f) = 1,$$

$$\xi_4(t_f) = R_1\phi(t_f),\quad \xi_5(t_f) = R_1\theta(t_f),\quad \xi_6(t_f) = R_1\psi(t_f), \qquad (4.2.4)$$

$$\xi_7(t_f) = R_2\Omega_x(t_f),\quad \xi_8(t_f) = R_2\Omega_y(t_f),\quad \xi_9(t_f) = R_2\Omega_z(t_f).$$

Again, using PMP it is easily verified that the optimal controls will be continuous functions of time. Hence, the gradient of J in the space of continuous input functions is given by (3.4.53). Thus it follows that:

$$g_1(u) = A_3\xi_1 + \xi_7 + \lambda_2 u_1\xi_{10}$$

$$g_2(u) = B_3\xi_2 + \xi_8 + \lambda_2 u_2\xi_{10} \qquad (4.2.5)$$

$$g_3(u) = C_5\xi_3 + \xi_9 + \lambda_2 u_3\xi_{10}$$

All the system constants are as shown for (2.2.7), however, C_x, C_y and C_z have not yet been defined. The values chosen for these quantities depend on system constrairts such as the maximum allowable flywheel velocity and the maximum slew rate of the satellite. These quantities were assumed to be 94.25 rad/s and .00262 rad/s respectively [4]. To estimate the desirable inertia of the flywheel, the following formula

from Greensite [4] can be used;

$$C_x \Omega_{max} = I_x \Omega_{smax} \tag{4.2.6}$$

Where:

I_x: x axis body inertia;

Ω_{smax}: maximum slew rate;

Ω_{max}: maximum flywheel velocity.

Let;

$$\left.\begin{aligned} \Omega_{smax} &= .00262 \text{ rad/s} \\ \Omega_{max} &= 94.25 \text{ rad/s} \\ I_x &= 645 \text{ slug-ft}^2 \\ I_y &= 100 \text{ slug-ft}^2 \\ I_z &= 669 \text{ slug-ft}^2 \end{aligned}\right\} \tag{4.2.7}$$

Therefore:

$$\left.\begin{aligned} C_x &= .01075 \text{ slug-ft}^2 \\ C_y &= .00167 \text{ slug-ft}^2 \\ C_z &= .00115 \text{ slug-ft}^2 \end{aligned}\right\}$$

The above values of C_x, C_y and C_z will be used throughout the sequel of this research. The terminal time can be selected in the following manner. Let Δ, the maximally perturbed Euler angle be .06 rad. Then choose:

$$t_f > \frac{\Delta}{\Omega_{smax}} = \frac{.06 \text{ rad}}{2.62 \times 10^{-3} \text{ rad/s}} = 22.95 \text{ s} \tag{4.2.8}$$

Thus a terminal time of 50 seconds was assumed for the sequel unless otherwise specified. The CGD approach was used to solve this problem for the following case.

$p(0)=0.$ rad/s $\quad \phi(0)=.02$ rad $\quad \Omega_x(0)=0.$ rad/s

$q(0)=0.$ rad/s $\quad \theta(0)=.04$ rad $\quad \Omega_y(0)=0.$ rad/s

$r(0)=0.$ rad/s $\quad \psi(0)=.06$ rad $\quad \Omega_z(0)=0.$ rad/s (4.2.9)

$I(0) = 0.,\quad \lambda_1 = 5,\quad \lambda_2 = 1 \times 10^{-3},\quad R_1 = 4 \times 10^4$

$R_2 = 100,\quad t_f = 50$ s.

Figures 4.2.1-4.2.5 illustrate the results obtained. Different initial conditions were tried and found to yield similar results.

It may be observed that indeed the flywheels are behaving in a non-saturating mode. Furthermore, the maximum values of Ω_x, Ω_y and Ω_z are in correspondence with the relative amplitude of the Euler angles. The next problem will focus attention on this point.

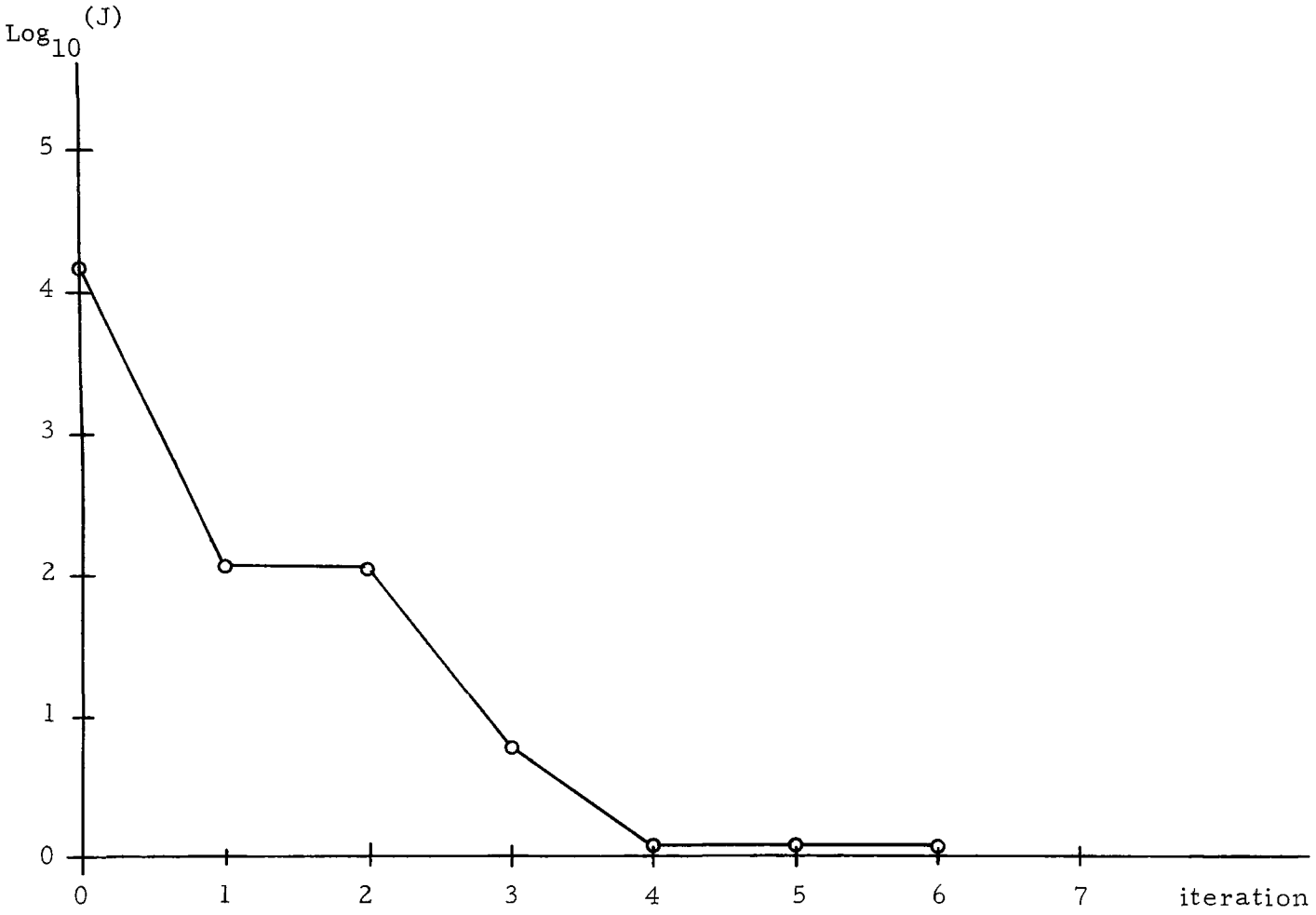


Fig. 4.2.1 $\text{Log}_{10}{}^{(J)}$ for Problem 4.2.A

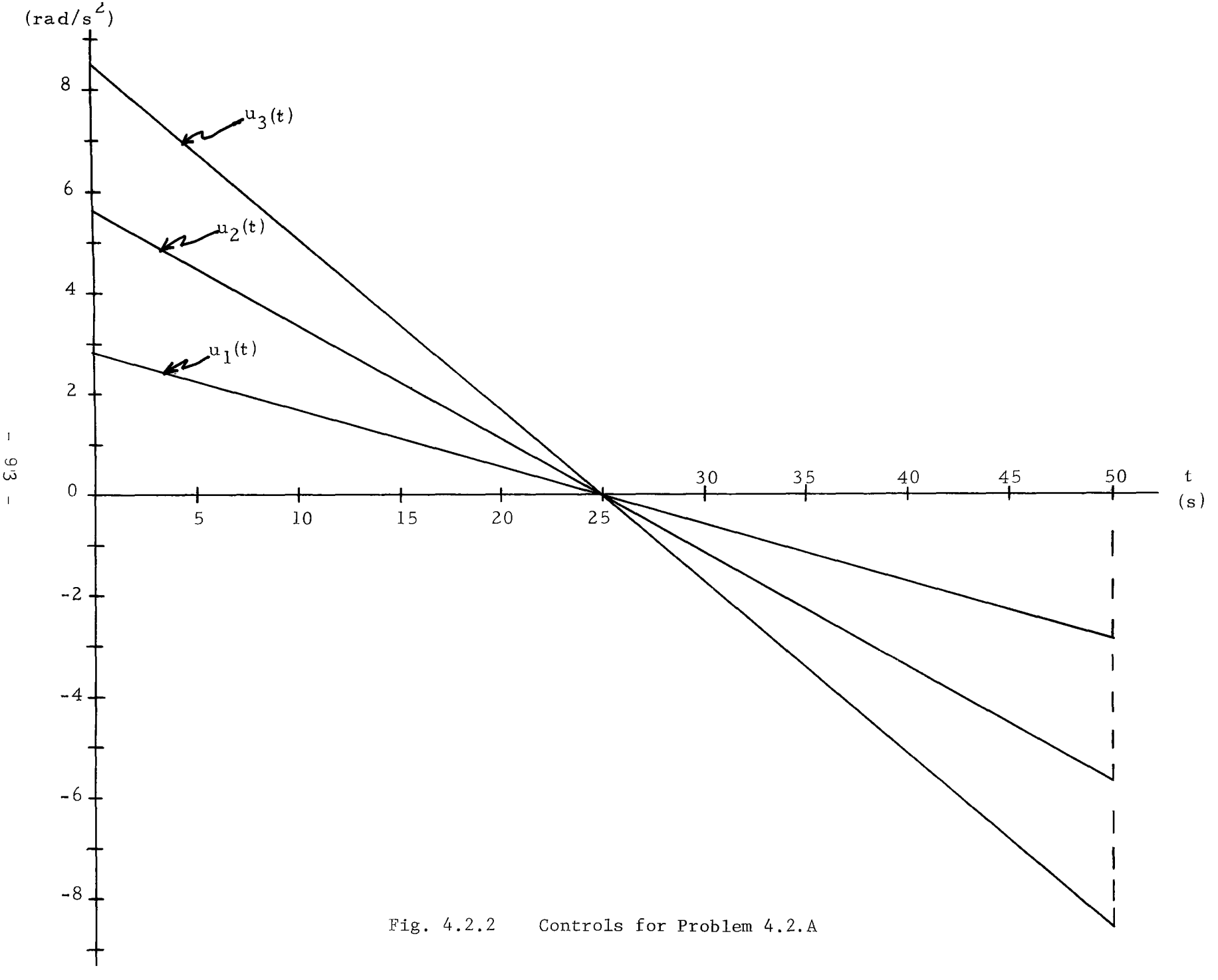


Fig. 4.2.2 Controls for Problem 4.2.A

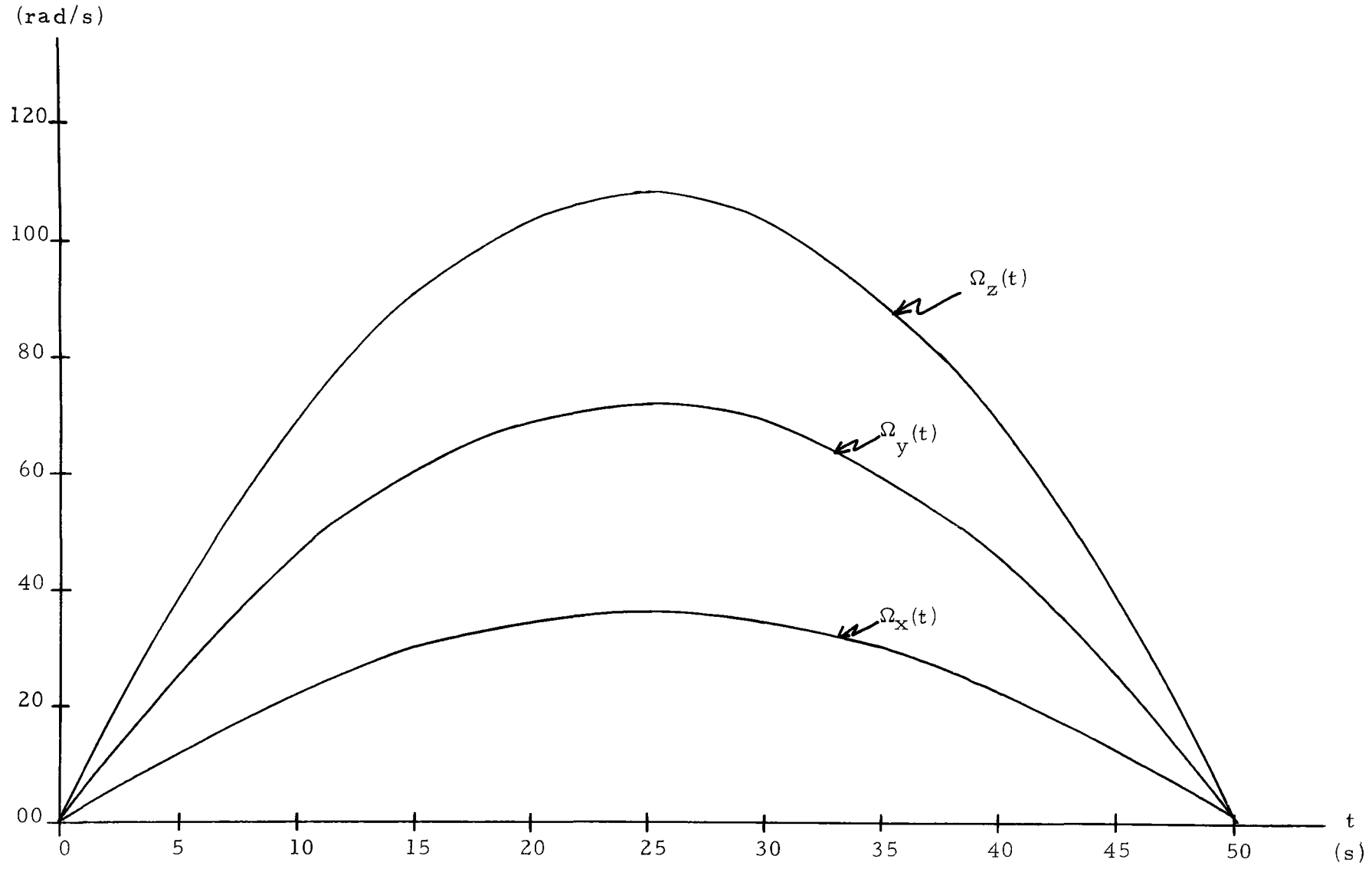


Fig. 4.2.3 Flywheel Angular Velocities for Problem 4.2.A

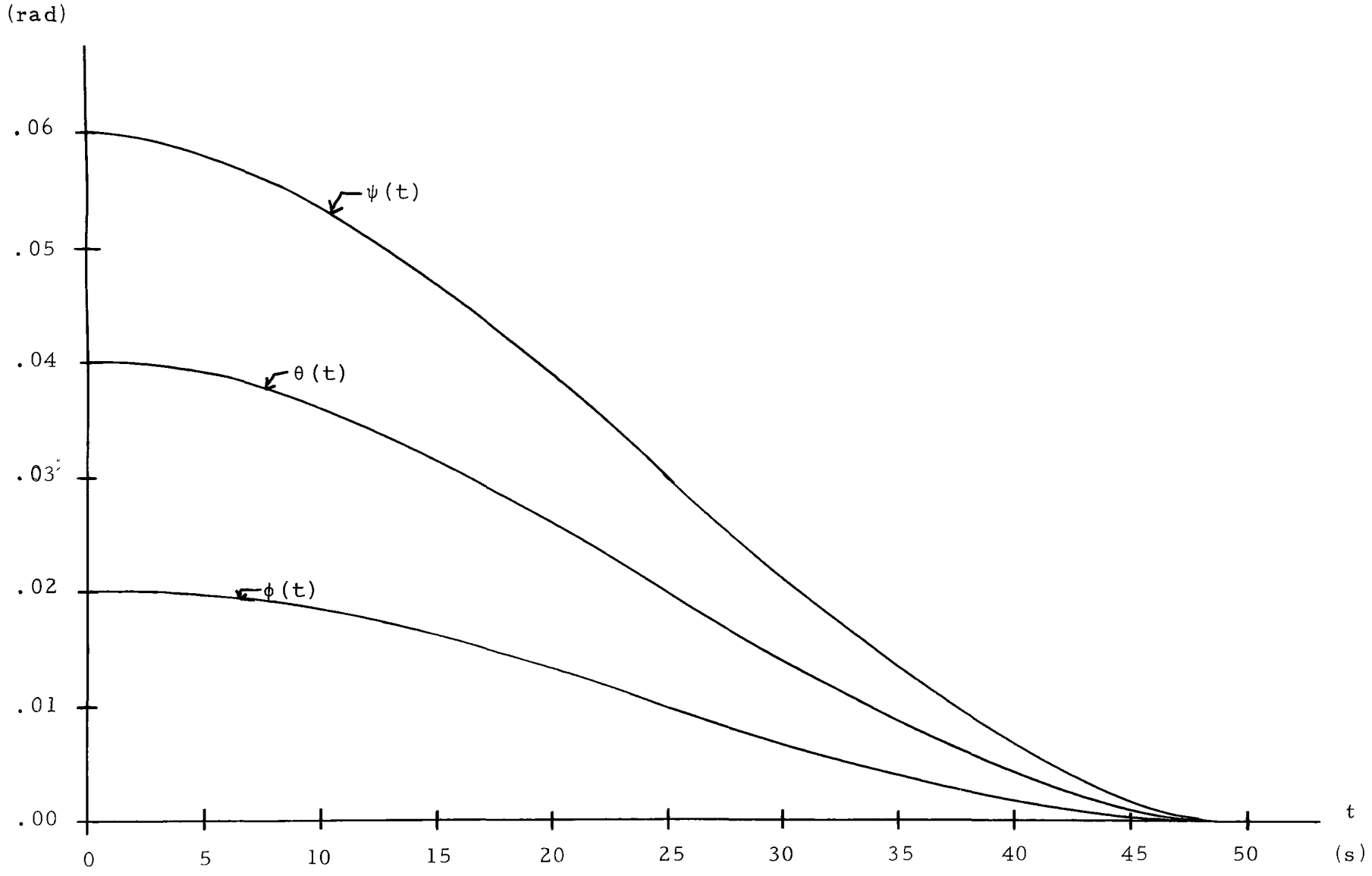


Fig. 4.2.4 Euler Angles for Problem 4.2.A

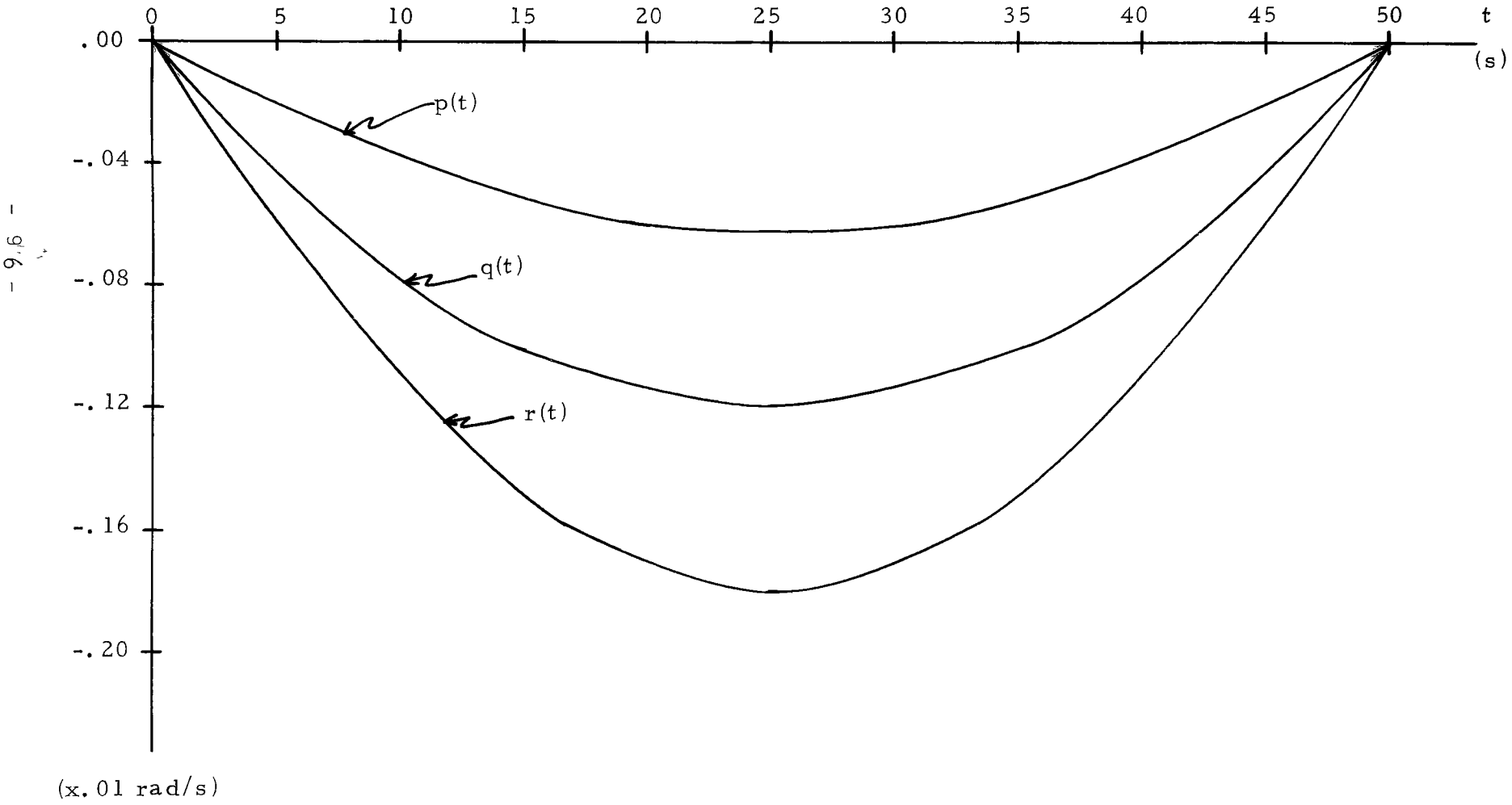


Fig. 4.2.5 Body Rates for Problem 4.2.A

Problem 4.2.B Flywheel Control (Non-Saturating and Inoperative): As mentiond earlier there appears to be a correspondence between the flywheel speeds and the initial conditions. In fact, the results of this problem will demonstrate that a particular flywheel is able to correct for a disturbance in a particular Euler angle. This also clarifies the question of controllability when one or more of the flywheels are defective. Three special cases are studied, each corresponding to two defective flywheels and one Euler angle initially disturbed. This will demonstrate the individuality of each flywheel's corrective capability.

The numerical values used are the same as for problem 4.2.A, with the following particular cases:

Case 1;

$$\left.\begin{aligned} \Omega_y(t) &= u_2(t) \equiv 0. \\ \Omega_z(t) &= u_3(t) \equiv 0. \end{aligned}\right\} \quad \text{for all } t\varepsilon\left[0, t_f\right] \qquad \left.\begin{array}{l} \\ \\ \\ \text{with initial conditions:} \\ \phi(0)=.02 \text{ rad}, \quad \theta(0)=.00 \text{ rad}, \quad \psi(0)=.00 \text{ rad} \end{array}\right\} \quad (4.2.10)$$

Case 2;

$$\left.\begin{aligned} \Omega_x(t) &= u_1(t) \equiv 0. \\ \Omega_z(t) &= u_3(t) \equiv 0. \end{aligned}\right\} \quad \text{for all } t\varepsilon\left[0, t_f\right] \qquad \left.\begin{array}{l} \\ \\ \\ \text{with initial conditions:} \\ \phi(0)=.00 \text{ rad}, \quad \theta(0)=.04 \text{ rad}, \quad \psi(0)=.00 \text{ rad} \end{array}\right\} \quad (4.2.11)$$

Case 3;

$$\left.\begin{aligned} \Omega_x(t) &= u_1(t) \equiv 0. \\ \Omega_y(t) &= u_2(t) \equiv 0. \end{aligned}\right\} \quad \text{for all } t\varepsilon\left[0, t_f\right] \qquad \left.\begin{array}{l} \\ \\ \\ \vdots \end{array}\right\} \quad (4.2.12)$$

with initial conditions:

$$\left.\begin{array}{l} \phi(0)=.00 \text{ rad}, \quad \theta(0)=.00 \text{ rad}, \quad \psi(0)=.06 \text{ rad} \end{array}\right\}$$

The graphical results shown in Figures 4.2.6-4.2.10 are for Case 3. The results for Cases 1 and 2 are similar. It can be observed that these results are essentially, disjoint parts of the complete solution given in problem 4.2.A. This special feature will be valuable in the design of feedback controls as discussed in the next chapter, because it demonstrates that each control can be treated individually. This amounts to a justification for decoupling the system, an approach commonly adopted when designing attitude control systems.

Thus it appears that all three flywheels must be operational to correct disturbances in any of the three Euler angles. To establish this more conclusively, another case was investigated. For this case it was assumed that all three Euler angles were initially perturbed. This will be referred to as Case 4 and in the context of the three previous cases, described by;

Case 4:

$$\left.\begin{array}{l} \Omega_x(t)=u_1(t)\equiv 0. \text{ for all } t\varepsilon\left[t_0,t_f\right] \\ \text{with initial conditions:} \\ \phi(0)=.02 \text{ rad}, \quad \theta(0)=.04 \text{ rad}, \quad \psi(0)=.06 \text{ rad} \end{array}\right\} \qquad (4.2.13)$$

The graphical results shown in Figures 4.2.11-4.2.15 are

for Case 4. These clearly demonstrate that all three flywheels must be operational to correct disturbances in any of the Euler angles. Again, it must be emphasized that no body rate disturbances can be corrected since there is no net change in the overall system's angular momentum.

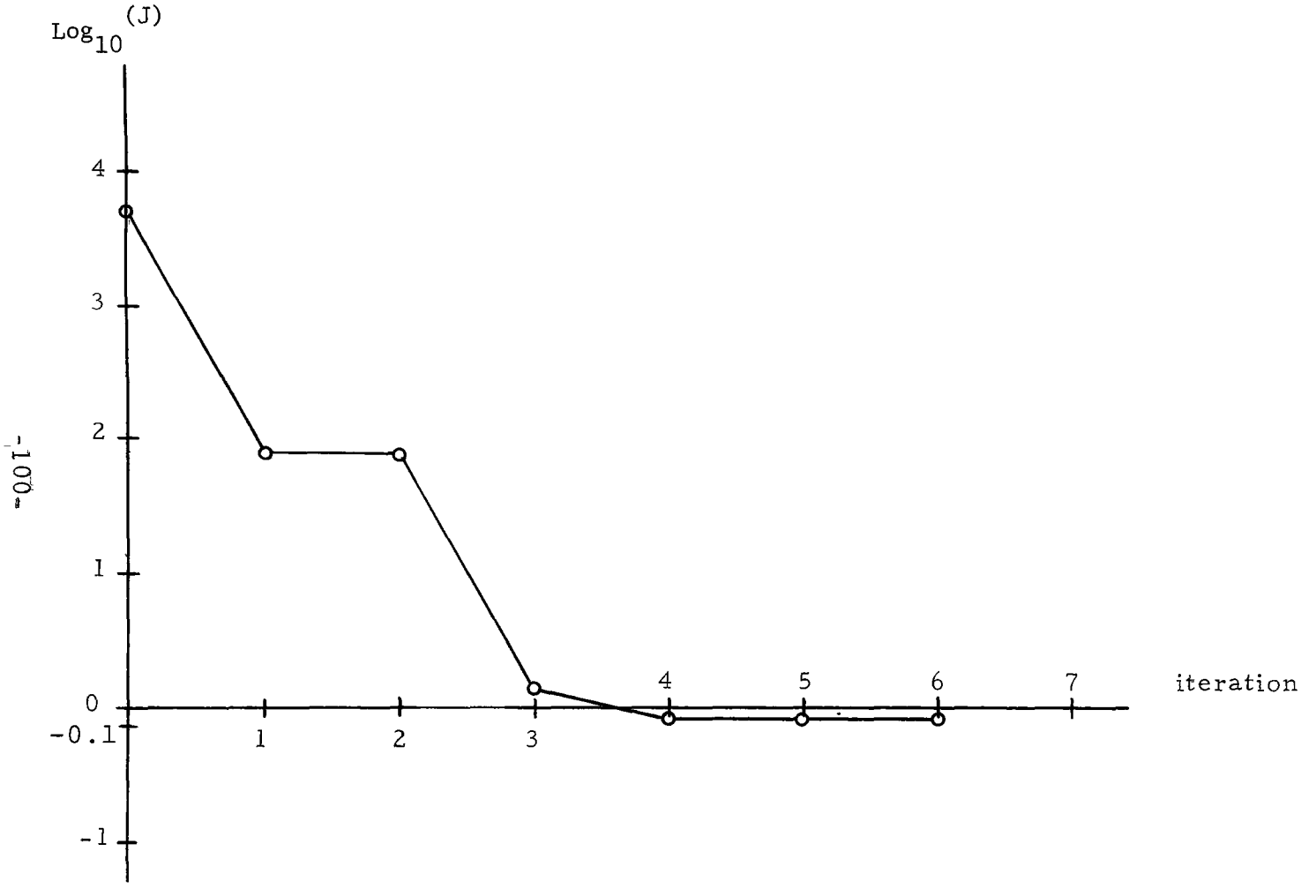


Fig. 4.2.6 $\text{Log}_{10}{}^{(J)}$ for Problem 4.2.B Case 3

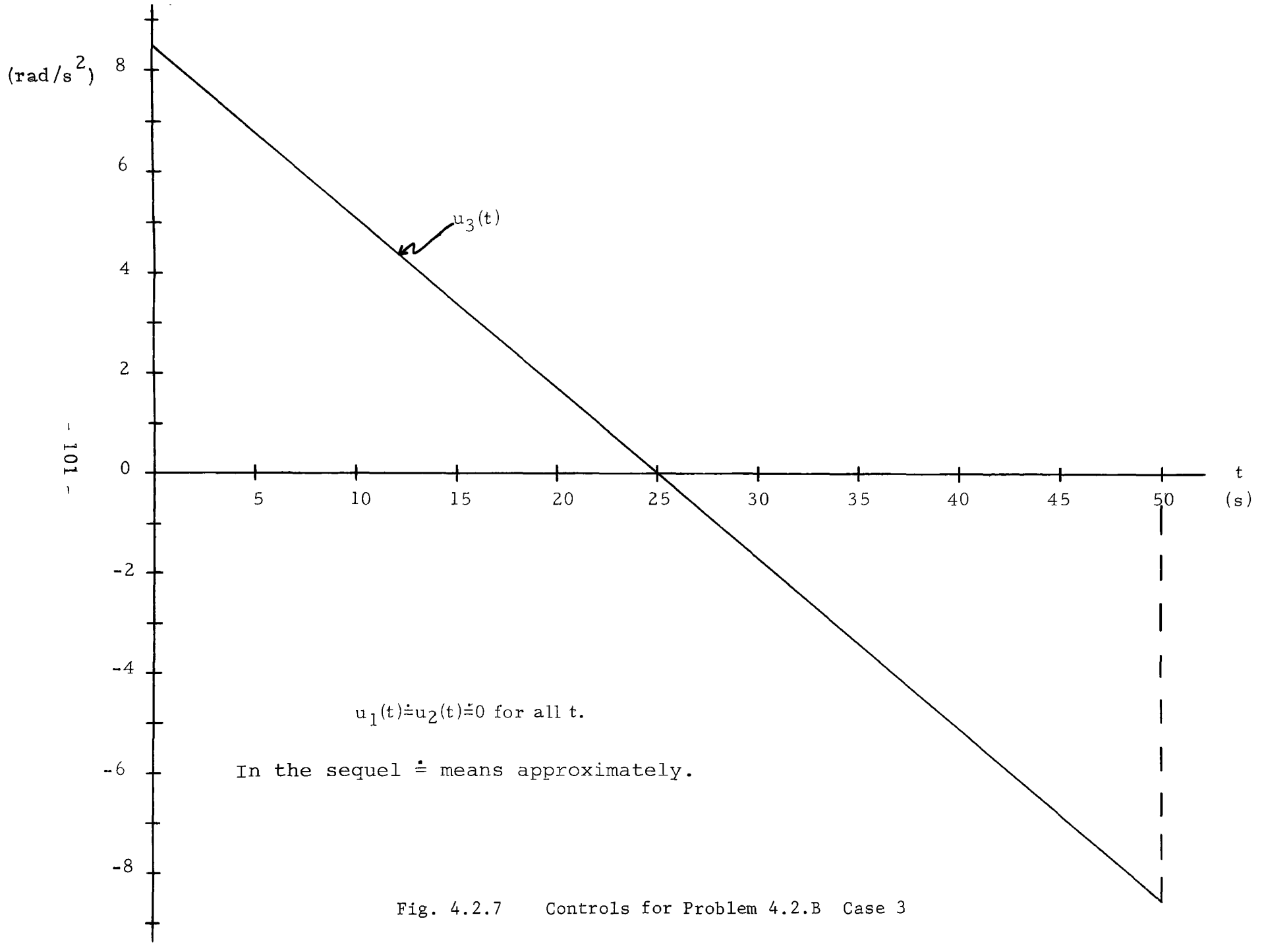


Fig. 4.2.7 Controls for Problem 4.2.B Case 3

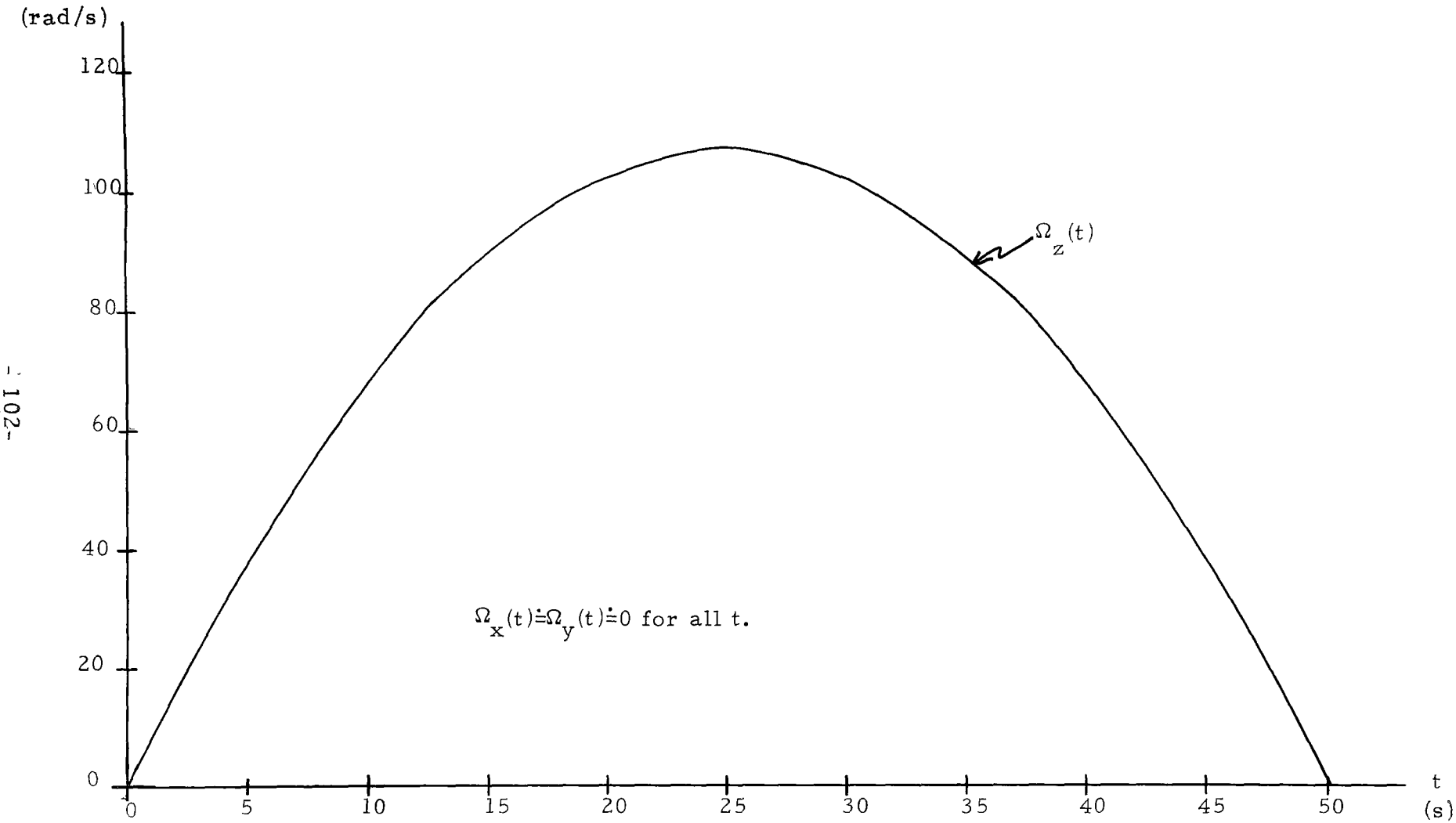


Fig. 4.2.8 Flywheel Angular Velocities for Problem 4.2.B Case 3

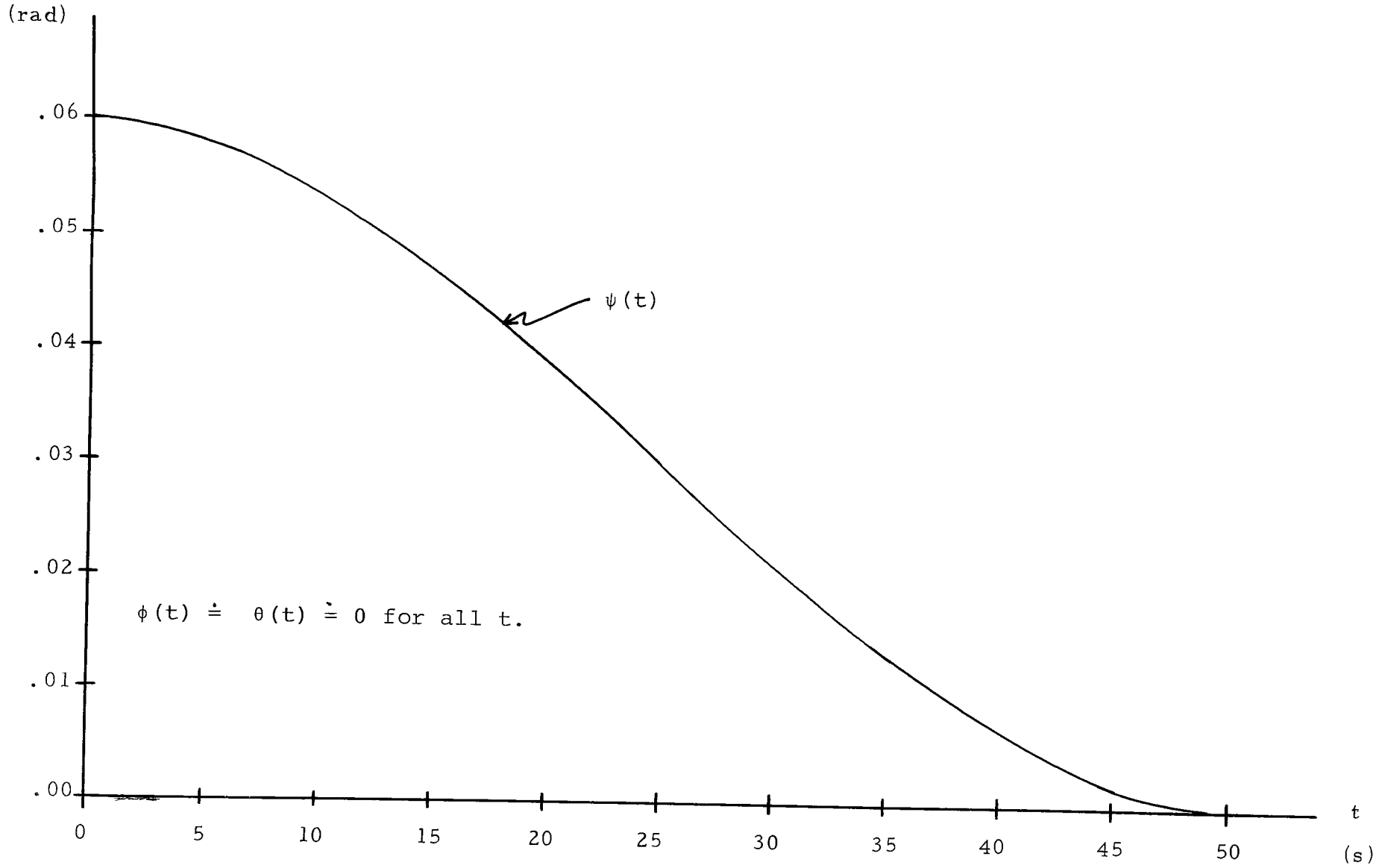


Fig. 4.2.9 Euler Angles for Problem 4.2.B Case 3

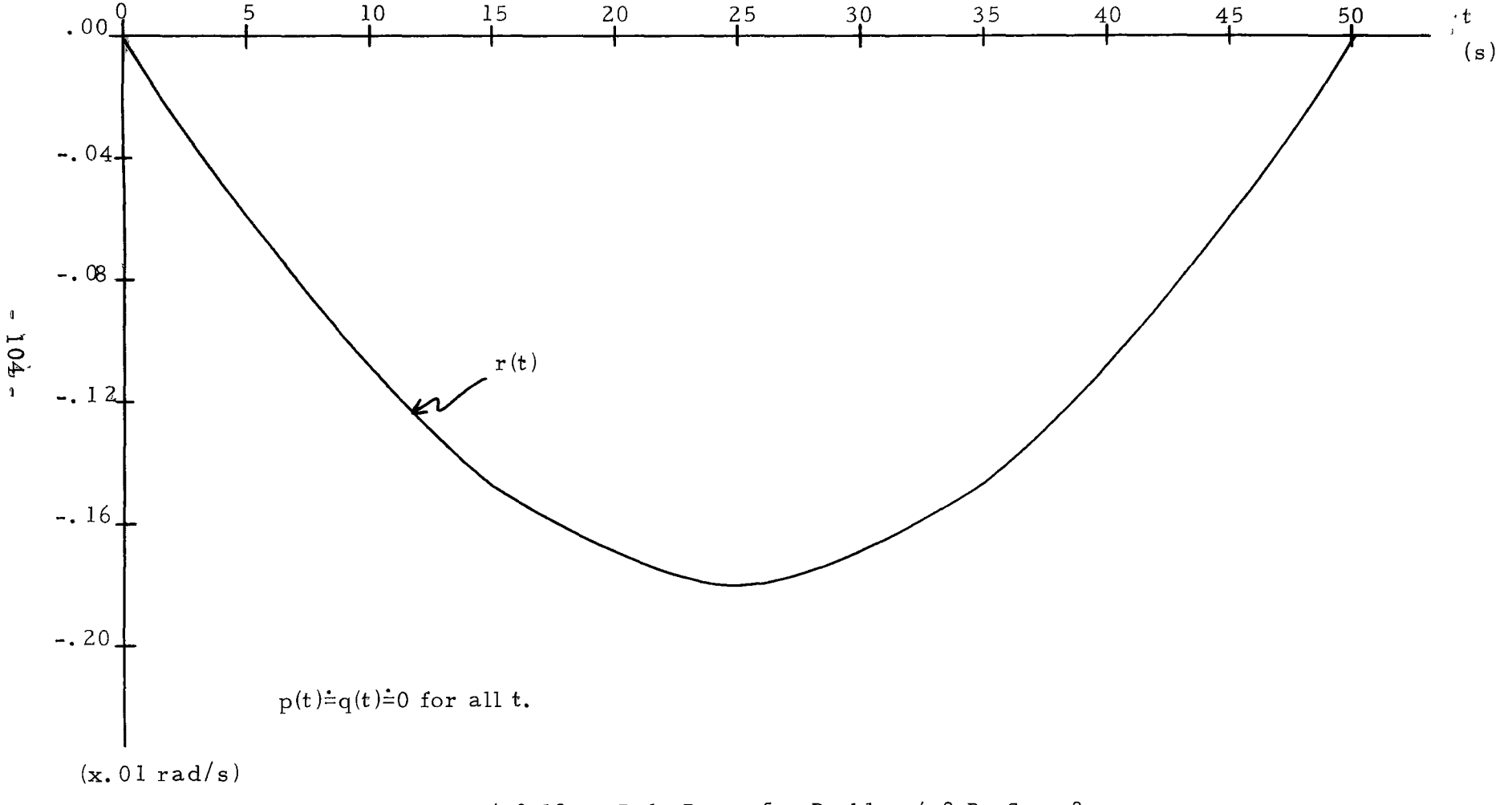


Fig. 4.2.10 Body Rates for Problem 4.2.B Case 3

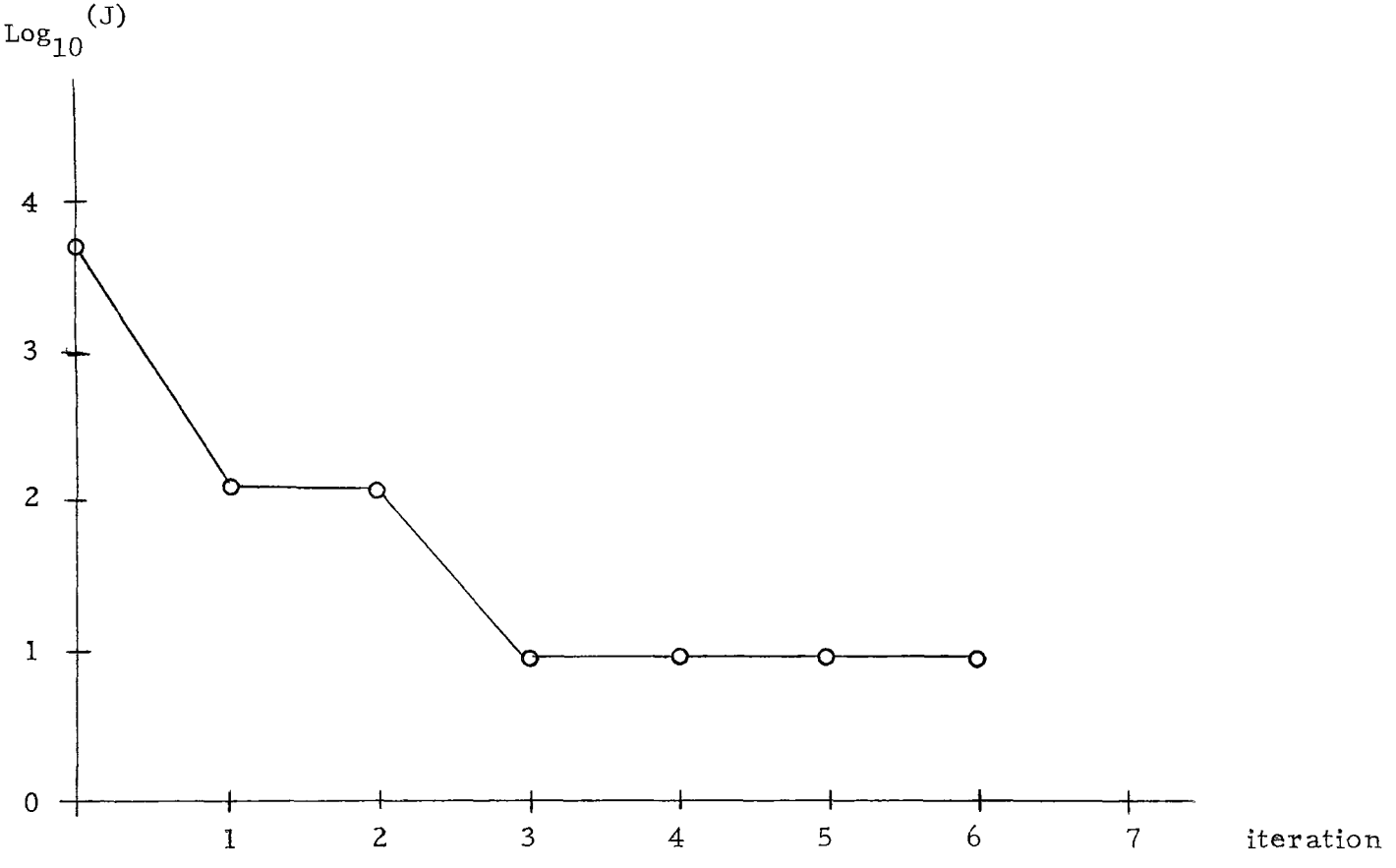


Fig. 4.2.11 $\text{Log}_{10}{}^{(J)}$ for Problem 4.2.B Case 4

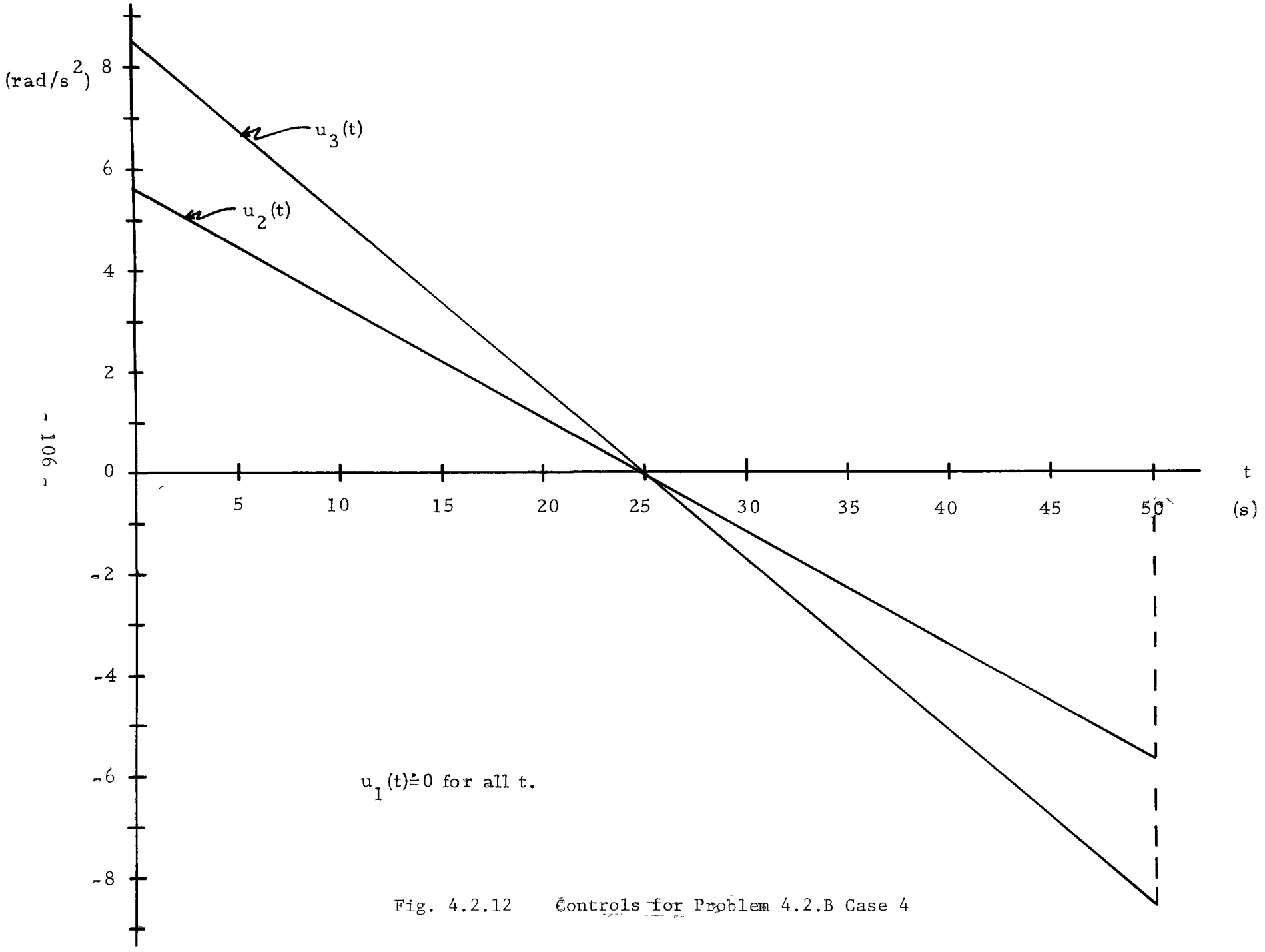


Fig. 4.2.12 Controls for Problem 4.2.B Case 4

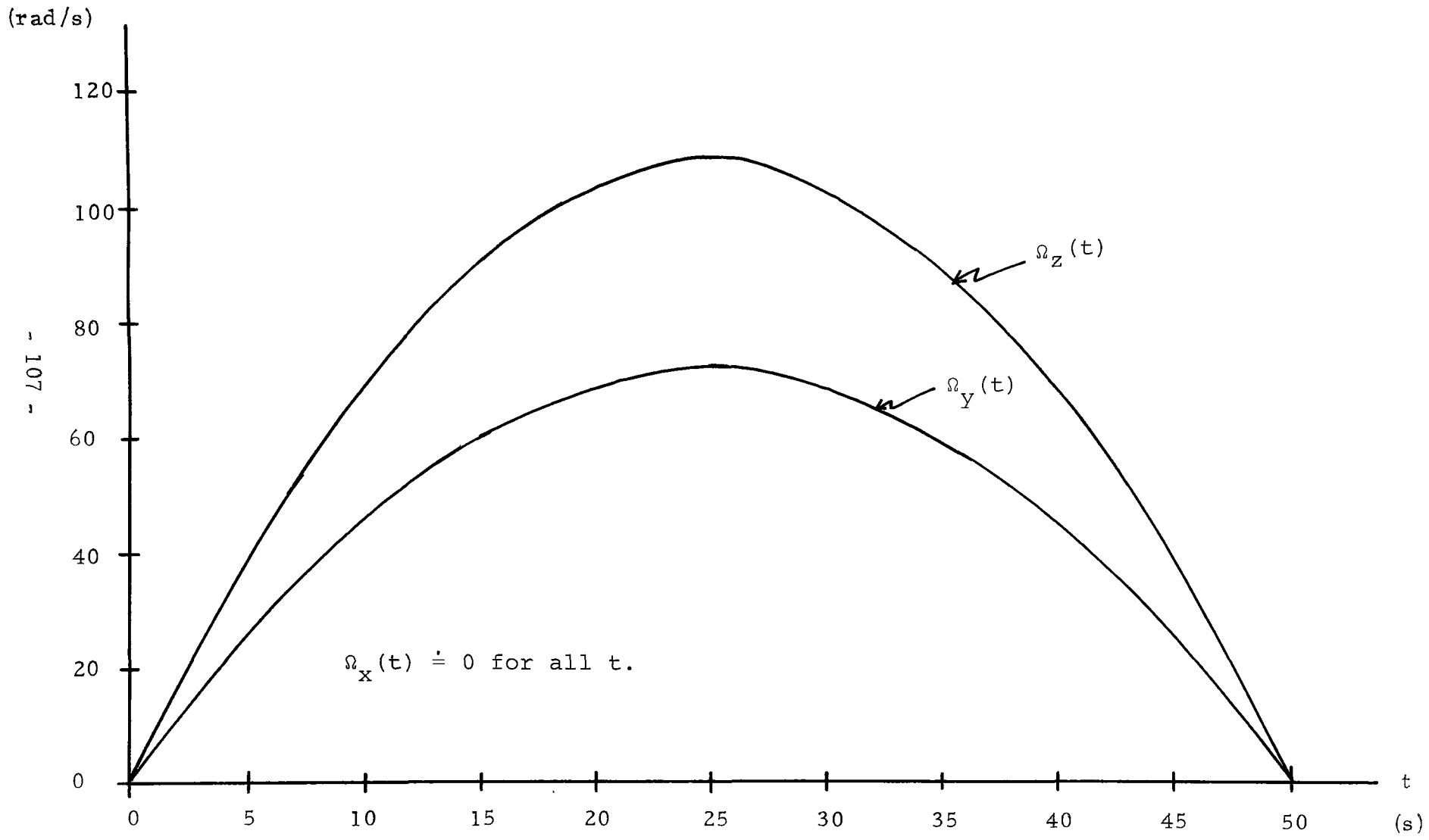


Fig. 4.2.13 Flywheel Angular Velocities for Problem 4.2.B Case 4

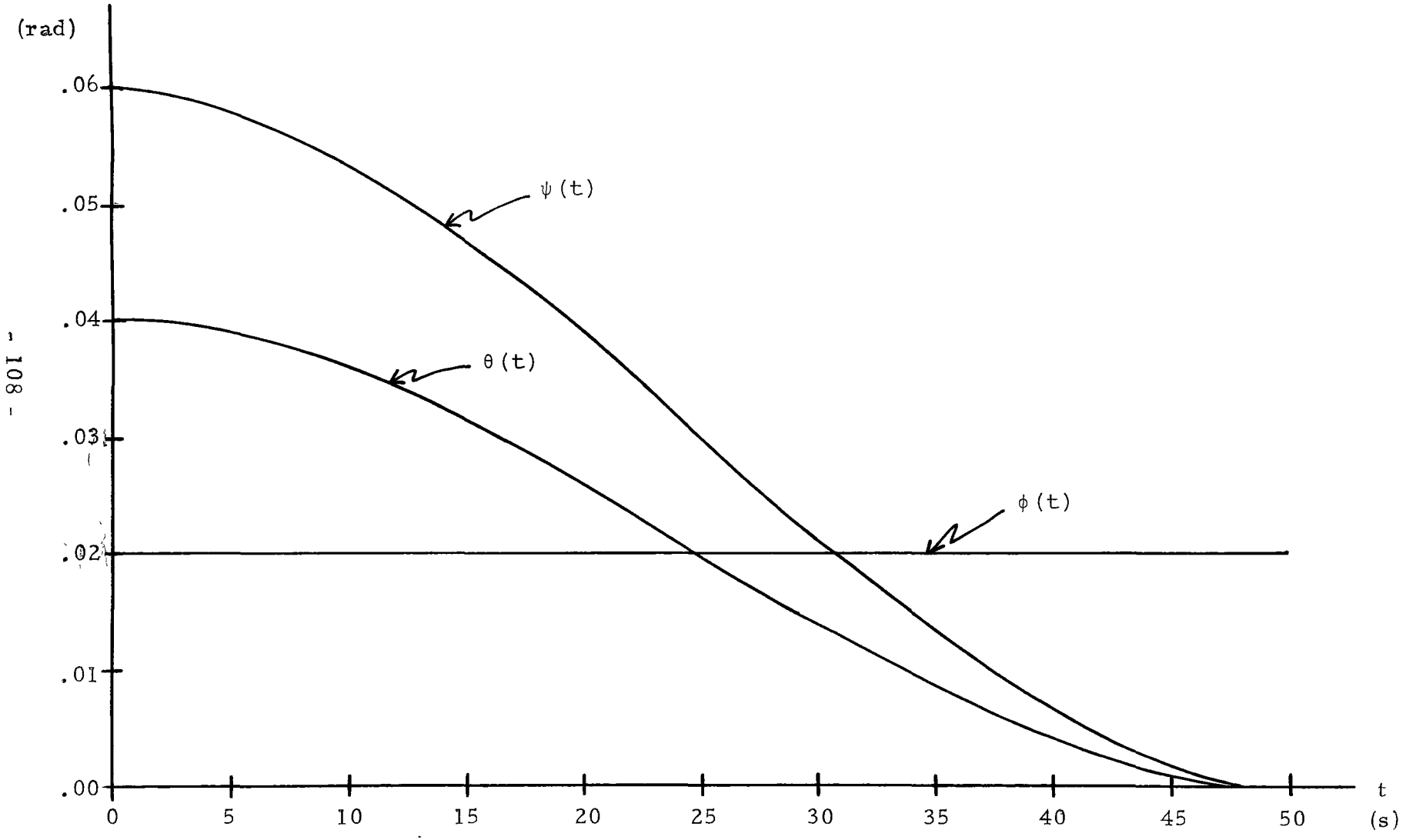


Fig. 4.2.14 Euler Angles for Problem 4.2.B Case 4

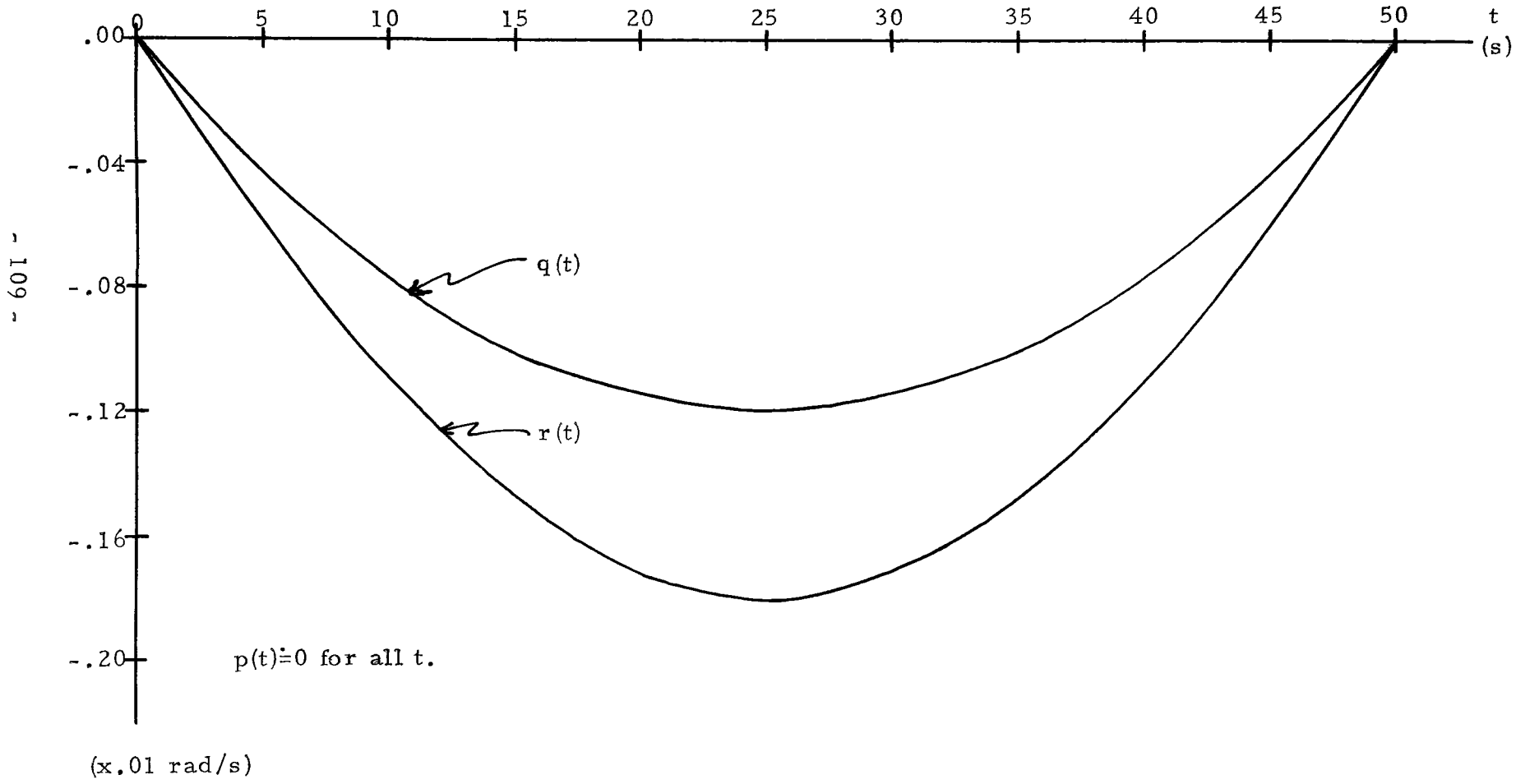


Fig. 4.2.15 Body Rates for Problem 4.2.B Case 4

Problem 4.2.C Flywheel Control (Non- Saturating with Speed Constraints): In this problem all the flywheels are assumed operational. However, their speed is constrained to be less than 94.25 rad./s. Since the flywheel speeds are considered to be state variables, it follows that speed constraints are in fact state constraints and can be treated according to the development given in section 3.6.

The problem to be solved can therefore be stated as follows;

$$\text{minimize } J = \left[I + \tfrac{1}{2}R_1(p^2+q^2+r^2+\phi^2+\theta^2+\psi^2) + \tfrac{1}{2}R_2(\Omega_x^2+\Omega_y^2+\Omega_z^2) \right]\Big|_{t=t_f}$$

$$u \varepsilon U^3$$

subject to;

$$|\Omega_\alpha| \leq \Omega_{\alpha max}, \quad \alpha = x,y,z \tag{4.2.14}$$

and to (4.2.1) for $t\varepsilon[0,t_f]$. Following the procedure outlined in section 3.6, a new state variable I_2 is introduced.

Let;

$$\dot{I}_2 = h_x^2(\Omega_x)\cdot H(h_x) + h_y^2(\Omega_y)\cdot H(h_y) + h_z^2(\Omega_z)\cdot H(h_z) \tag{4.2.15}$$

with $I_2(0)=0$, where;

$$h_\alpha(\Omega_\alpha) = \Omega_{\alpha max} - |\Omega_\alpha|, \quad \alpha = x,y,z \tag{4.2.16}$$

and:

$$H(h_\alpha(\Omega_\alpha)) = \begin{cases} 0 & \text{if } h_\alpha(\Omega_\alpha) \geq 0 \\ 1 & \text{if } h_\alpha(\Omega_\alpha) < 0 \end{cases} \qquad \alpha = x,y,z \tag{4.2.17}$$

Now since the value of I_2 at the terminal time is a direct indication of the violation of the constraints in (4.2.14), the requirement that $I_2(0)=0$ implies that these are satisfied for $t\varepsilon[0,t_f]$. Hence the problem is reformulated as;

$$\underset{u\in U^3}{\text{minimize}}\ J = \left[I+I_2+\tfrac{1}{2}R_1(p^2+q^2+r^2+\phi^2+\theta^2+\psi^2) + \tfrac{1}{2}R_2(\Omega_x^2+\Omega_y^2+\Omega_z^2)\right]\Big|_{t=t_f}$$

subject to (4.2.18) for $t\in[0\ ,t_f]$:

$$\begin{aligned}
\dot{p} &= A_1 r + A_2 qr + A_3 u_1 + A_4\Omega_y + A_5\Omega_z q\\
\dot{q} &= B_1 pr + B_2\Omega_x r + B_3 u_2 + B_4\Omega_z p\\
\dot{r} &= C_1 p + C_2 qp + C_3\Omega_x q + C_4\Omega_y p + C_5 u_3\\
\dot{\phi} &= p\\
\dot{\theta} &= q\\
\dot{\psi} &= r\\
\dot{\Omega}_x &= u_1\\
\dot{\Omega}_y &= u_2\\
\dot{\Omega}_z &= u_3\\
\dot{I} &= \tfrac{1}{2}\lambda_1(p^2+q^2+r^2+\phi^2+\theta^2+\psi^2) + \tfrac{1}{2}\lambda_2(u_1^2+u_2^2+u_3^2)\\
\dot{I}_2 &= h_x^2(\Omega_x)\cdot H(h_x) + h_y^2(\Omega_y)\cdot H(h_y) + h_z^2(\Omega_z)\cdot H(h_z)
\end{aligned} \tag{4.2.18}$$

It may be observed that the right hand side of $\dot{I}_2$ is indeed continuous and continuously differentiable with respect to the state variables Ω_x, Ω_y and Ω_z respectively. In fact:

$$\frac{d\left[h^2(\Omega_\alpha)\cdot H(h_\alpha)\right]}{d\Omega_\alpha} = \begin{cases} 2(\Omega_\alpha(t)-\Omega_{\alpha max}); \text{if } \Omega_\alpha(t) > \Omega_{\alpha max}\\ 0\ ;\ \text{if } |\Omega_\alpha(t)| \leq \Omega_{\alpha max}\\ 2(\Omega_\alpha(t)+\Omega_{\alpha max}); \text{if } \Omega_\alpha(t) \leq -\Omega_{\alpha max}\end{cases} \tag{4.2.19}$$

$\alpha = x,y,z$

From PMP it can easily be checked that the optimal controls are continuous functions of time. Therefore the control gradients are defined using (3.4.53).

The results shown in Figures 4.2.16-4.2.20 correspond to the following initial conditions and parameter values:

$p(0)=.00$ rad/s $\quad \phi(0)=.02$ rad $\quad \Omega_x(0)=0.$rad/s

$q(0)=.00$ rad/s $\quad \theta(0)=.04$ rad $\quad \Omega_y(0)=0.$rad/s

$r(0)=.00$ rad/s $\quad \psi(0)=.06$ rad $\quad \Omega_z(0)=0.$rad/s (4.2.20)

$I(0)=0.$ $\quad I_2(0)=0.$ $\quad R_1=4\times10^4$ $\quad R_2=100$

$\lambda_1=5$ $\quad \lambda_2=.001$ $\quad t_f=50$ s

From the graphical results, it can be observed that the maximum flywheel velocities do not grossly exceed the upper limit set at approximately 94 rad/s. Therefore, this method does provide an effective means of incorporating state constraints.

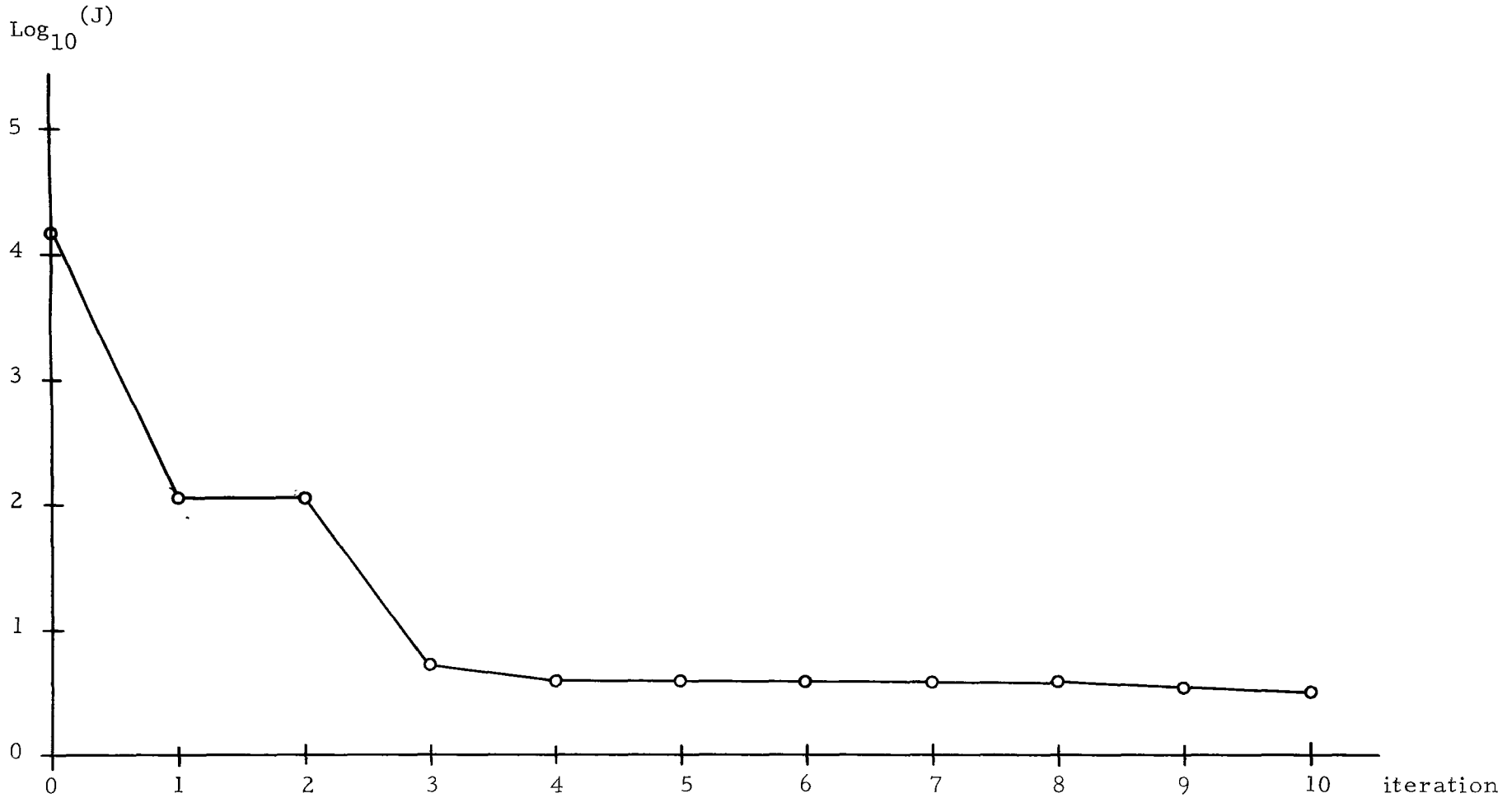


Fig. 4.2.16 $\text{Log}_{10}{}^{(J)}$ for Problem 4.2.C

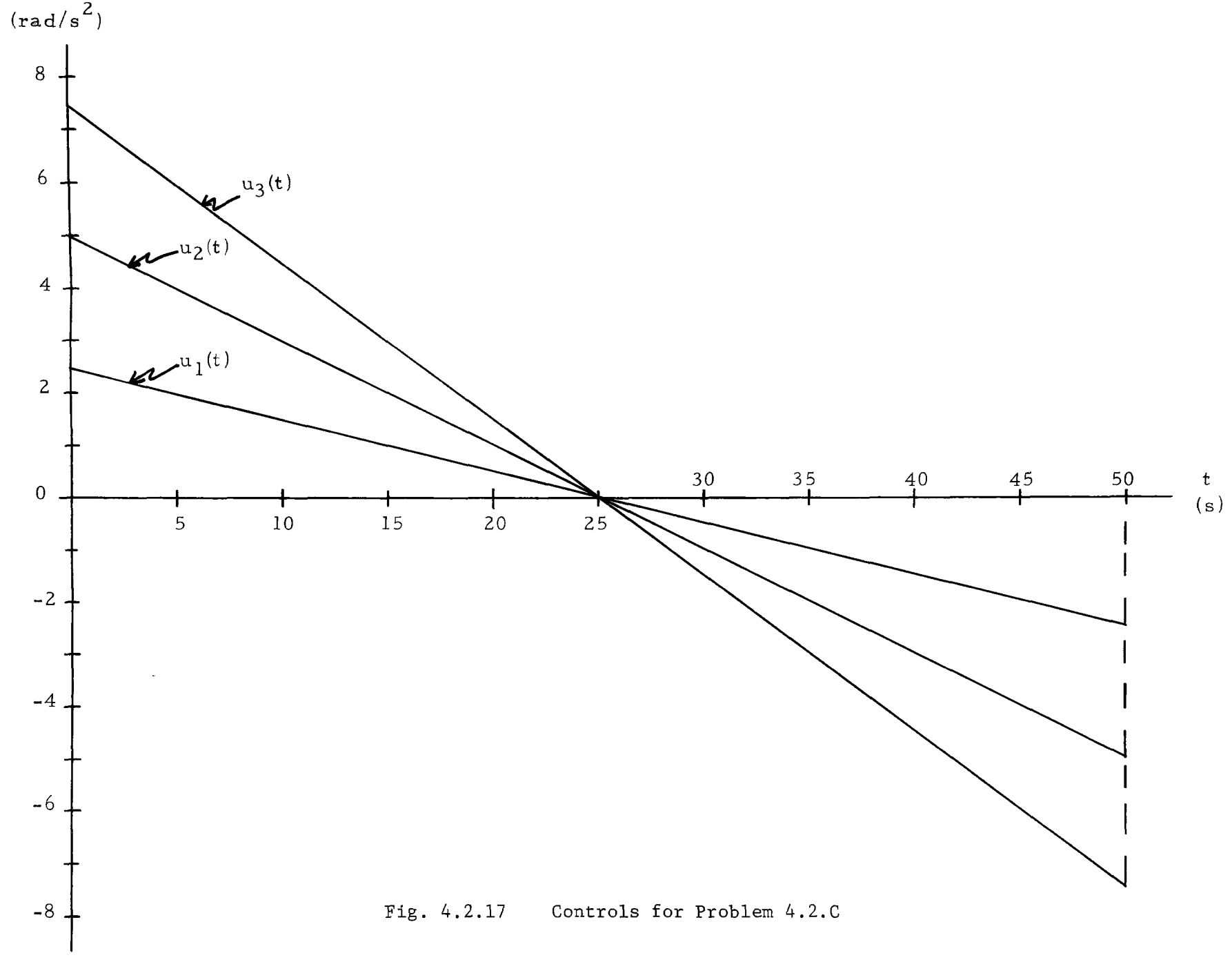


Fig. 4.2.17 Controls for Problem 4.2.C

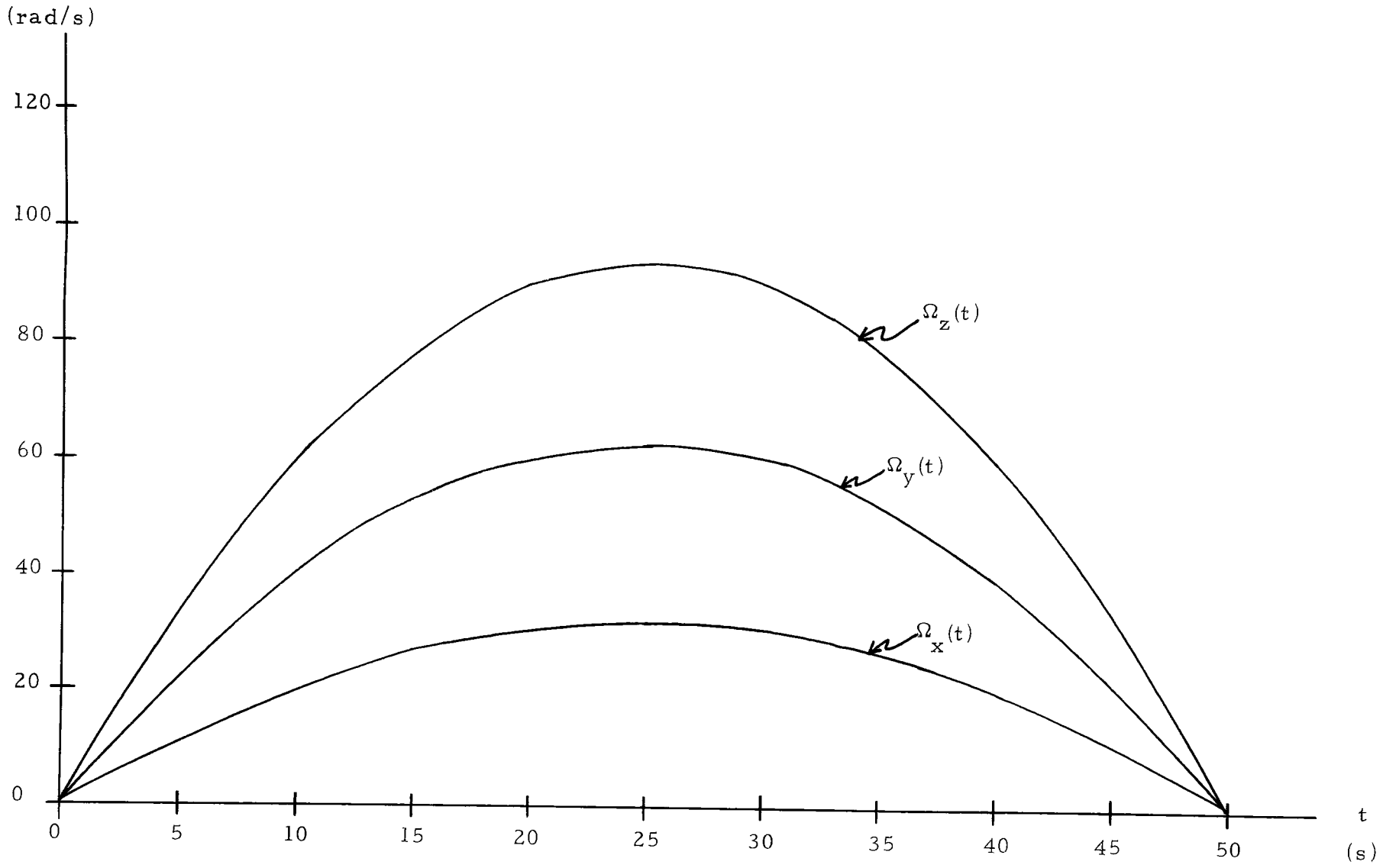


Fig. 4.2.17 Flywheel Angular Velocities for Problem 4.2.C

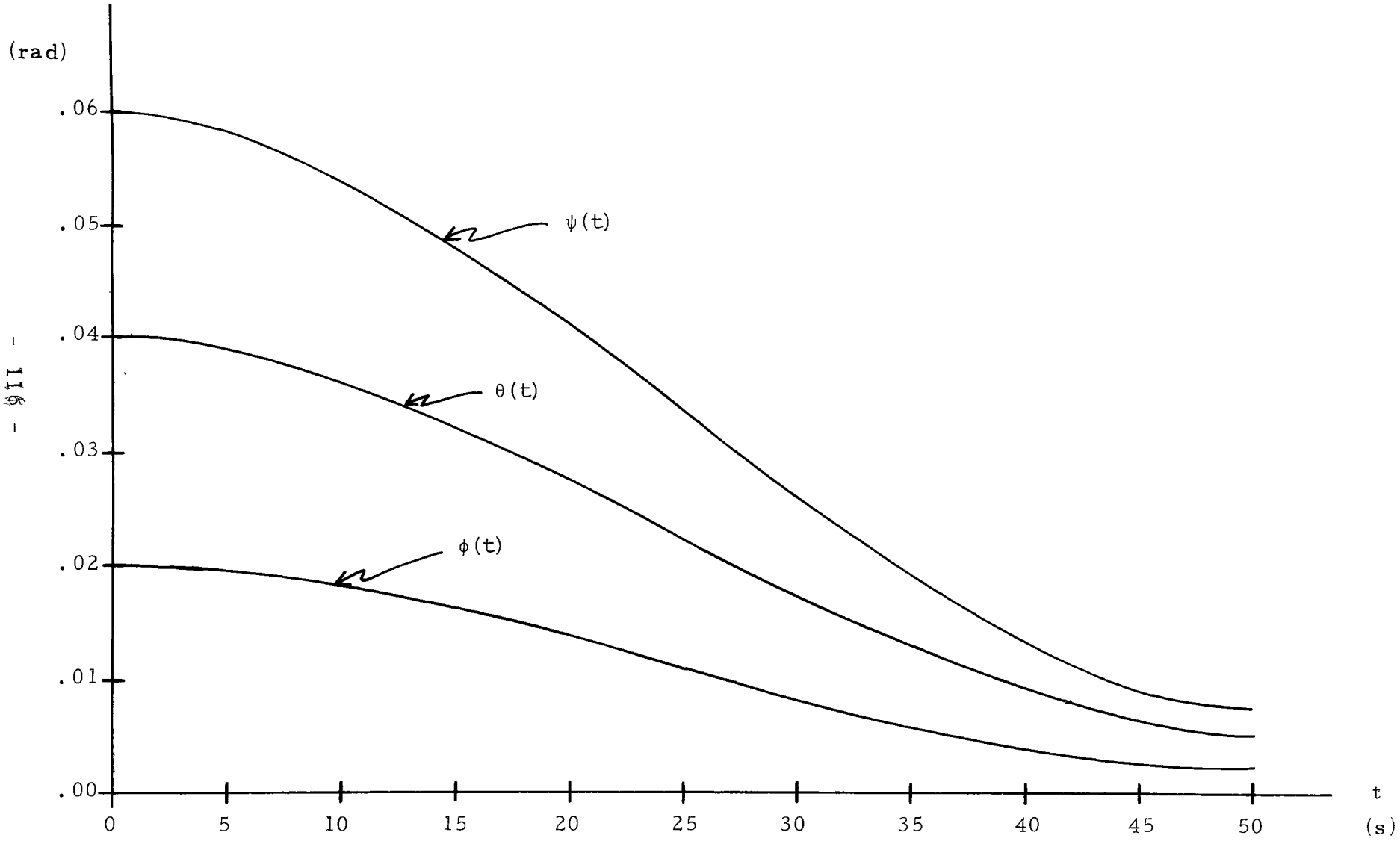


Fig. 4.2.19 Euler Angles for Problem 4.2.C

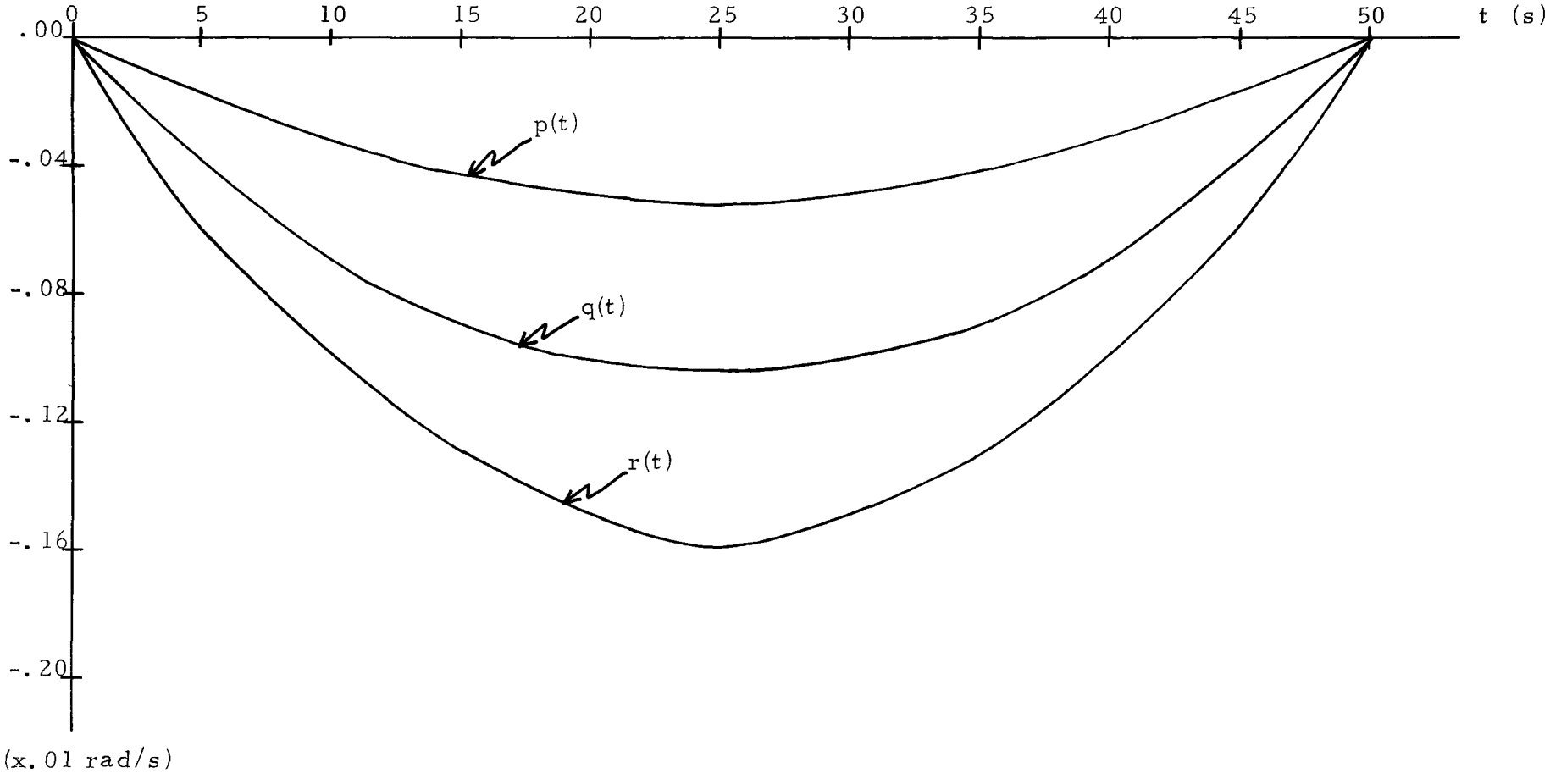


Fig. 4.2.20 Body Rates for Problem 4.2.C

Problem 4.2.D Flywheel Control (Saturating): The use of non-saturating flywheels has the major disadvantage of being incapable of correcting initial disturbances in the satellite's body rates. To overcome this, saturating flywheels or reaction jets must be used.

The problem statement is very similar to that of 4.2.A and 4.2.B and is as follows;

$$\underset{u\in U^3}{\text{minimize}}\ J = \left[I+\tfrac{1}{2}R_1(p^2+q^2+r^2+\phi^2+\theta^2+\psi^2)\right]\Big|_{t=t_f}$$

subject to (4.2.21) for $t\in[0,t_f]$, where:

$$\begin{aligned}
\dot{p} &= A_1 r + A_2 qr + A_3 u_1 + A_4 \Omega_y r + A_5 \Omega_z q \\
\dot{q} &= B_1 pr + B_2 \Omega_x r + B_3 u_2 + B_4 \Omega_z p \\
\dot{r} &= C_1 p + C_2 qp + C_3 \Omega_x q + C_4 \Omega_y p + C_5 u_3 \\
\dot{\phi} &= p \\
\dot{\theta} &= q \\
\dot{\psi} &= r \\
\dot{\Omega}_x &= u_1 \\
\dot{\Omega}_y &= u_2 \\
\dot{\Omega}_z &= u_3 \\
\dot{I} &= \tfrac{1}{2}\lambda_1(p^2+q^2+r^2+\phi^2+\theta^2+\psi^2) + \tfrac{1}{2}\lambda_2(u_1^2+u_2^2+u_3^2) + \tfrac{1}{2}\lambda_3(\Omega_x^2+\Omega_y^2+\Omega_z^2)
\end{aligned} \qquad (4.2.21)$$

U, the set of admissible controls, consists of all continuous functions of time on $[0,t_f]$. It can be observed that in this problem the requirement of having the angular velocities Ω_x, Ω_y, Ω_z return to zero at the final time has been removed. The value of λ_3 can be chosen quite small because

it simply serves as a preventive mechanism against having a runaway situation in the angular velocities. The co-state differential equations and control gradients are obtained in a manner similar to that of problem 4.2.A. The optimal controls and associated state trajectories were obtained for the following initial conditions and constants:

$$
\begin{array}{lll}
p(0)=.001 \text{ rad/s} & \phi(0)=.02 \text{ rad} & \Omega_x(0)=0. \text{rad/s} \\
q(0)=.002 \text{ rad/s} & \theta(0)=.04 \text{ rad} & \Omega_y(0)=0. \text{rad/s} \\
r(0)=.003 \text{ rad/s} & \psi(0)=.06 \text{ rad} & \Omega_z(0)=0. \text{rad/s} \\
I(0)=0 & \lambda_1= 5 & \lambda_2= .001 \\
\lambda_3=1\text{x}10^{-8} & R_1= 4\text{x}10^{6} & t_f= 50 \text{ s}
\end{array}
\qquad (4.2.21)
$$

The graphical results displayed in Figures 4.2.21-4.2.25 demonstrate the ability to move the system to the origin with non-zero initial velocities. It may be observed that even with fairly small initial disturbances in the body rates, the maximum flywheel speed achieved is guite large. Thus,for any substantial disturbance in the body rates it will be necessary to use reaction jets before applying flywheel control.

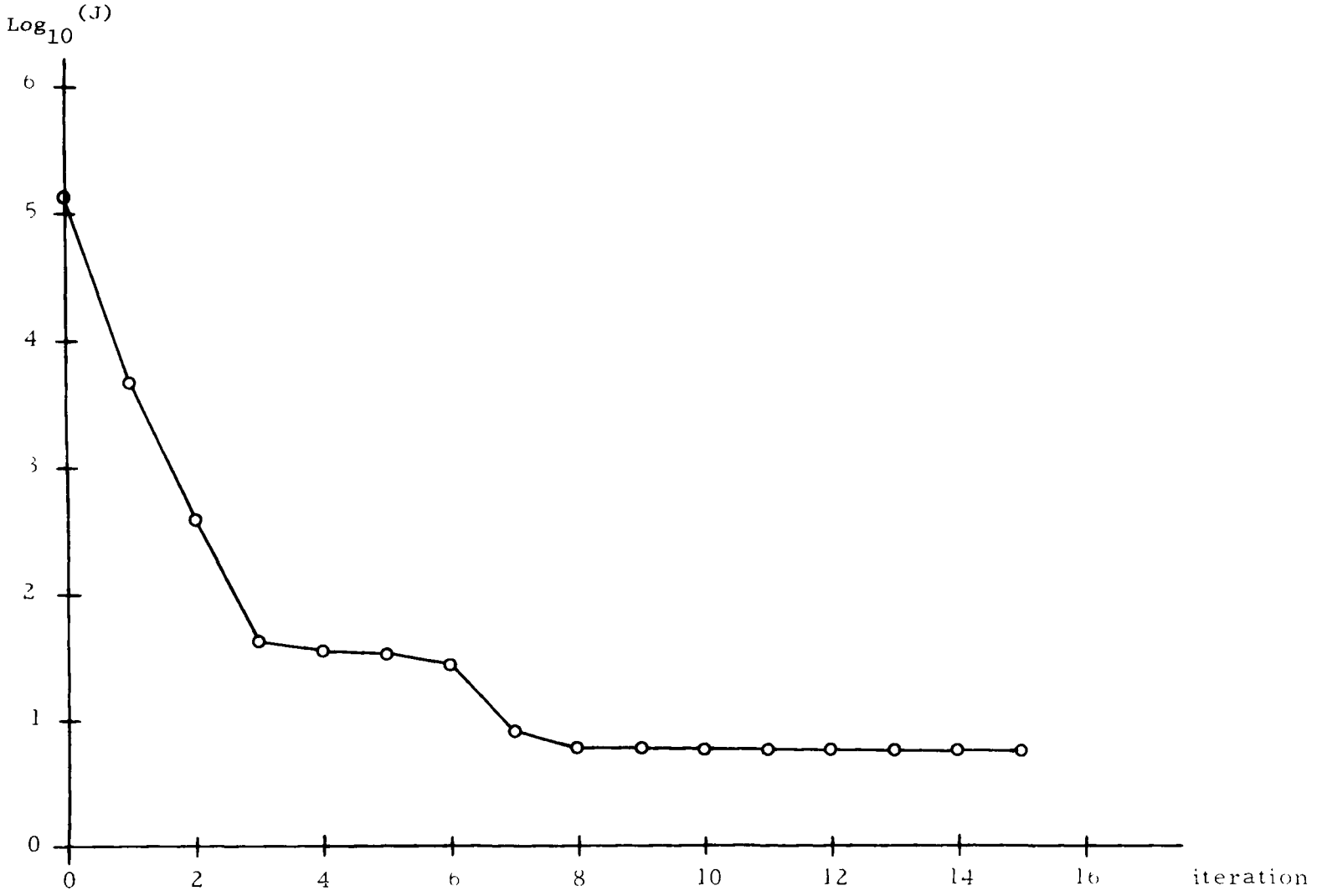


Fig. 4.2.21 $\text{Log}_{10}^{(J)}$ for Problem 4.2.D

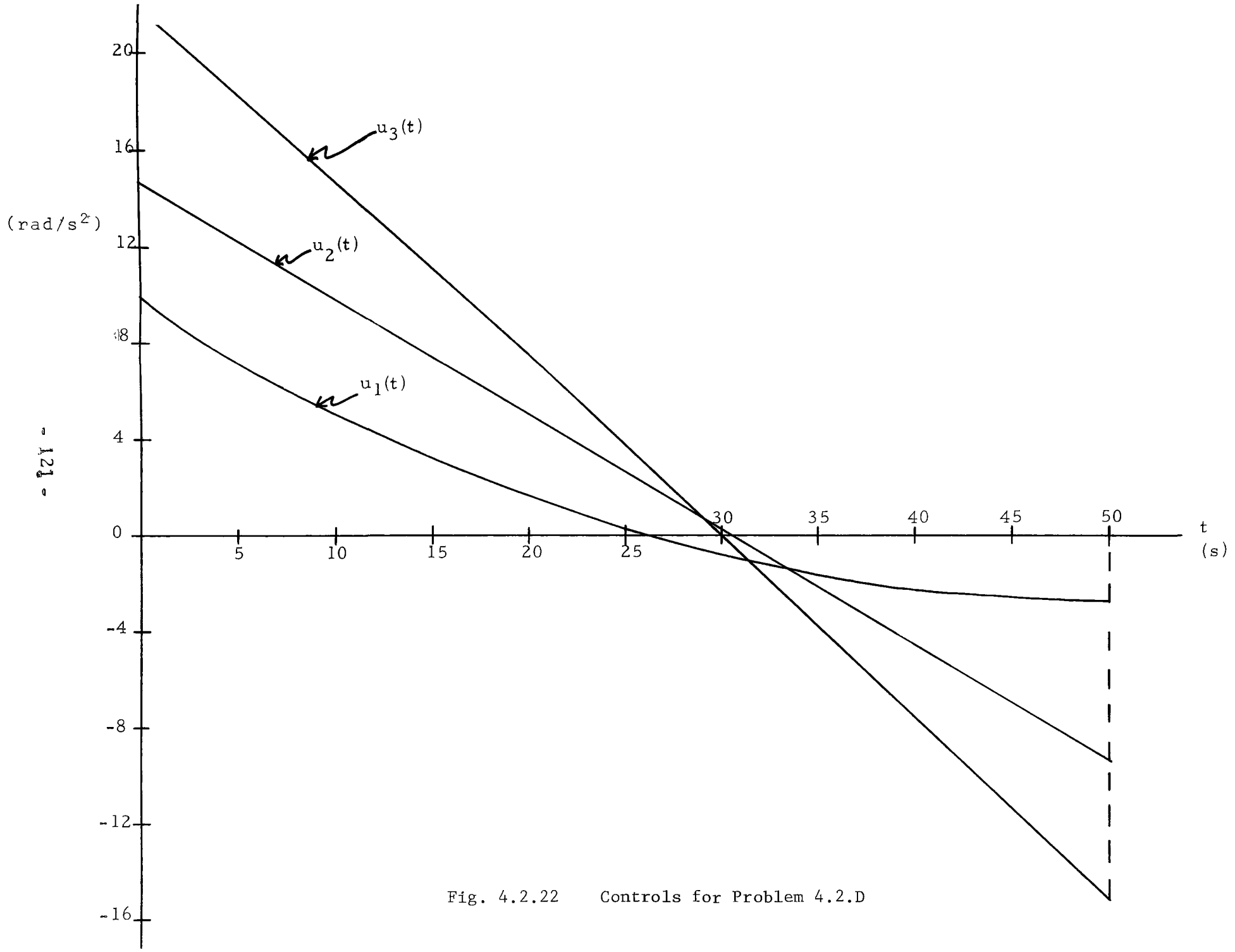


Fig. 4.2.22 Controls for Problem 4.2.D

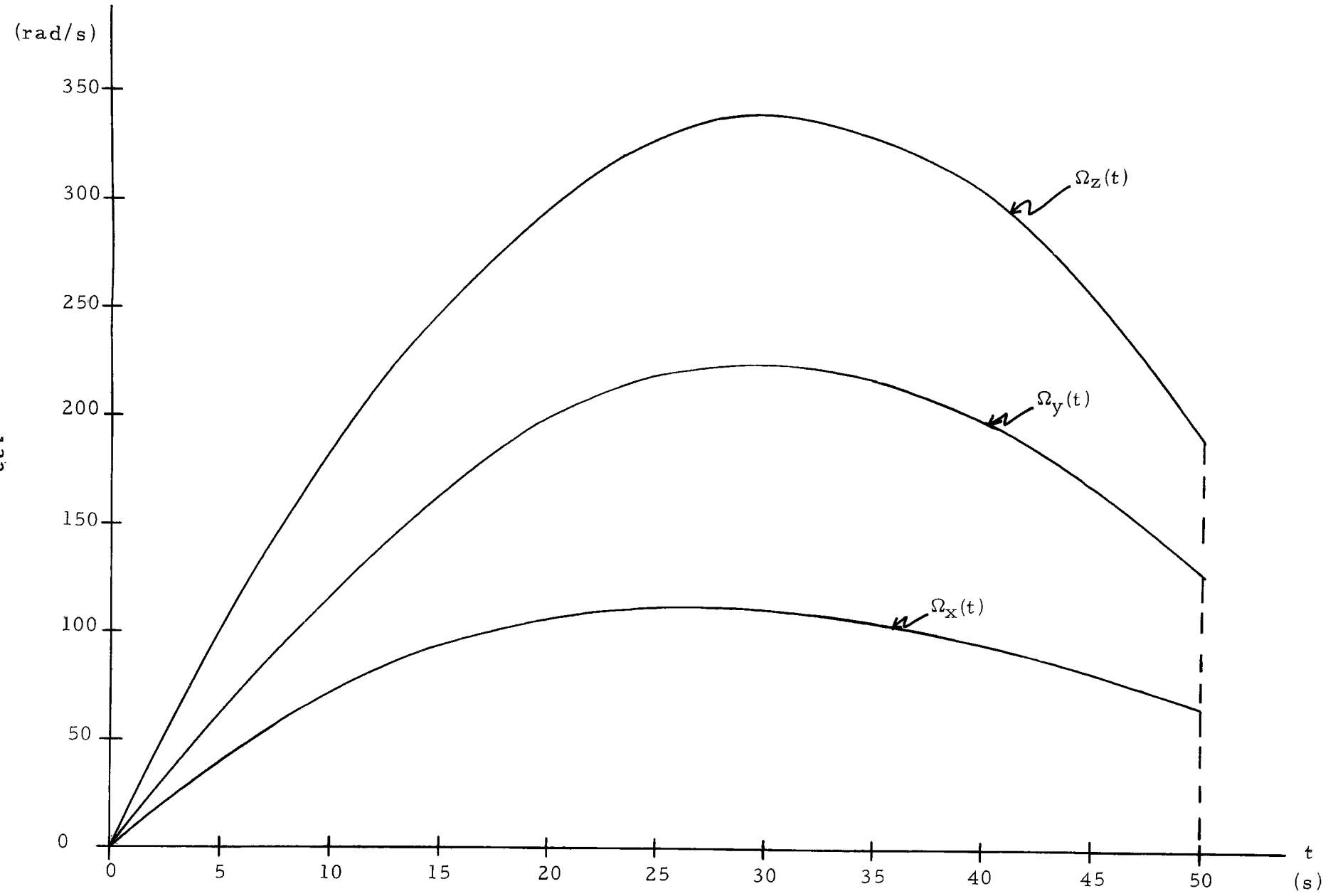


Fig. 4.2.23 Flywheel Angular Velocities for Problem 4.2.D

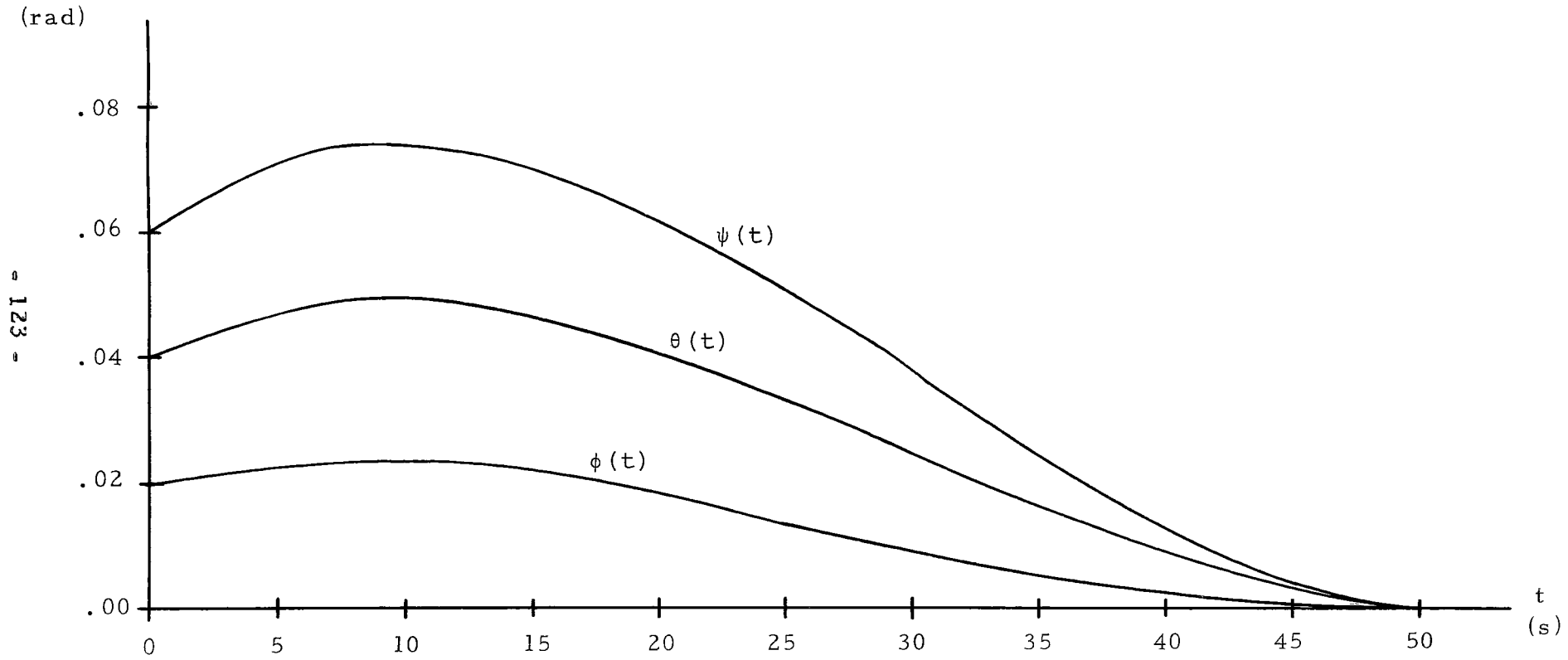


Fig. 4.2.24 Euler Angles for Problem 4.2.D

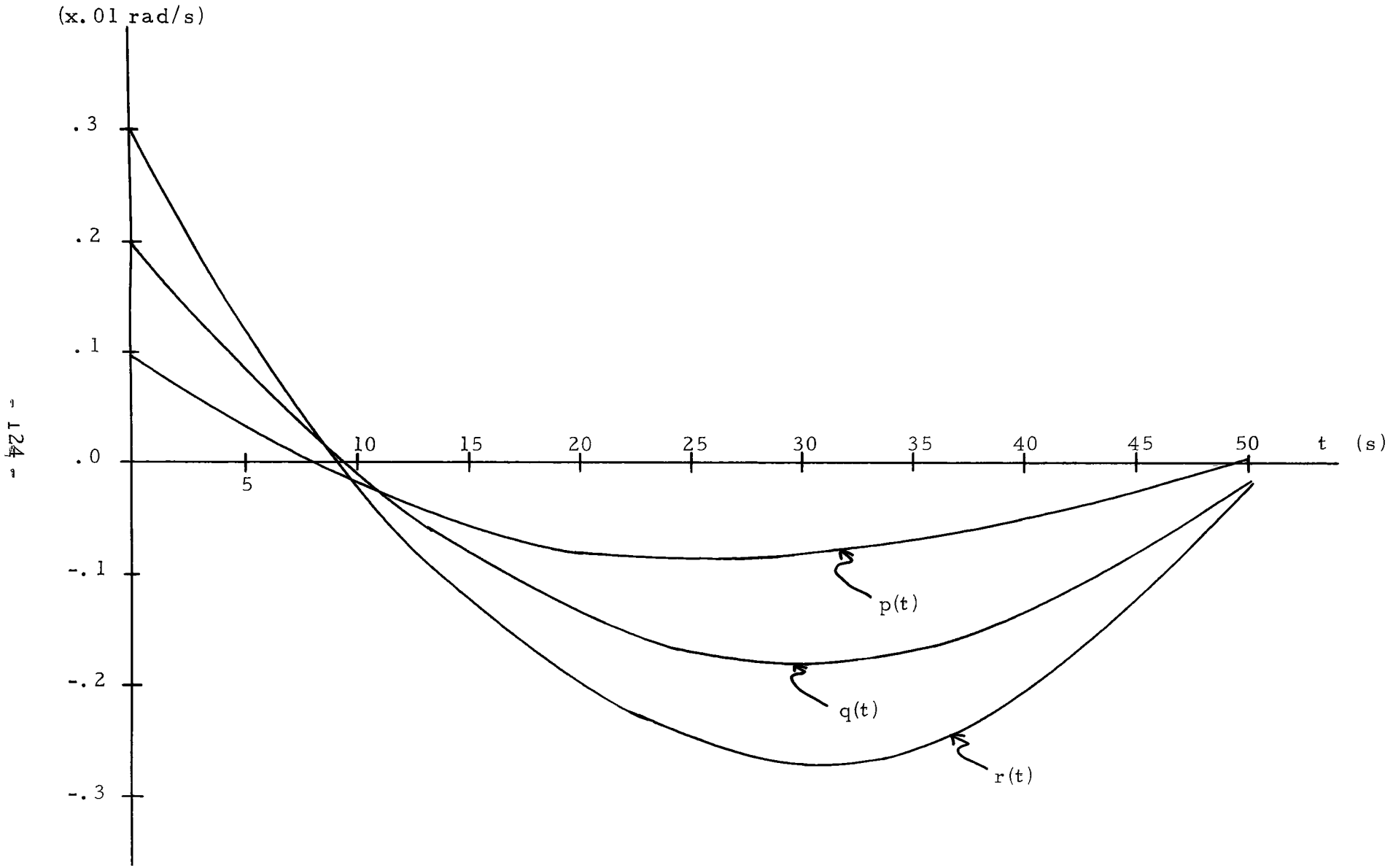


Fig. 4.2.25 Body Rates for Problem 4.2.D

4.3 Gyrotorquer Attitude Control.

In this section an example of attitude control using only gyrotorquers is given. As in problem 4.2.A, the objective is to drive the system to the origin from an initial disturbance in the Euler angles. The non-saturating case will be considered. Other variations of the problem could be formulated and solved but due to the computational overhead (as a result of the complexity of the model), this was not feasible. However, the particular case considered does illustrate the type of results to be expected using gyrotorquer control.

Problem 4.3.A: The problem considered is stated as follows:

$$\underset{u\in U^3}{\text{minimize}}\ J = \left[I+\tfrac{1}{2}R_1(p^2+q^2+r^2+\phi^2+\theta^2+\psi^2) + \tfrac{1}{2}R_2(\delta_x^2+\delta_y^2+\delta_z^2) \right]\Big|_{t=t_f}$$

subject to (4.3.1) for $t\in[0\ ,t_f]$, where;

$$\dot{p} = A_1 r + A_2 qr + A_3\Omega_x rs\delta_x + A_4\Omega_y qs\delta_y + A_5\Omega_y s\delta_y + A_6 u_3\Omega_z c\delta_z$$

$$\dot{q} = B_1 pr + B_2 u_1\Omega_x c\delta_x + B_3\Omega_y ps\delta_y + B_4\Omega_z rs\delta_z + B_5\Omega_z c\delta_z$$

$$\dot{r} = C_1 p + C_2 qp + C_3 p\Omega_x s\delta_x + C_4\Omega_x s\delta_x + C_5 u_2\Omega_y c\delta_y + C_6\Omega_z qs\delta_z$$

$$\dot{\phi} = p$$

$$\dot{\theta} = q$$

$$\dot{\psi} = r \qquad (4.3.1)$$

$$\dot{\delta}_x = u_1$$

$$\dot{\delta}_y = u_2$$

$$\dot{\delta}_z = u_3$$

$$\dot{I} = \tfrac{1}{2}\lambda_1(p^2+q^2+r^2+\phi^2+\theta^2+\psi^2) + \tfrac{1}{2}\lambda_2(u_1^2+u_2^2+u_3^2)$$

with initial conditions and constants:

$p(0)=.00$ rad/s	$\phi(0)=.02$ rad	$\delta_x(0)=.00$ rad
$q(0)=.00$ rad/s	$\theta(0)=.04$ rad	$\delta_y(0)=.00$ rad
$r(0)=.00$ rad/s	$\psi(0)=.06$ rad	$\delta_z(0)=.00$ rad

$R_1=4\times10^4$ $\quad R_2=4\times10^4$ $\quad \lambda_1=5$

$\lambda_2=\cdot1$ $\quad t_f=50$ s (4.3.2)

All the system constants are as defined for (2.2.21) and the product $C_\alpha\Omega_\alpha=.8/s$, $\alpha=x, y, z$. This yields reasonable physical parameters. Again, using PMP it is easily verified that the optimal controls for this problem are indeed continuous functions of time. Therefore, as with the previous problems, when applying the CGD, the control gradients in the space of continuous functions are defined using (3.4.53). Similarly, the co-state differential equations and their terminal conditions are easily obtained by applying (4.1.1) and (4.1.2). The results are displayed in Figures 4.3.1-4.3.5. A behaviour similar to that of the non-saturating flywheel control is obtained. This is as expected because of the similarity of the dynamics.

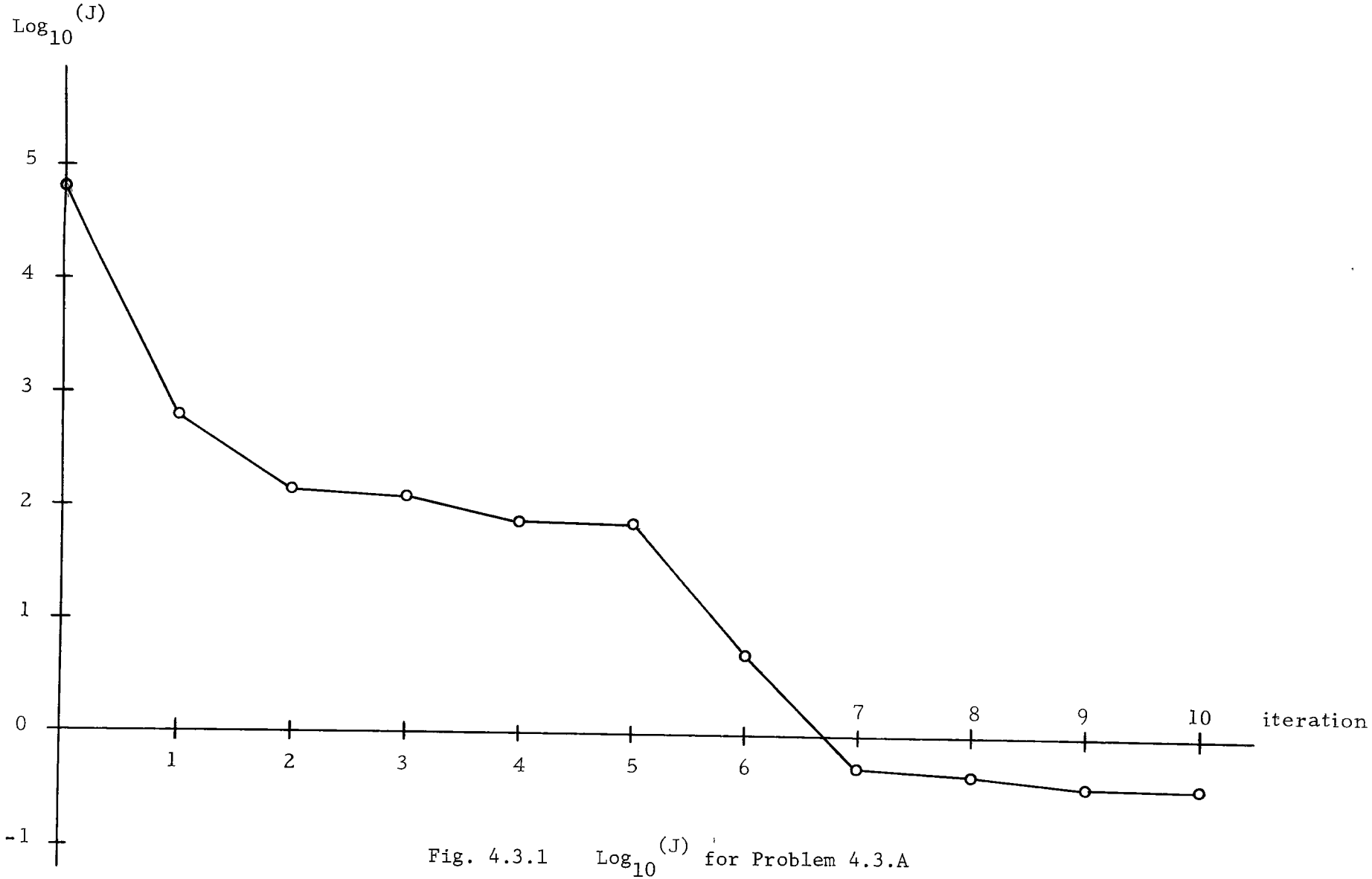


Fig. 4.3.1 $\text{Log}_{10}^{(J)}$ for Problem 4.3.A

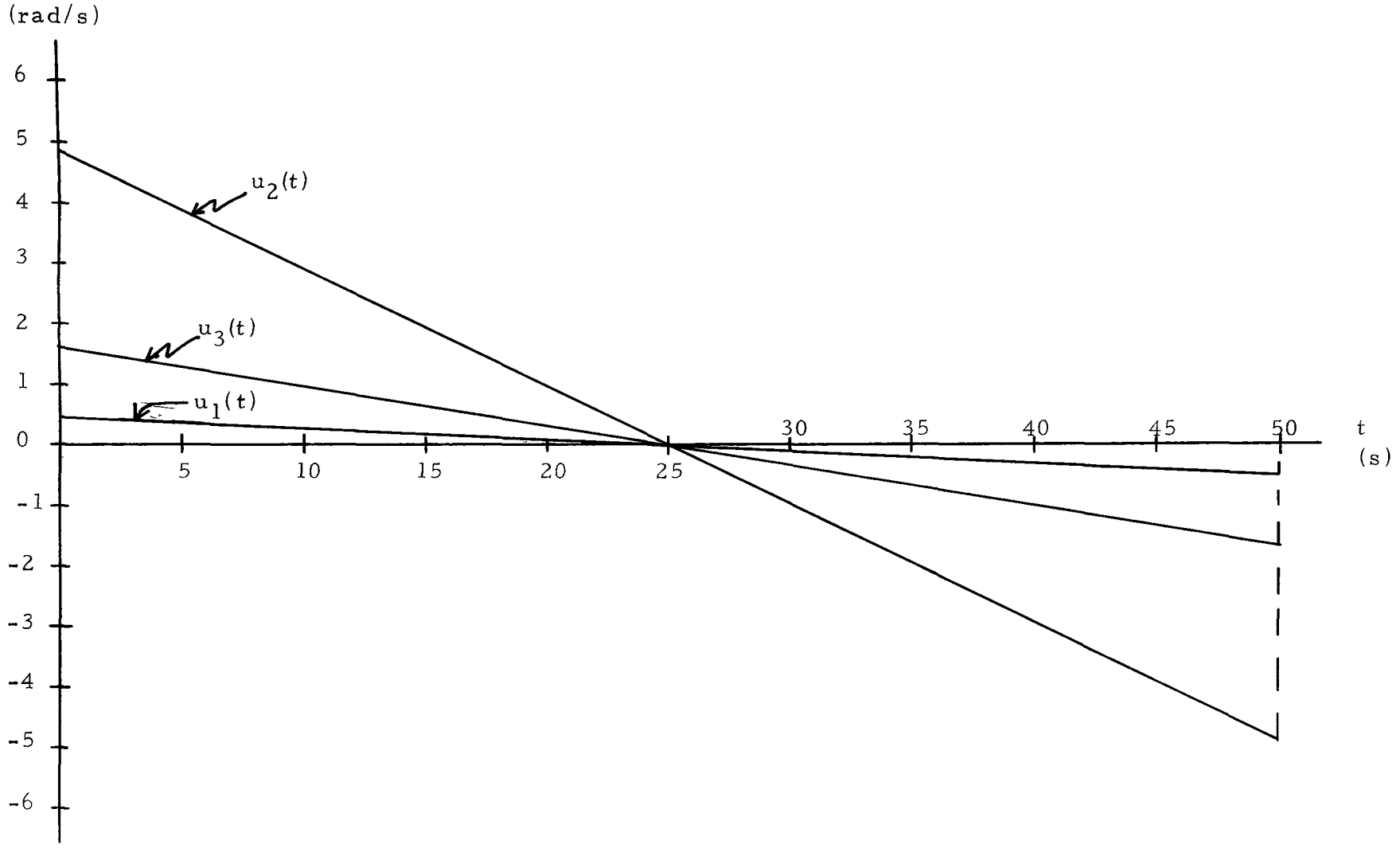


Fig. 4.3.2 Controls for Problem 4.3.A

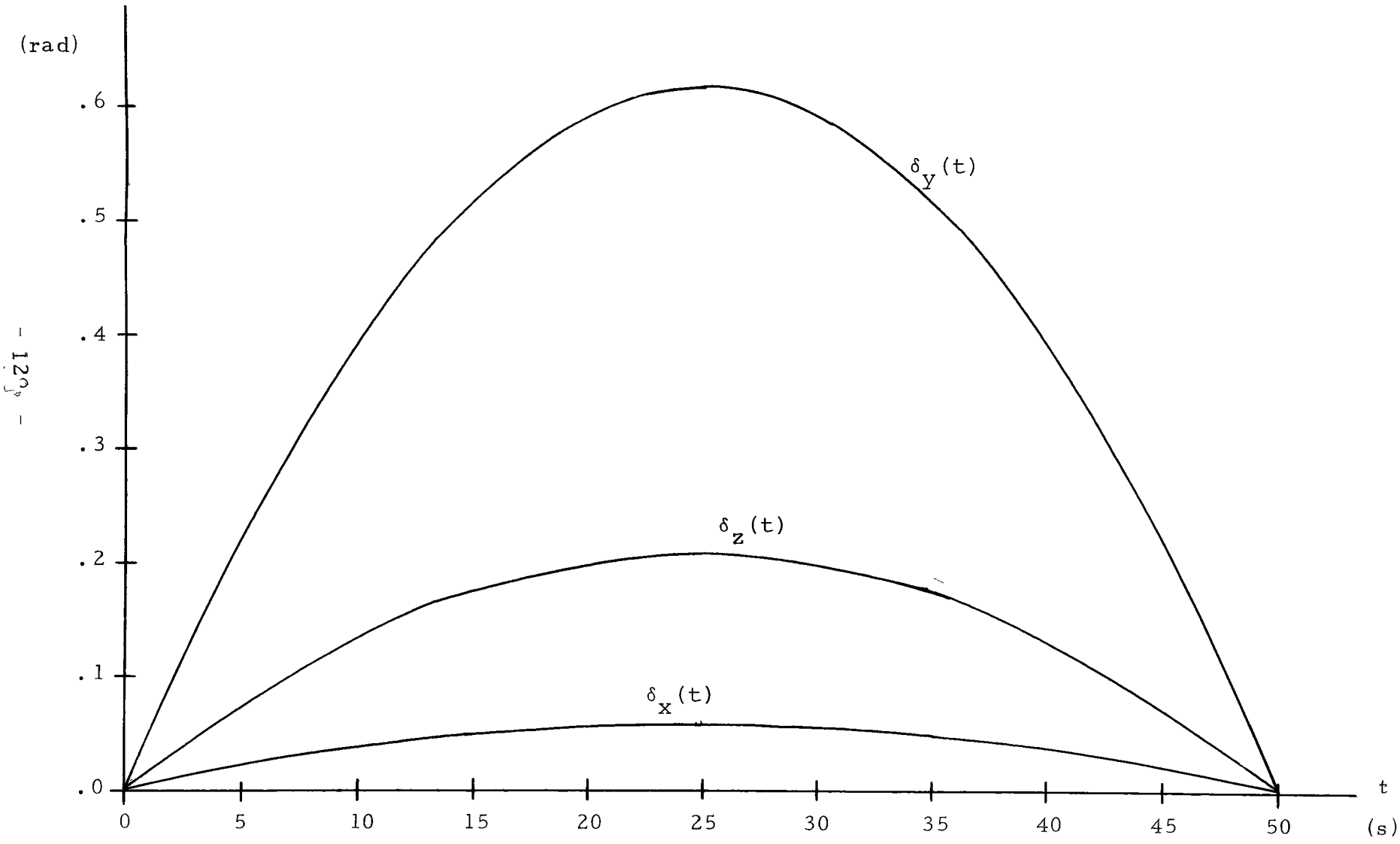


Fig. 4.3.3 Gimbal Angles for Problem 4.3.A

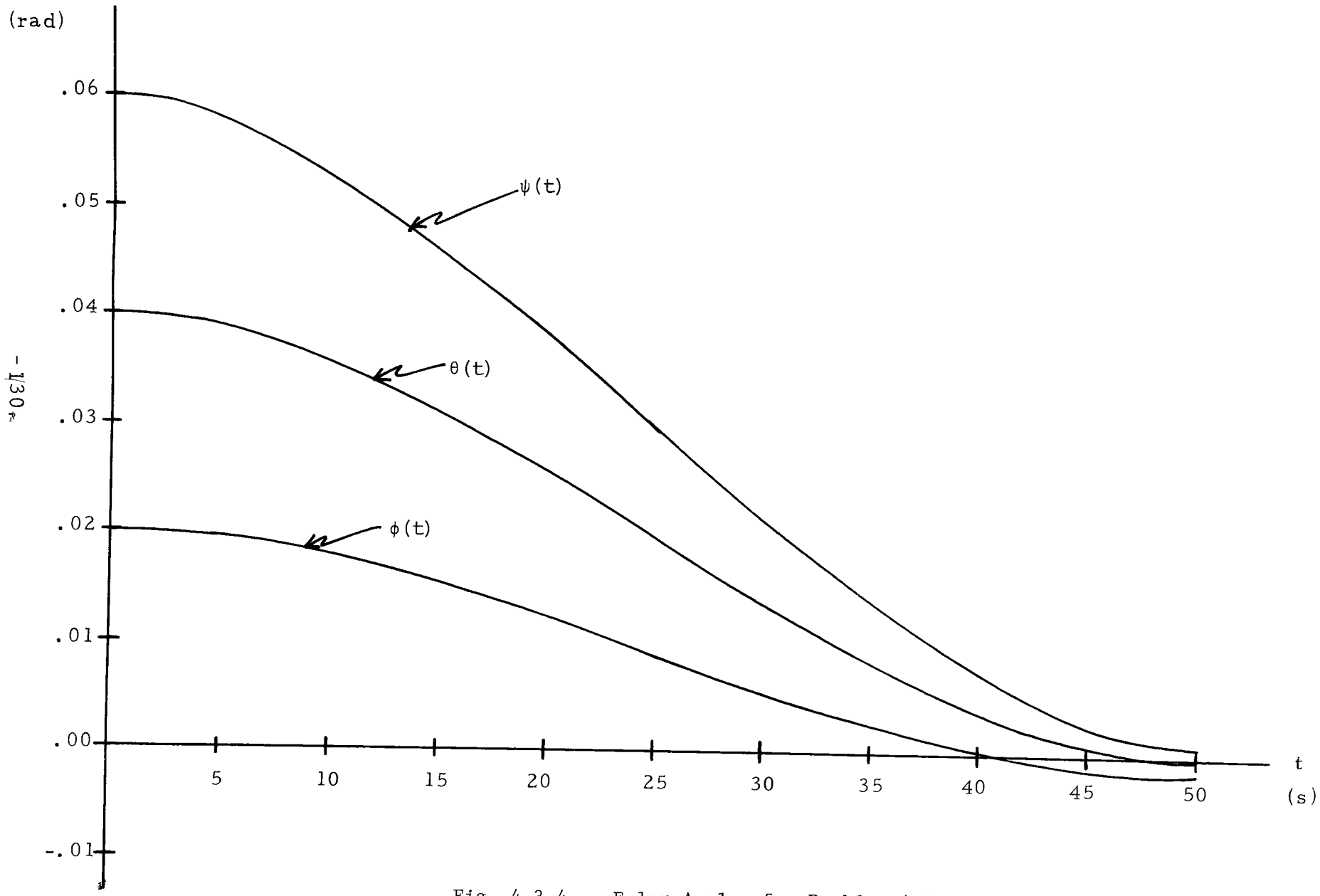


Fig. 4.3.4 Euler Angles for Problem 4.3.A

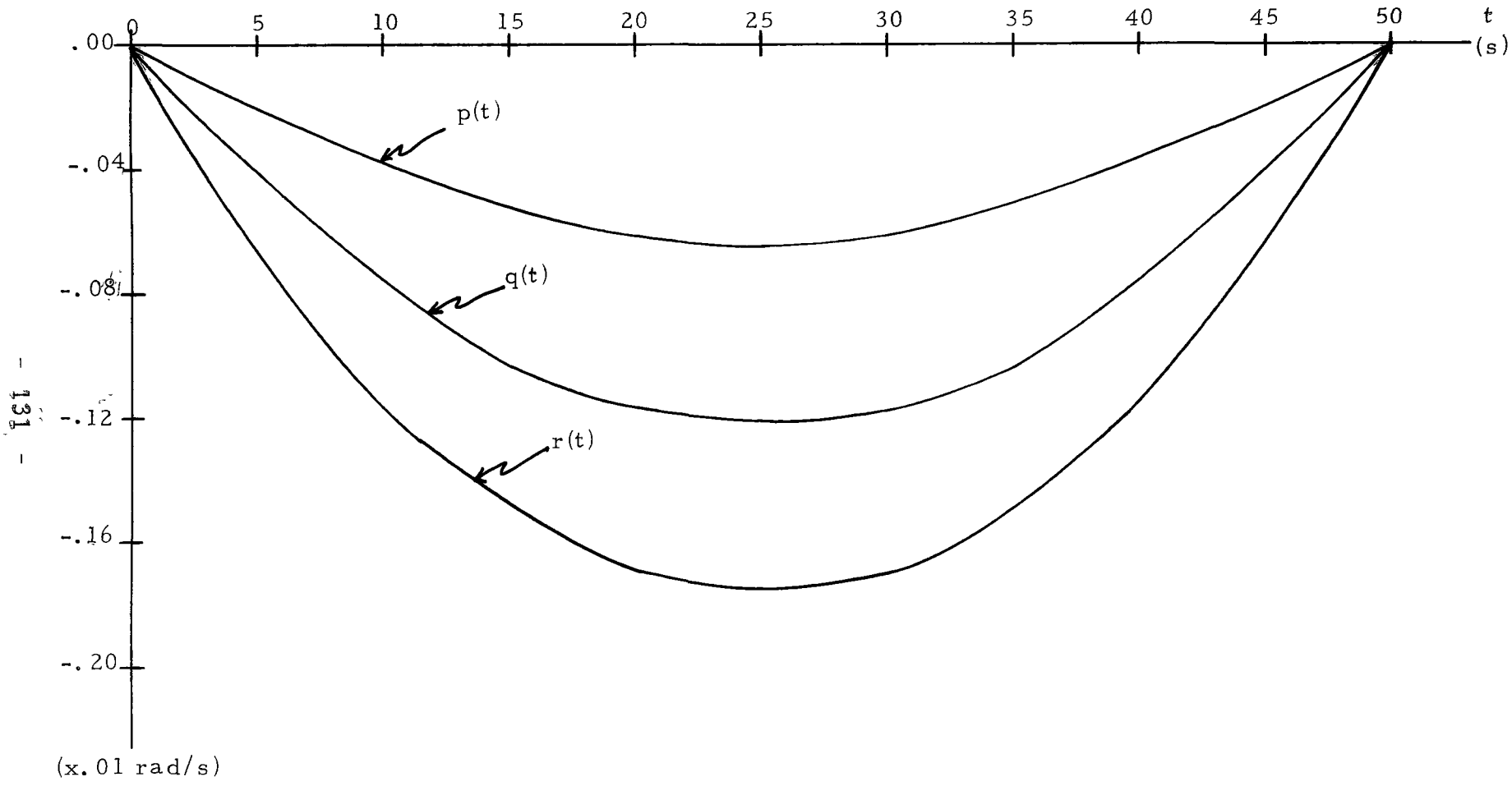


Fig. 4.3.5 Body Rates for Problem 4.3.A

Summary.

In this chapter, specific examples of satellite attitude control problems have been solved. To begin with, the use of reaction jets was studied, with the general observation that on-off jets are best suited to correct large disturbances or to drive the body rates to the origin regardless of the Euler angles. Then, flywheel control was studied with particular emphasis placed on the use of non-saturating flywheels. It was shown, that in the non-saturating mode, flywheel control can efficiently correct disturbances in the Euler angles and furthermore, the controls are effectively independent of one another. This implies, that each of the three flywheels is responsible for the controllability of its respective Euler angle. An example using saturating flywheels then followed and showed that flywheels can be used to correct disturbances in the body rates. Any large disturbance in the body rates must be corrected by the reaction jets because otherwise, saturation of the flywheels will result. Finally, an example using gyrotorquer control was given to demonstrate the nature of the results, to be expected with this type of control, which are similar to the case using flywheels.

Synthesis Of Open-Loop And Feedback Controls.

Thus far, satellite attitude control has been considered from a variety of viewpoints, ranging from the formulation to the solution of specific optimal control problems. As mentioned earlier, non-saturating flywheel control is of particular interest. In the previous chapter, optimal controls were obtained for a variety of cases. However, the optimal controls obtained apply only to the specific examples treated. This implies, that for each new problem it is necessary to recompute the optimal control - a rather unsatisfactory situation. The intent of this chapter is to investigate the possibility of approximating the optimal controls with sub-optimal controls, which are realized via a simple specification. Such an approach would form the basis for practical applications.

5.1 Open-Loop Control.

In the previous chapter, a number of optimal attitude control problems using non-saturating flywheels were solved. From their results it can be observed that the optimal controls are essentially linear functions of time and can be expressed as simple expressions of the form:

$$u^0(t) = k\left[1-2(t-t_0)/(t_f-t_0)\right] \qquad (5.1.1)$$

for $t\varepsilon[t_0,t_f]$ and where k is an appropriate constant. To investigate the validity of this observation, it was decided to solve problem 4.2.A with controls of the form given in equation (5.1.1). Since each of the three controls u_i^0, i=1,2,3 has a different value of k; the problem essentially consists of finding the optimal values for three constants. The results will permit us to evaluate the system's behaviour with simple open-loop controls. The problem solved here is the following:

Problem 5.1.A

$$\underset{k_1,k_2,k_3\varepsilon R}{\text{minimize}}\ J = \left[I+\tfrac{1}{2}R_1(p^2+q^2+r^2+\phi^2+\theta^2+\psi^2) + \tfrac{1}{2}R_2(\Omega_x^2+\Omega_y^2+\Omega_z^2)\right]\Big|_{t=t_f}$$

subject to (5.1.2) for $t\varepsilon[t_0,t_f]$, where:

$$\begin{aligned}
\dot{p} &= A_1 r + A_2 qr + A_3 u_1^0 + A_4\Omega_y r + A_5\Omega_z q\\
\dot{q} &= B_1 pr + B_2\Omega_x r + B_3 u_2^0 + B_4\Omega_z p\\
\dot{r} &= C_1 p + C_2 qp + C_3\Omega_x q + C_4\Omega_y p + C_5 u_3^0\\
\dot{\phi} &= p\\
\dot{\theta} &= q\\
\dot{\psi} &= r\\
\dot{\Omega}_x &= u_1^0\\
\dot{\Omega}_y &= u_2^0\\
\dot{\Omega}_z &= u_3^0\\
\dot{I} &= \tfrac{1}{2}\lambda_1(p^2+q^2+r^2+\phi^2+\theta^2+\psi^2) + \tfrac{1}{2}\lambda_2(u_1^{0\,2}+u_2^{0\,2}+u_3^{0\,2})\\
u_i^0(t) &= k_i\left[1-2(t-t_0)/(t_f-t_0)\right],\ i=1,2,3
\end{aligned} \qquad (5.1.2)$$

All the system constants are the same as those of problem 4.2.A. To find the optimal values of k_1, k_2 and k_3, a parameter search was performed using the parameter optimization package developed by Birta [15]. The following optimal values were obtained:

$$k_1 = 2.828, \qquad k_2 = 5.767, \qquad k_3 = 8.621 \tag{5.1.3}$$

with final cost J=1.23, which is identical to that obtained in problem 4.2.A. At this point, it should be recalled, that $u_1^0(t)$ controls $\phi(t)$, $u_2^0(t)$ controls $\theta(t)$ and $u_3^0(t)$ controls $\psi(t)$ (see problem 4.2.B). Thus, it may be possible to relate the constants to the initial Euler angles. Observe also that:

$$\frac{k_1}{\phi_0} = \frac{2.828}{.02} = 141.4$$

$$\frac{k_2}{\theta_0} = \frac{5.767}{.04} = 144.2 \tag{5.1.4}$$

$$\frac{k_3}{\psi_0} = \frac{8.621}{.06} = 143.7$$

Hence;

$$\frac{k_1}{\phi_0} \doteq \frac{k_2}{\theta_0} \doteq \frac{k_3}{\psi_0} \doteq k_{av} = \left[\frac{k_1}{\phi_0} + \frac{k_2}{\theta_0} + \frac{k_3}{\psi_0}\right]/3 \tag{5.1.5}$$

From the above observations, it appears that a suitable open-loop control could be obtained directly from knowledge of the initial Euler angles. To verify this, controls of the following form were used:

$$u_1^0(t) = k_{av}\,\phi_0\left[1-2(t-t_0)/(t_f-t_0)\right]$$

$$u_2^0(t) = k_{av}\,\theta_0\left[1-2(t-t_0)/(t_f-t_0)\right]$$

$$u_3^0(t) = k_{av}\,\psi_0\left[1-2(t-t_0)/(t_f-t_0)\right] \qquad (5.1.6)$$

The graphs shown in Figures 5.1.1-5.1.2, illustrate the results obtained by integrating (5.1.2) with $u_i^0(t)$, i=1,2,3 as given in (5.1.6) with k_{av}=143. From these graphs, one can observe, that the open-loop controls, yield results practically identical to the optimal results corresponding to problem (4.2.A). Thus a model for synthesizing "practically" optimal open-loop controls for non-saturating flywheels has been obtained.

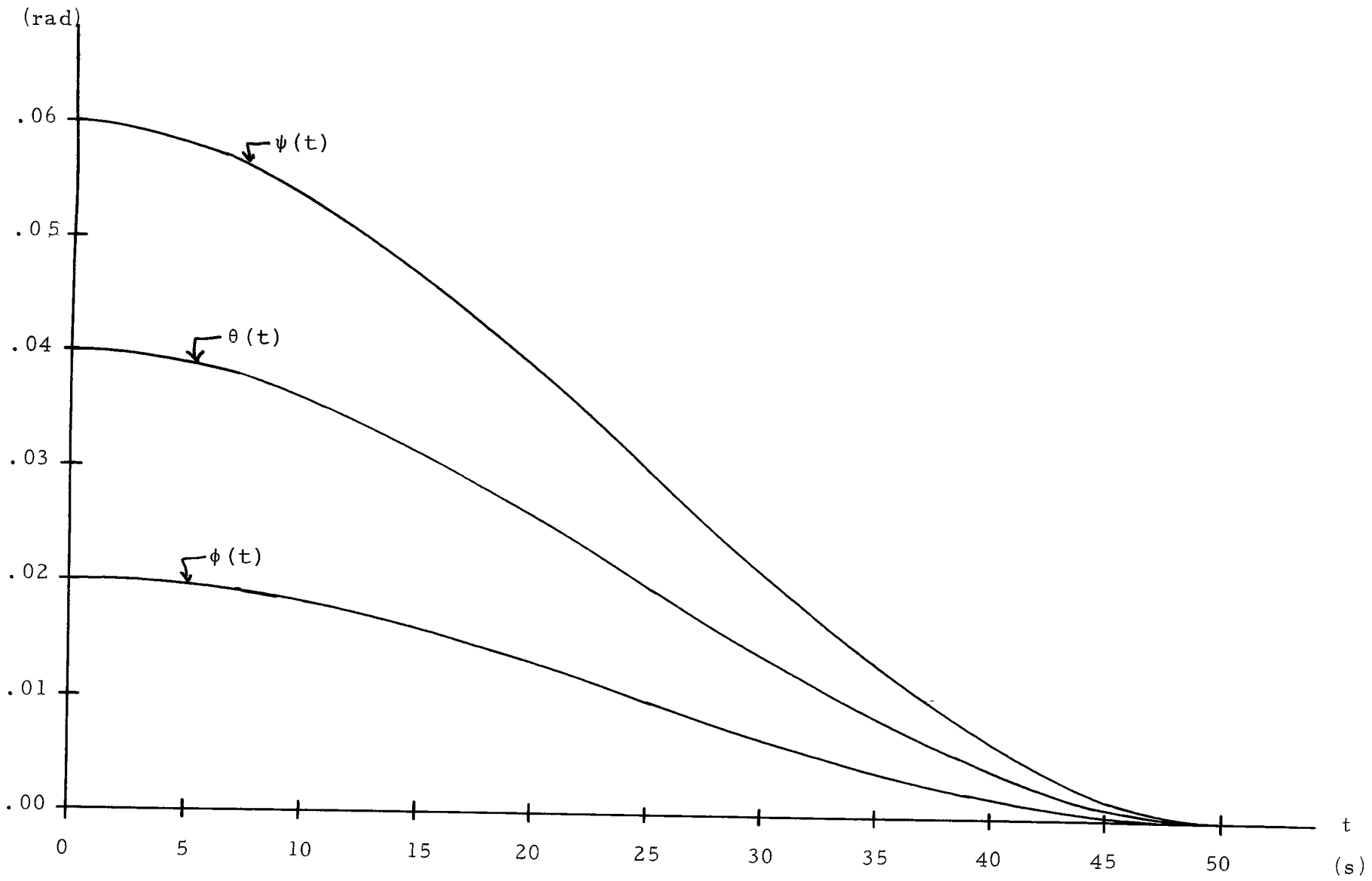


Fig. 5.1.1 Euler Angles for Problem 5.1.A

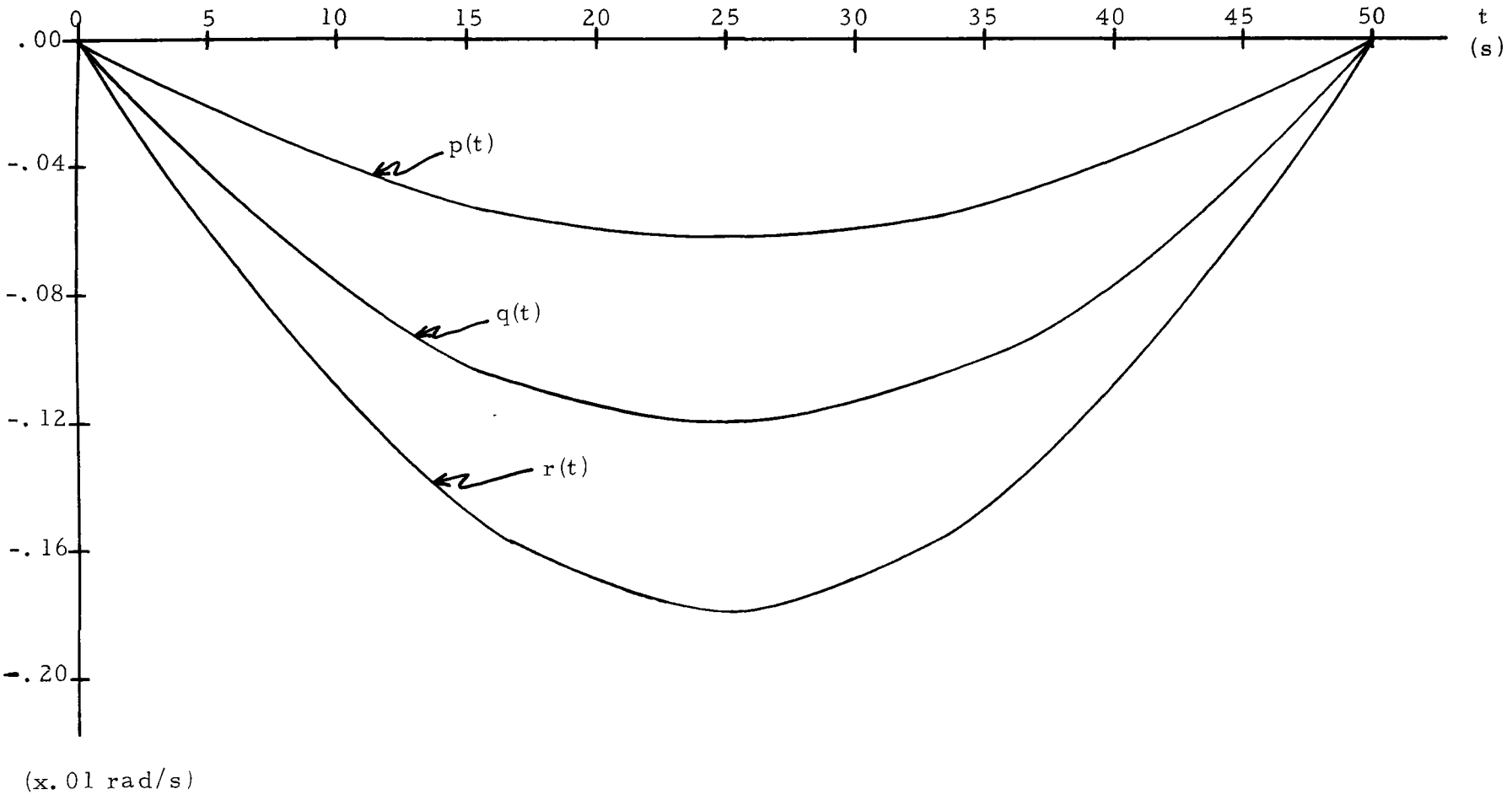


Fig. 5.1.2 Body Rates for Problem 5.1.A

5.2 Feedback Controls.

In practice, the synthesis of feedback controls is preferred to that of open-loop controls because of the inherently more stable nature.

Problem 5.2.A: In this problem, an attempt is made to design simple feedback controls that will return to the origin, initial disturbances in the Euler angles. A fixed terminal time is assumed. The feedback controls are synthesized from observations of the optimal controls and the behaviour of the Euler angles when these are applied. The optimal controls have the following form;

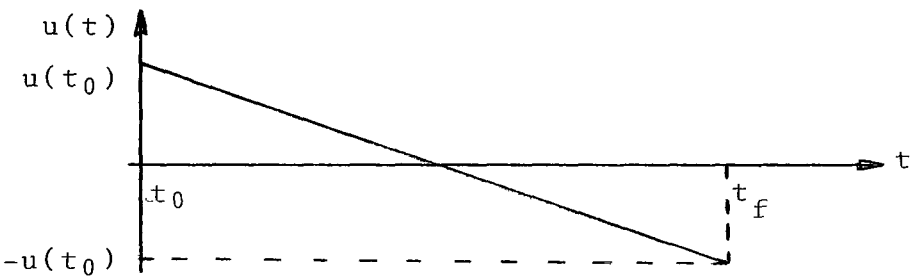


Fig. 5.2.1 Typical Form of the Optimal Controls.

and the optimal trajectories of the Euler angles have the following form:

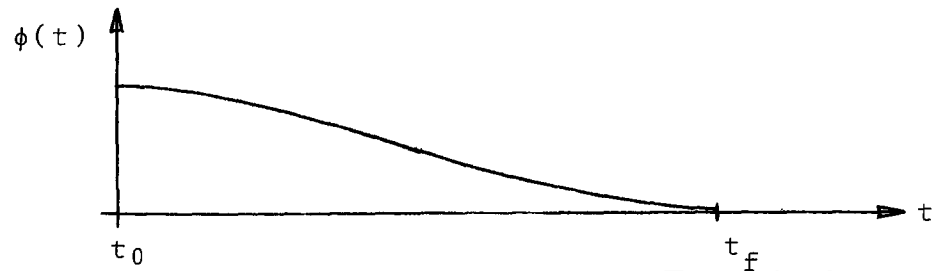


Fig. 5.2.2 Typical Optimal Trajectories of the Euler Angles.

From the above obsevations it was decided to try feedback controls of the form:

$$u_1^f(t) = k_1(\phi+\phi_0/2) + k_2\phi(\phi-\phi_0)\cos\left[\pi(t-t_0)/(t_f-t_0)\right] \quad (5.2.1)$$
$$\text{for } t\varepsilon[t_0,t_f]$$

Similar expressions can be written for $u_2^f(t)$ and $u_3^f(t)$. From problem 5.1.A, it is known, that the open-loop controls $u_i^0(t)$, i=1,2,3, yield practically optimal solutions. Therefore, our problem may be stated as;

$$\underset{k_1,k_2\varepsilon R}{\text{minimize}}\ J = \left[I\right]\Big|_{t=t_f}$$

subject to (5.2.2) for $t\varepsilon[0,t_f]$, where;

$$\begin{aligned}
\dot{p} &= A_1r + A_2qr + A_3u_1^f + A_4\Omega_y r + A_5\Omega_z q\\
\dot{q} &= B_1pr + B_2\Omega_x r + B_3u_2^f + B_4\Omega_y p\\
\dot{r} &= C_1p + C_2qp + C_3\Omega_x q + C_4\Omega_y p + C_5u_3^f\\
\dot{\phi} &= p\\
\dot{\theta} &= q\\
\dot{\psi} &= r\\
\dot{\Omega}_x &= u_1^f\\
\dot{\Omega}_y &= u_2^f\\
\dot{\Omega}_z &= u_3^f\\
\dot{I} &= (u_1^f-u_1^0)^2 + (u_2^f-u_2^0)^2 + (u_3^f-u_3^0)^2
\end{aligned} \quad (5.2.2)$$

with the following initial conditions:

$p(0)=.00$ rad/s $\quad \phi(0)=.02$ rad $\quad \Omega_x(0)=.00$rad/s $\quad I(0)=0$

$q(0)=.00$ rad/s $\quad \theta(0)=.04$ rad $\quad \Omega_y(0)=.00$rad/s

$r(0)=.00$ rad/s $\quad \psi(0)=.06$ rad $\quad \Omega_z(0)=.00$rad/s $\quad (5.2.3)$

Again, a parameter search using Birta's software package

[15] was performed to find the optimal values of k_1 and k_2. These were found to be 239. and 56.1 respectively. With these values of k_1 and k_2, the system (5.2.2) was integrated and the results are shown in Figures 5.2.3-5.2.6.

From these graphs, it can be readily observed that the results obtained are indeed very close to the optimal solutions given in 4.2.A . Therefore, a fixed terminal time feedback controller has been designed. In the next problem a more general feedback controller is synthesized.

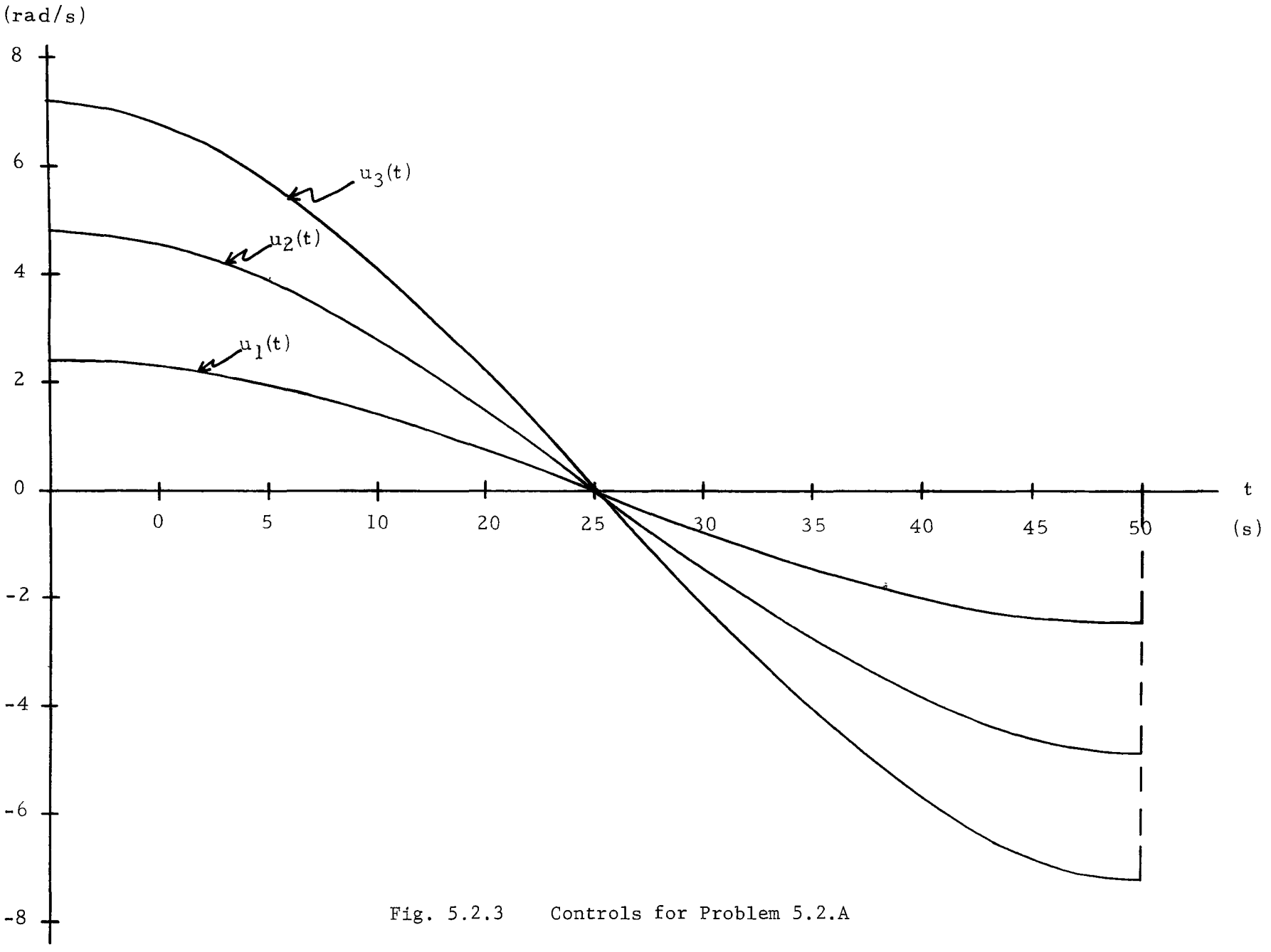


Fig. 5.2.3 Controls for Problem 5.2.A

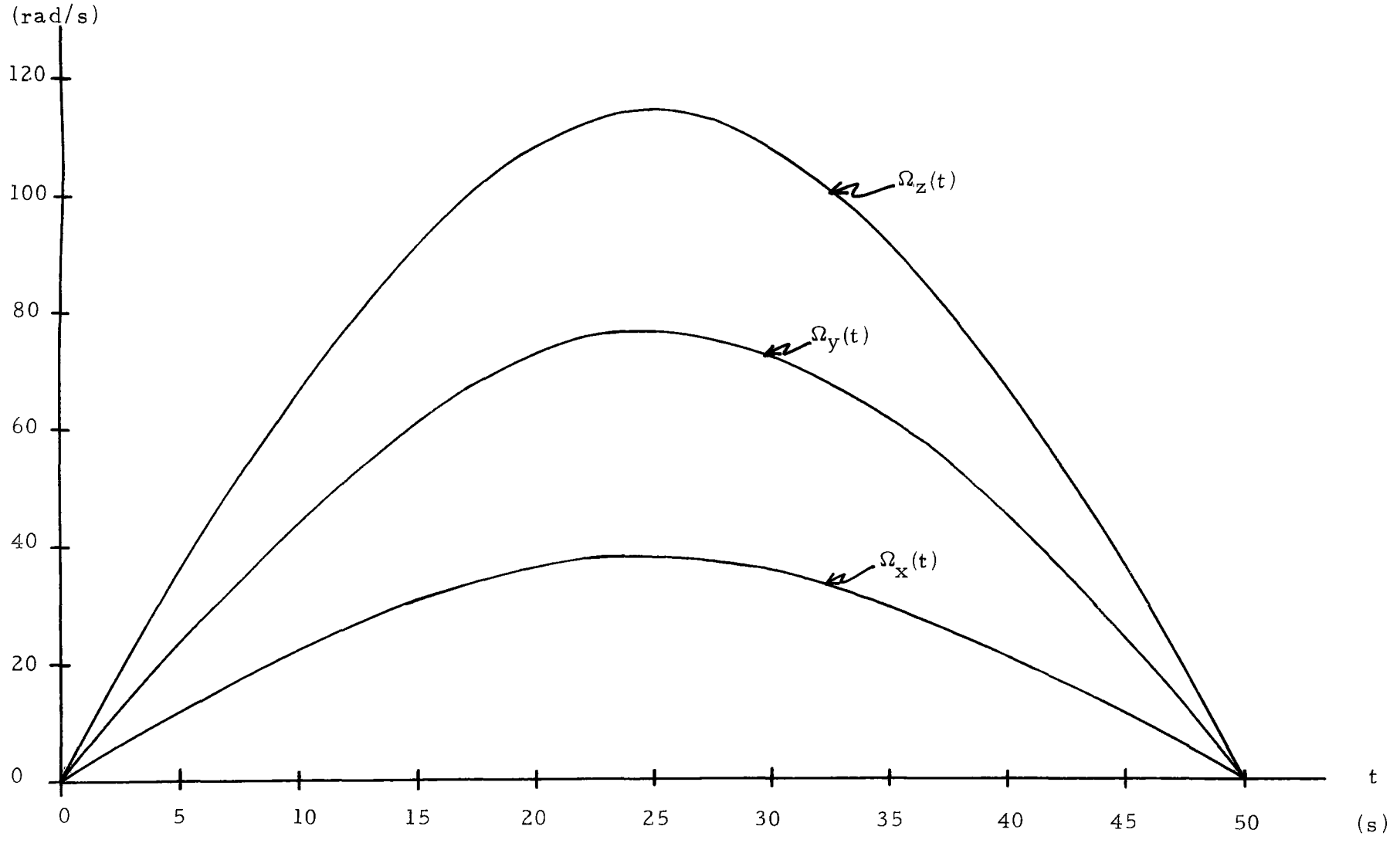


Fig. 5.2.4 Flywheel Angular Velocities for Problem 5.2.A

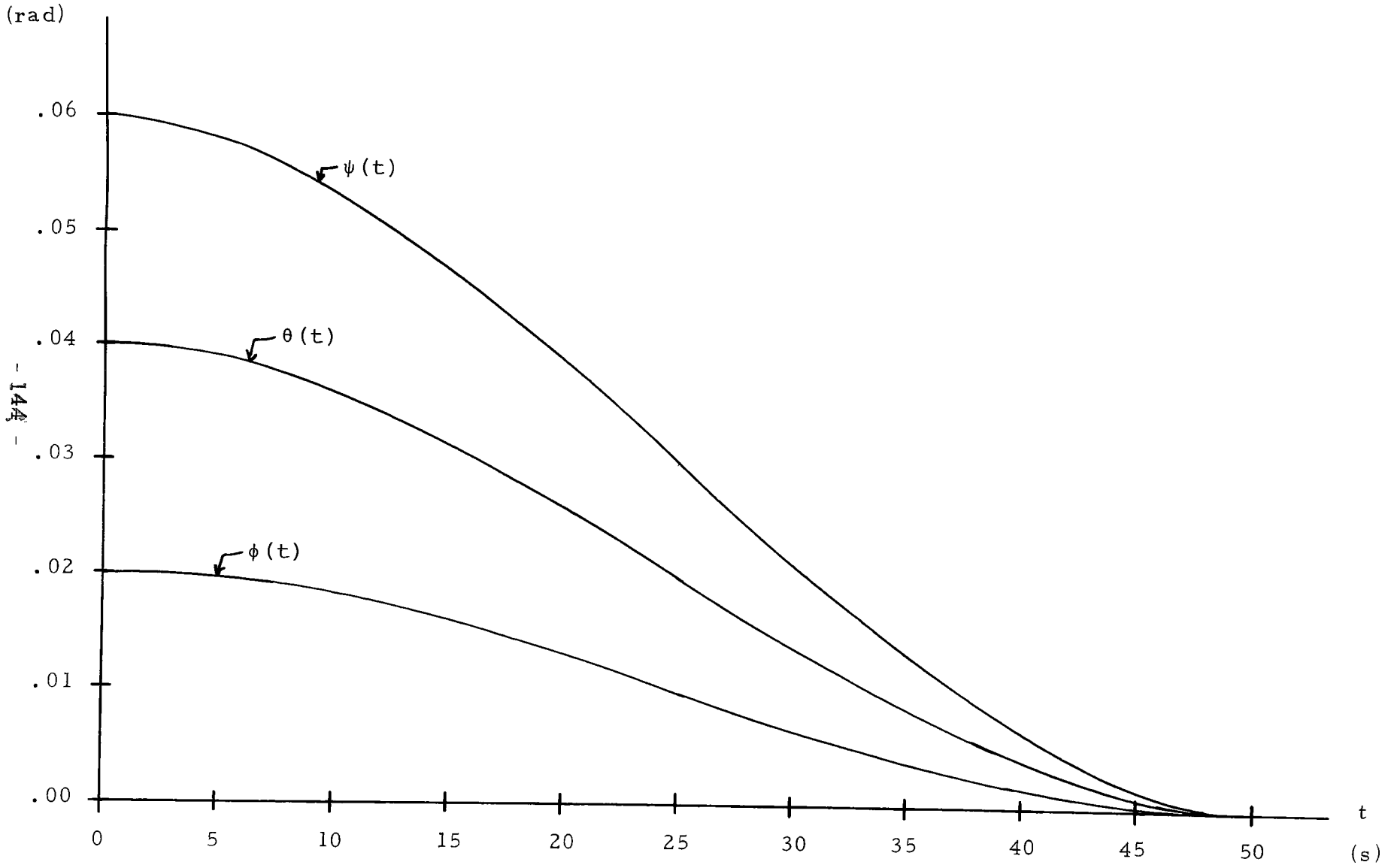


Fig. 5.2.5 Euler Angles for Problem 5.2.A

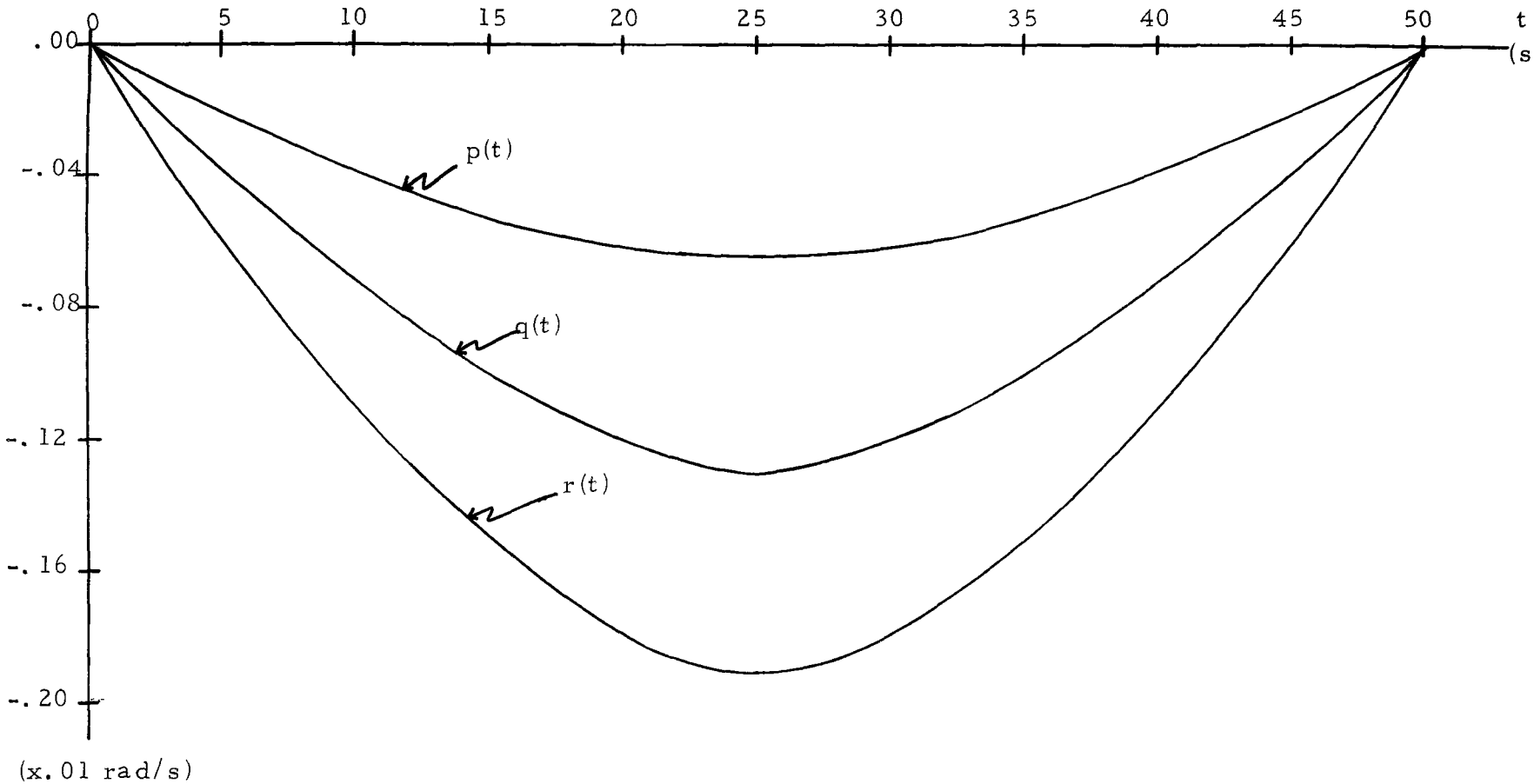


Fig. 5.2.6 Body Rates for Problem 5.2.A

Problem 5.2.B: In this problem, the synthesis of a time invariant feedback controller is considered. The basic assumption here is that the body rates are initially negligible and that the problem essentially consists of correcting initial disturbances in the Euler angles.

The feedback controls are assumed to be polynomial functions of the Euler angles, until the latter are within a specified distance from the origin. At that point, this feedback control policy is replaced by simple negative feedback proportional to the body rates. For example:

$$u_1^f = \begin{cases} [k_{11}(\frac{\phi}{\phi_0}) + k_{12}(\frac{\phi}{\phi_0})^2 + k_{13}(\frac{\phi}{\phi_0})^3] \cdot \text{sgn}(\phi_0); \text{ if } |\phi| \geq \rho \\ k_4 p \cdot \text{sgn}(\phi_0); \text{if } |\phi| < \rho \end{cases} \quad (5.2.4)$$

The controls u_2^f and u_3^f can be similarly described using the other Euler angles and body rates. As a first step in the solution of this problem, a fixed terminal time was assumed and a parameter search was performed to determine the optimal values of k_{ij}, i,j=1,2,3. The problem solved can be stated as follows:

$$\underset{k_{ij}\varepsilon R}{\text{minimize}}\ J = [\tfrac{1}{2}R_1(p^2+q^2+r^2+\phi^2+\theta^2+\psi^2) + \tfrac{1}{2}R_2(\Omega_x^2+\Omega_y^2+\Omega_z^2) + \tfrac{1}{2}R_3(u_1^{f^2}+u_2^{f^2}+u_3^{f^2})]\Big|_{t=t_f}$$

subject to (5.2.5) for $t\varepsilon[0,t_f]$, where:

$$\dot{p} = A_1 r + A_2 qr + A_3 u_1^f + A_4 \Omega_y r + A_5 \Omega_z q$$

$$\dot{q} = B_1 pr + B_2 \Omega_x r + B_3 u_2^f + B_4 \Omega_z p$$

$$\dot{r} = C_1 p + C_2 qp + C_3 \Omega_x q + C_4 \Omega_y p + C_5 u_3^f$$

$$
\begin{aligned}
\dot{\phi} &= p \\
\dot{\theta} &= q \\
\dot{\psi} &= r \\
\dot{\Omega}_x &= u_1^f \\
\dot{\Omega}_y &= u_2^f \\
\dot{\Omega}_z &= u_3^f
\end{aligned}
\qquad (5.2.5)
$$

The initial conditions were taken to be the same as those of problem 5.2.A and the constants are:

$$R_1 = 4\times10^4, \quad R_2 = .2, \quad R_3 = 20., \quad t_f = 50\ s, \quad \rho = 1\times10^{-3} \qquad (5.2.6)$$

After completion of the parameter search, the following optimal values were obtained:

$$
\begin{aligned}
k_{11} &= -29.83 & k_{12} &= 82.65 & k_{13} &= -50.94 \\
k_{21} &= -59.66 & k_{22} &= 165.3 & k_{23} &= -101.9 \\
k_{31} &= -89.49 & k_{32} &= 248.0 & k_{33} &= -152.8
\end{aligned}
\qquad (5.2.7)
$$

The constant k_4 (equ'n 5.2.4) was simply obtained by trial and error until convergence occurred after switching to the simple negative feedback. A final value of $k_4 = 1\times10^4$ was adopted. With the above constants, the differential equations in (5.2.5) were integrated for 100 seconds and the results obtained are shown in Figures 5.2.7-5.2.10. From these results it may be concluded that a stable feedback controller has been designed. This controller gives responses very similar to those obtained in problem 4.2.A, however, with somewhat higher maximum flywheel speeds.

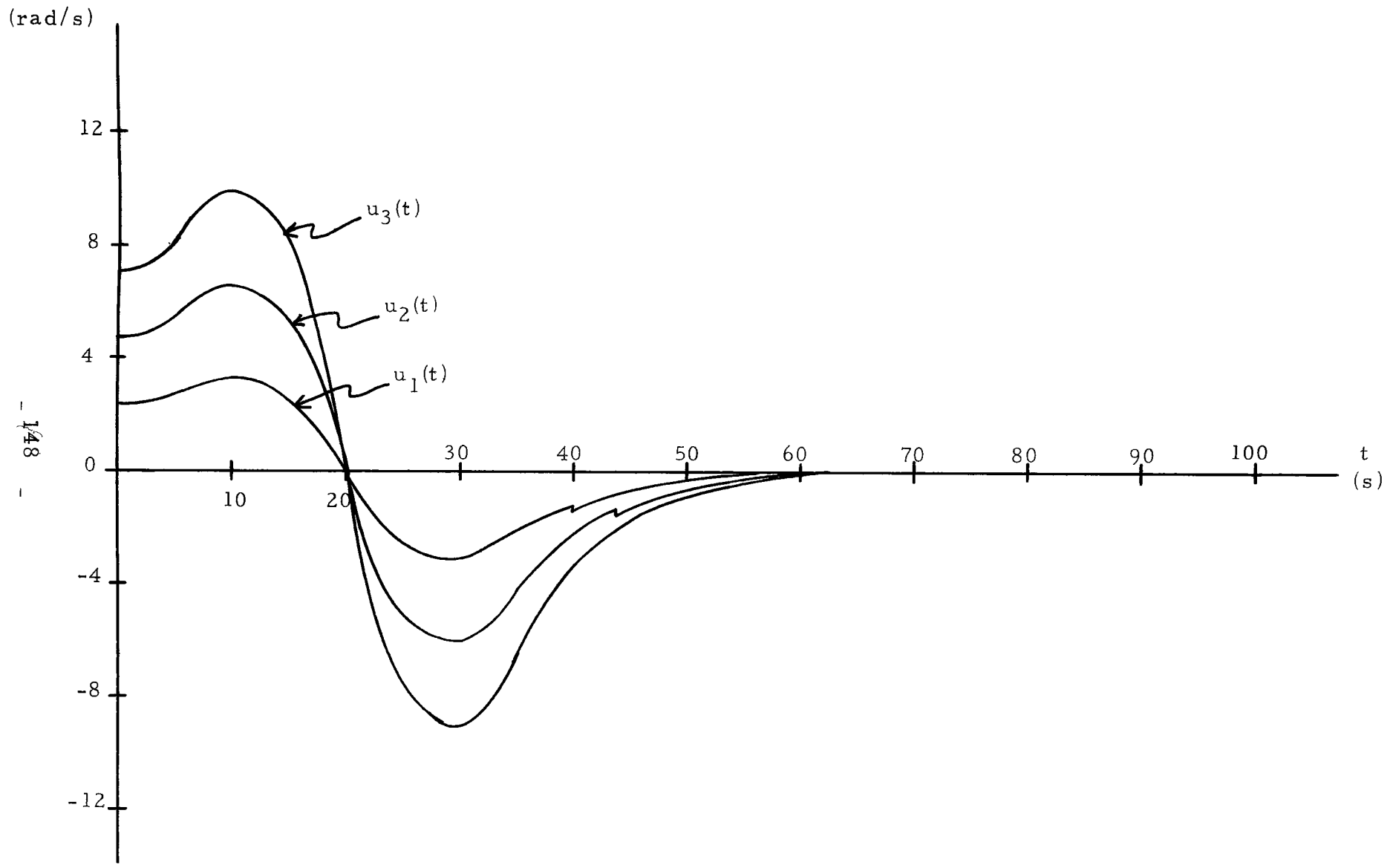


Fig. 5.2.7 Controls for Problem 5.2.B

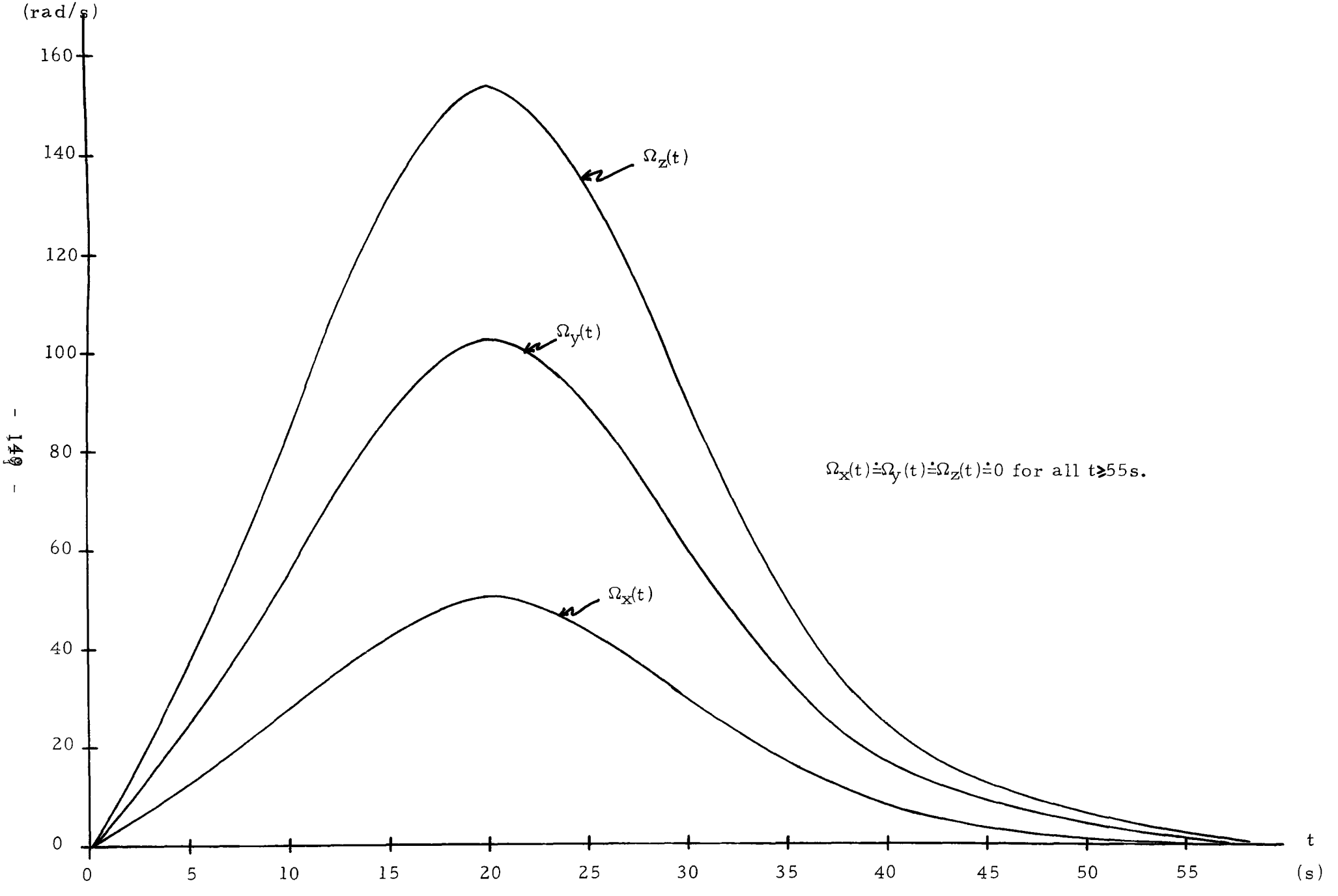


Fig. 5.2.8 Flywheel Angular Velocities for Problem 5.2.B

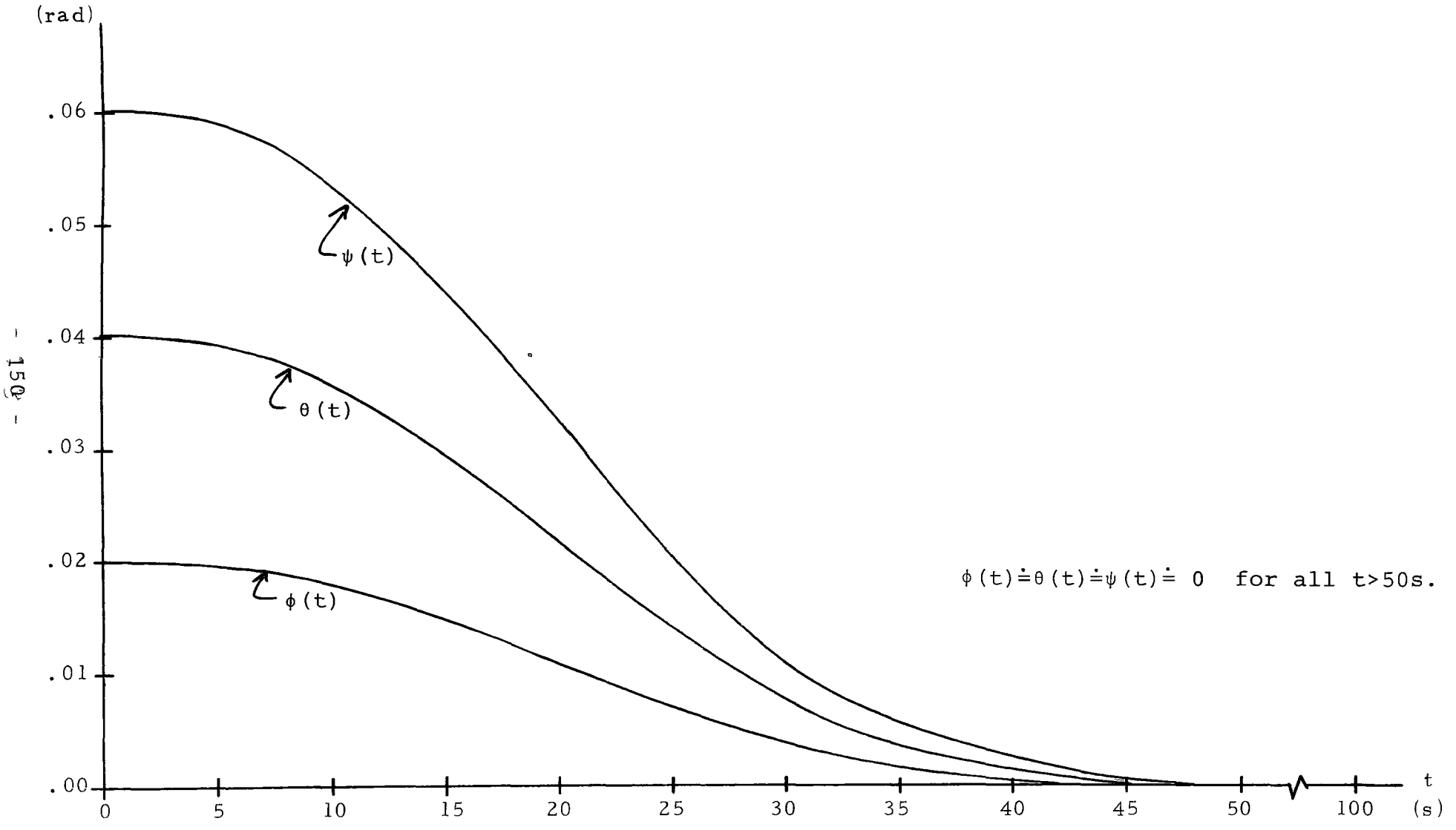
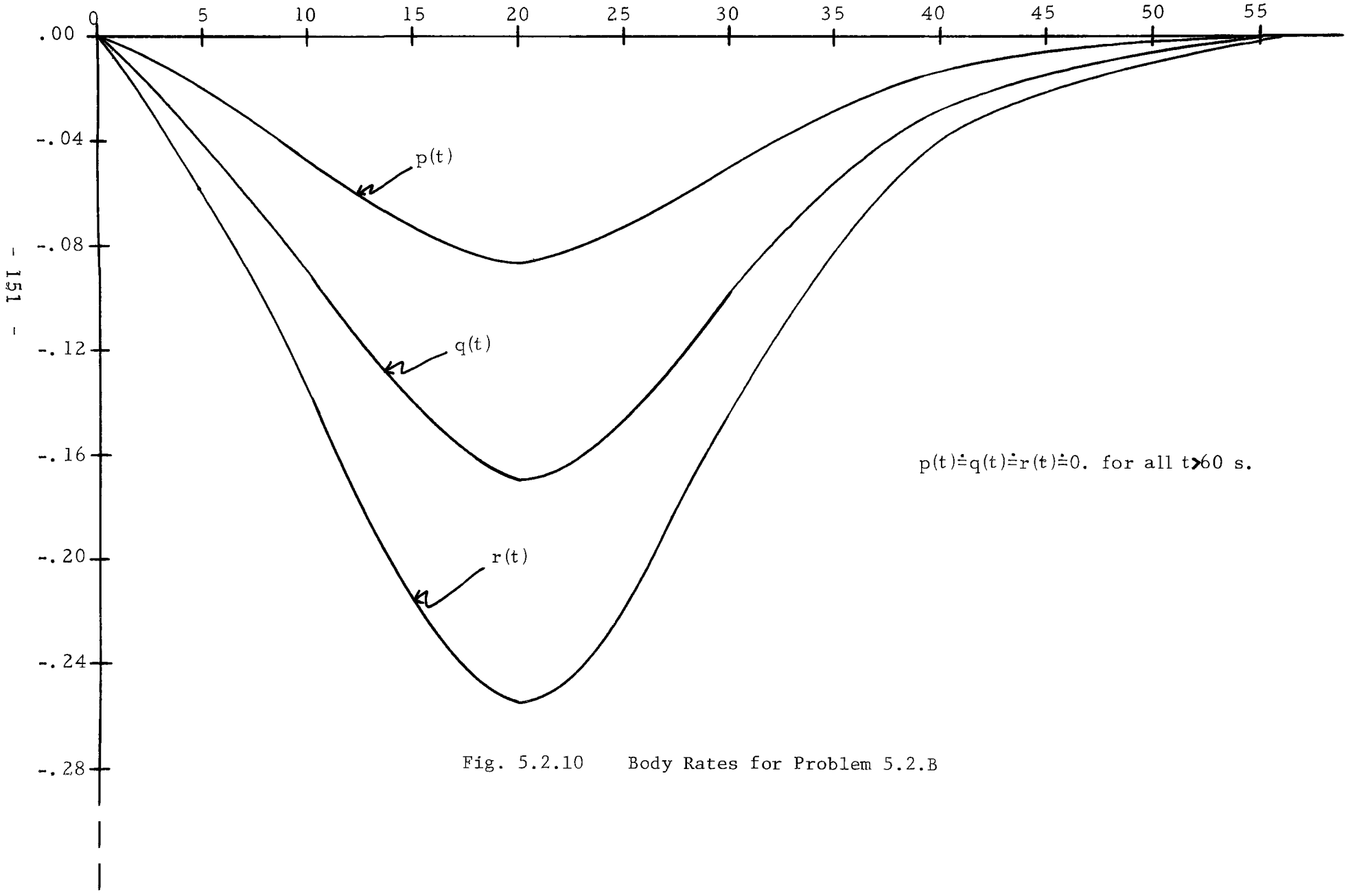


Fig. 5.2.9 Euler Angles for Problem 5.2.B



$p(t) \doteq q(t) \doteq r(t) \doteq 0.$ for all $t > 60$ s.

Fig. 5.2.10 Body Rates for Problem 5.2.B

Summary.

In this chapter, open-loop and feedback controllers have been synthesized for non-saturating flywheel control. The results obtained from these controllers are very close to those obtained from the direct solution of the optimal control problem, as discussed in Chapter Four.

CONCLUSIONS

In this thesis, optimal control theory has been successfully applied, to the study of certain aspects of satellite attitude control.

The mathematical model for a zero-momentum three-axis attitude controlled satellite was first derived and then used to formulate a set of optimal control problems. Various techniques available for the solution of optimal control problems were surveyed and based on the related observations, it was decided, that the CGD method was best suited for the optimal control problems to be solved in this study. Problems with piecewise continuous controls having, state and control constraints, as well as a free terminal time, can all be solved by this method.

A variety of optimal attitude control problems were then solved and the conclusions drawn from these are elaborated here. From problem 4.1.A, it appears, that using throttled jets, the system is controllable to the origin. However, in problem 4.1.B, when on-off jets were used, this is not the case and furthermore, the terminal state is very sensitive to the switching times. Thus, the use of reaction jets alone is not recommended; rather, they should be complemented by flywheels or gyrotorquers and used for desaturation purposes. When studying flywheel control,it was decided, to

study in particular the non-saturating mode. From the solution of problem 4.2.B, it can be concluded, that one particular flywheel serves to correct disturbances in one particular Euler angle.

Furthermore, for controllability, using only flywheels, it is necessary that all three flywheels be operational. From this conclusion, it was decided to attempt to generate, simple open-loop and feedback control policies, which are each dependent on a different Euler angle. These simple control policies were obtained by performing a parameter search and were found to give results near the optimal values. This could form the basis for a practical application.

Thus, it is the feeling of the author, that optimal control theory can be used to advantage, for the study and control, of attitude for satellites.

Appendix A

Vector Definitions.

Let Z be a matrix of dimension m x (m•M+1) defined by:

$$Z \equiv \begin{bmatrix} z_{01} & z_{11} & \cdots & z_{m\cdot M1} \\ z_{02} & & & \vdots \\ \vdots & & & \\ z_{0m} & & & z_{m\cdot M\,m} \end{bmatrix} \qquad \text{(A.A.1)}$$

Consider for example a case with two controls (m=2), each having one discontinuity (M=1), thus, the controls have the following form:

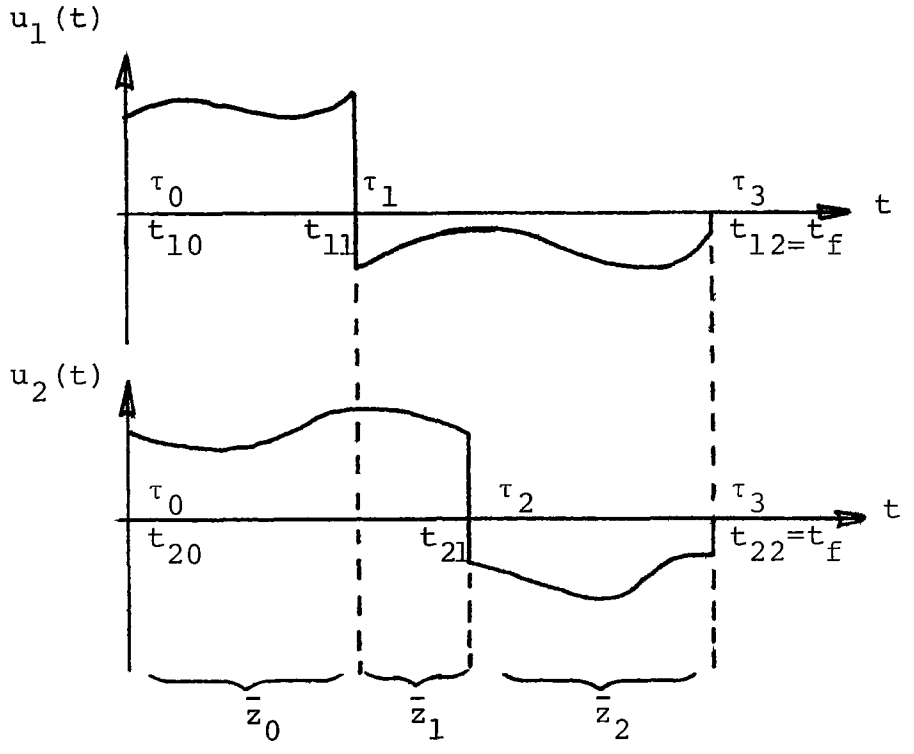


Fig. A.A.1 Typical Piecewise Continuous Controls.

Let: $\bar{z}_0 = \begin{bmatrix} 1 \\ \frac{1}{2} \end{bmatrix}$, $\bar{z}_1 = \begin{bmatrix} -1 \\ \frac{1}{2} \end{bmatrix}$, $\bar{z}_2 = \begin{bmatrix} -1 \\ -2 \end{bmatrix}$. (A.A.2)

Now define:

$$\bar{y}_{10}=\begin{bmatrix}1\\0\end{bmatrix}\ ,\quad \bar{y}_{11}=\begin{bmatrix}-1\\0\end{bmatrix}\ ,\quad \bar{y}_{20}=\begin{bmatrix}0\\\frac{1}{2}\end{bmatrix}\ ,\quad \bar{y}_{21}=\begin{bmatrix}0\\-2\end{bmatrix} \tag{A.A.3}$$

Thus, from the above definitions, for any 2x2 matrix denoted by v(t), it follows that:

$$\sum_{i=0}^{2x1}\int_{\tau_i}^{\tau_{i+1}} v(\sigma)\,\bar{z}_i\,d\sigma = \sum_{j=1}^{2}\sum_{k=0}^{1}\int_{t_{jk}}^{t_{jk+1}} v(\sigma)\bar{y}_{jk}d\sigma \tag{A.A.4}$$

A similar procedure can be applied for any m and M.

Appendix B.

Flow Chart of the Conjugate Gradient Descent Procedure.

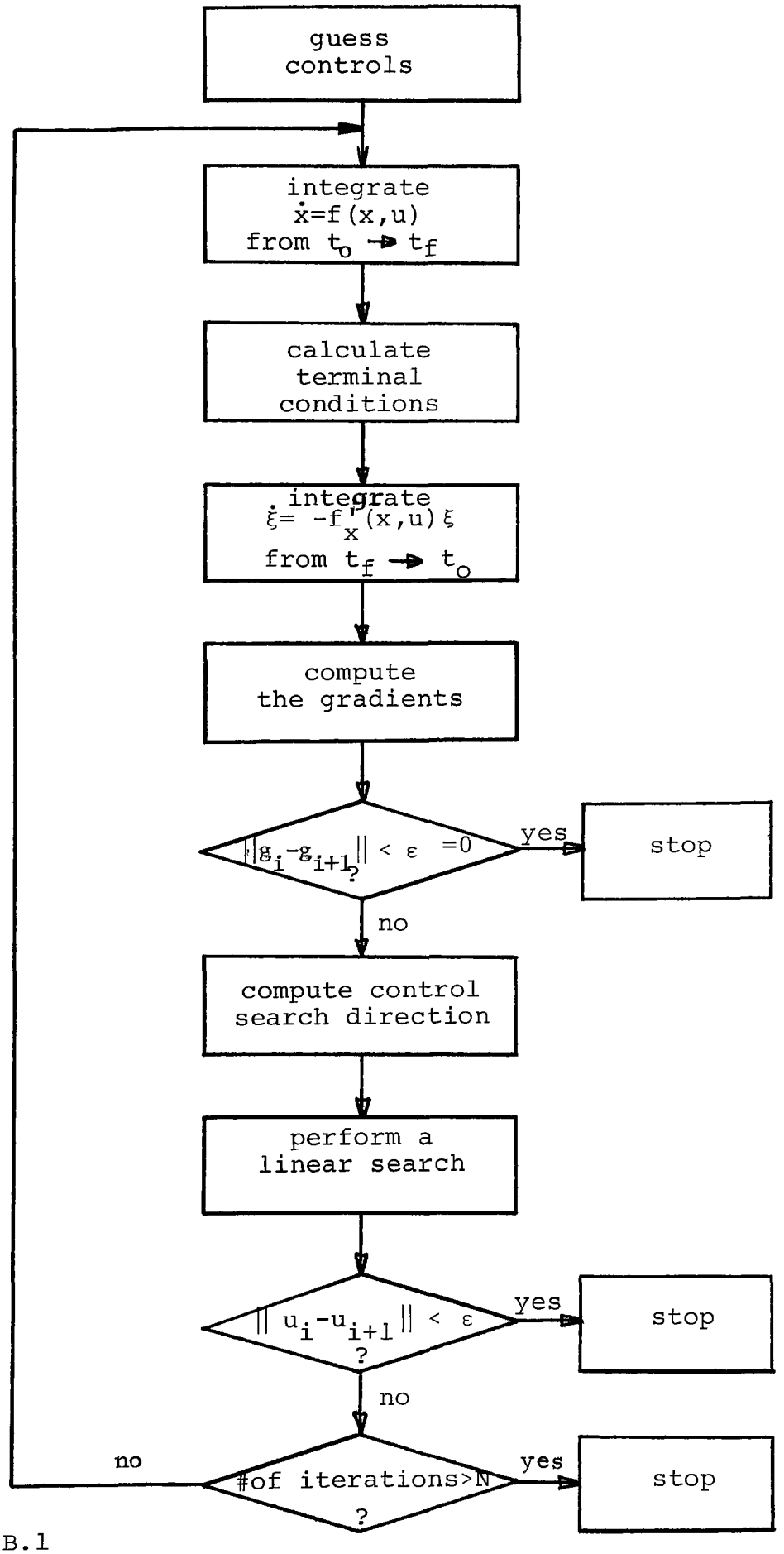


Fig. A.B.1

```
$JOB [redacted],GUIBORD,PAGES=40,TIME=2                                  4.
C*****************************                                          5.
C      PROBLEM 4.1.A              *                                      6.
C*************************************************************         7.
C      THIS PROGRAM COMPUTES OPTIMAL CONTROLS BY THE             *      8.
C      CONJUGATE GRADIENT DESCENT METHOD.                        *      9.
C       ******                                                   *     10.
C       LEGEND.                                                  *     11.
C       ******                                                   *     12.
C       PRMT(1)=INITIAL TIME.                                    *     13.
C       PRMT(2)=TERMINAL TIME.                                   *     14.
C       PRMT(3)=INTEGRATION STEP SIZE.                           *     15.
C       Y(1)-Y(7)=STATES.                                        *     16.
C       DERY(1)-DERY(7)=STATE OR CO-STATE DERIVATIVES.           *     17.
C       STORX=STORAGE ALLOCATION FOR STATE TRAJECTORIES.         *     18.
C       STORXC=STORAGE ALLOC. FOR THE CO-STATE TRAJECTORIES.     *     19.
C       STORU=STORAGE ALLOCATION FOR CONTROLS.                   *     20.
C       GRADU=STORAGE ALLOCATION FOR GRADIENTS.                  *     21.
C       JCOST=TERMINAL COST FUNCTION.                            *     22.
C       A0=FIRST ESTIMATE OF THE MINIMIZING ARGUMENT FOR         *     23.
C       THE LINEAR SEARCH.                                       *     24.
C       TS=TIME SCALING FACTOR.                                  *     25.
C       N=NUMBER OF STATES.                                      *     26.
C       NC=NUMBER OF CONTROLS.                                   *     27.
C       MAX=MAXIMUM NUMBER OF ITERATIONS ON CONTROLS.            *     28.
C       A1,A2,A3=SYSTEM CONSTANTS.                               *     29.
C       B1,B2=SYSTEM CONSTANTS.                                  *     30.
C       C1,C2,C3=SYSTEM CONSTANTS.                               *     31.
C       L1,L2=LAMBDA1 AND LAMBDA2 RESPECTIVELY.                  *     32.
C       R=TERMINAL TIME WEIGHTING COEFFICIENT.                   *     33.
C*************************************************************         34.
      1       REAL*8 Y(7),DERY(7),STORX(7,101),STORXC(7,101)             35.
      2       REAL*8 STORU(3,101),GRADU(3,101),PRMT(4)                   36.
      3       REAL*8 JCOST,DENOM,BETAI,ANS,EPSLN                         37.
      4       REAL*8 YY,XXX,P(3,101),FUI                                 38.
      5       REAL*8 A0                                                  39.
      6       REAL*8 TS,A1,A2,A3,B1,B2,C1,C2,C3,L1,L2,R                  40.
      7       COMMON TS,A1,A2,A3,B1,B2,C1,C2,C3,L1,L2,R                  41.
      8       COMMON /AA/ A0                                             42.
C*****SET THE COMMON VARIABLES.                                          43.
      9       TS=10.D0                                                   44.
     10       A0=1.D-8                                                   45.
     11       A1=1.3721D-4                                               46.
     12       A2=-.88217D0                                               47.
     13       A3=1.D0                                                    48.
     14       B1=.24D0                                                   49.
     15       B2=1.D0                                                    50.
     16       C1=-1.32290-4                                              51.
     17       C2=.81465D0                                                52.
     18       C3=1.D0                                                    53.
     19       L1=10.D0                                                   54.
     20       L2=1.D0                                                    55.
     21       R=4.D4                                                     56.
C*****INITIALIZE PROGRAM.                                                57.
     22       N=7                                                        58.
     23       NC=3                                                       59.
     24       PRMT(1)=0.D0                                               60.
     25       PRMT(2)=1.D0                                               61.
     26       PRMT(3)=.01D0                                              62.
     27       MAX=9
```

maximum number of iterations

```
      [redacted]=0
      M=PRMT(2)/PRMT(3)
      MM=M+1
      DO 5 I=1,NC
      DO 5 J=1,M
    5 STORU(I,J)=-.01D0
      YY=1.D0
C*****START ALGORITHM.
    7 CALL INCON(N,Y)
      CALL INTFOR(N,NC,MM,PRMT,Y,DERY,STORX,STORU)
      FUI=Y(7)+.5D0*R*(Y(1)**2+Y(2)**2+Y(3)**2+Y(4)**2+Y(5)**2
     &+Y(6)**2)
      CALL CUTP(N,PRMT(2),Y,DERY)
      CALL TERCON(N,Y)
      CALL INTBAK(N,PRMT,Y,DERY,STORX,STORXC)
      CALL CUTP(N,PRMT(1),Y,DERY)
      CALL GRAD(N,PRMT,STORXC,STORU,GRADU)
      XXX=0.D0
      DO 65 I=1,3
      DO 65 J=1,M
   65 XXX=XXX+GRADU(I,J)**2/FLCAT(M)
      BETAI=XXX/YY
      YY=XXX
      IF(III-1) 30,30 ,50
   30 DO 40 I=1,3
      DO 40 J=1,M
   40 P(I,J)=-GRADU(I,J)
      GOTO 60
   50 DO 55 I=1,3
      DO 55 J=1,M
   55 P(I,J)=-GRADU(I,J)+BETAI*P(I,J)
   60 CONTINUE
      CALL ALPHA(N,NC,MM,PRMT,P,STORU,FUI)
      III=III+1
      IF (III.GT.MAX) GOTO 90
      GOTO 7
   90 PRINT200
  100 CALL INCON(N,Y)
      CALL INTFR(N,PRMT,Y,DERY,STORU)
  200 FORMAT (5X,'THE VALUE OF III EXCEEDS MAX.')
      STOP
      END
C*****END OF MAIN PROGRAM.
C
C*************************************************************
C     THIS SUBROUTINE GIVES THE INITIAL                      *
C     CONDITIONS FOR THE STATE DIFFERENTIAL EQUATIONS.       *
C*************************************************************
C

      SUBROUTINE INCON(N,Y)
      REAL*8 Y(N)
      Y(1)=0.01D0
      Y(2)=0.03D0
      Y(3)=0.05D0
      Y(4)=0.01D0
      Y(5)=0.03D0
      Y(6)=0.05D0
      Y(7)=0.D0
      RETURN
```

guess controls

← control search direction

```
      END                                                                  122.
C                                                                          123.
C*************************************************************           124.
C     THIS SUBROUTINE DEFINES THE STATE DIFFERENTIAL        *              125.
C     EQUATIONS.                                            *              126.
C*************************************************************           127.
C                                                                          128.
      SUBROUTINE FCT1(N,Y,DERY,U)                                          129.
      REAL*8 Y(N),DERY(N),U(3)                                             130.
      REAL*8 TS,A1,A2,A3,B1,B2,C1,C2,C3,L1,L2,R                            131.
      COMMON TS,A1,A2,A3,B1,B2,C1,C2,C3,L1,L2,R                            132.
      DERY(1)=TS*(A1*Y(3)+A2*Y(2)*Y(3)+A3*U(1))                            133.
      DERY(2)=TS*(B1*Y(1)*Y(3)+B2*U(2))                                    134.
      DERY(3)=TS*(C1*Y(1)+C2*Y(1)*Y(2)+C3*U(3))                            135.
      DERY(4)=TS*Y(1)                                                      136.
      DERY(5)=TS*Y(2)                                                      137.
      DERY(6)=TS*Y(3)                                                      138.
      DERY(7)=TS*.5D0*L1*(Y(1)**2+Y(2)**2+Y(3)**2+Y(4)**2+                 139.
     $Y(5)**2+Y(6)**2)+TS*.5D0*L2*(U(1)**2+U(2)**2+U(3)**2)                140.
      RETURN                                                               141.
      END                                                                  142.
C                                                                          143.
C*************************************************************           144.
C     THIS SUBROUTINE DEFINES THE CO-STATE DIFFERENTIAL     *              145.
C     EQUATIONS.                                            *              146.
C*************************************************************           147.
C                                                                          148.
      SUBROUTINE FCT2(N,Y,DERY,STORX,JJ)                                   149.
      REAL*8 Y(N),DERY(N),STORX(N,101),X(10)                               150.
      REAL*8 TS,A1,A2,A3,B1,B2,C1,C2,C3,L1,L2,R                            151.
      COMMON TS,A1,A2,A3,B1,B2,C1,C2,C3,L1,L2,R                            152.
      DO 1 I=1,N                                                           153.
    1 X(I)=STORX(I,JJ)                                                     154.
      DERY(1)=-TS*(B1*X(3)*Y(2)+C1*Y(3)+C2*X(2)*Y(3)+Y(4)+L1*X(1)*Y(7))    155.
      DERY(2)=-TS*(A2*X(3)*Y(1)+C2*X(1)*Y(3)+Y(5)+L1*X(2)*Y(7))            156.
      DERY(3)=-TS*(A1*Y(1)+A2*X(2)*Y(1)+B1*X(1)*Y(2)+Y(6)+L1*X(3)*Y(7))    157.
      DERY(4)=-TS*L1*X(4)*Y(7)                                             158.
      DERY(5)=-TS*L1*X(5)*Y(7)                                             159.
      DERY(6)=-TS*L1*X(6)*Y(7)                                             160.
      DERY(7)=0.D0                                                         161.
      RETURN                                                               162.
      END                                                                  163.
C                                                                          164.
C*************************************************************           165.
C     THIS SUBROUTINE DEFINES THE GRADIENT OF THE           *              166.
C     COST FUNCTION.                                        *              167.
C*************************************************************           168.
C                                                                          169.
      SUBROUTINE GRAD(N,PRMT,STORXC,STORU,GRADU)                           170.
      REAL*8 STORXC(N,101),STORU(3,101),GRADU(3,101),PRMT(4)               171.
      REAL*8 TS,A1,A2,A3,B1,B2,C1,C2,C3,L1,L2,R                            172.
      COMMON TS,A1,A2,A3,B1,B2,C1,C2,C3,L1,L2,R                            173.
      M=PRMT(2)/PRMT(3)                                                    174.
      DO 5 I=1,3                                                           175.
      DO 5 J=1,M                                                           176.
    5 GRADU(I,J)=TS*(STORXC(I,J)+L2*STORU(I,J))                            177.
      RETURN                                                               178.
```

Define $\dot{x} = f(x,u)$

Define $\dot{\lambda} = -f_x'(x,u)$

To compute gradients

```
      END                                                                179.
C                                                                        180.
C*************************************************************         181.
C     THIS SUBROUTINE GIVES THE TERMINAL CONDITIONS FOR      *         182.
C     THE ADJOINT DIFFERENTIAL EQUATIONS.                    *         183.
C*************************************************************         184.
C                                                                        185.
      SUBROUTINE TERCON(N,Y)                                             186.
      REAL*8 Y(N)                                                        187.
      REAL*8 TS,A1,A2,A3,B1,B2,C1,C2,C3,L1,L2,R                          188.
      COMMON TS,A1,A2,A3,B1,B2,C1,C2,C3,L1,L2,R                          189.
      DO 1 I=1,6                                                         190.
    1 Y(I)=R*Y(I)                                                        191.
      Y(7)=1.D0                                                          192.
      RETURN                                                             193.
      END                                                                194.
C                                                                        195.
C*************************************************************         196.
C     THIS SUBROUTINE INTEGRATES THE STATE DIFFERENTIAL      *         197.
C     EQUATIONS AND STORES THE TRAJECTORIES.                 *         198.
C*************************************************************         199.
C                                                                        200.
      SUBROUTINE INTFOR(N,NC,M,PRMT,Y,DERY,STORX,STORU)                  201.
      REAL*8 PRMT(4),Y(N),DERY(N),STORX(N,M)                             202.
      REAL*8 STORU(NC,M),U(10)                                           203.
      REAL*8 A(10),B(10),C(10),D(10),Z(10),XX(10),T,H                    204.
      NSTEPS=PRMT(2)/PRMT(3)                                             205.
      DO 2 I=1,N                                                         206.
    2 STORX(I,1)=Y(I)                                                    207.
      DO 13 J=1,NSTEPS                                                   208.
      DO 1 I=1,3                                                         209.
    1 U(I)=STORU(I,J)                                                    210.
      CALL FCT1(N,Y,DERY,U)                                              211.
      DO 6 I=1,N                                                         212.
      A(I)=PRMT(3)*DERY(I)                                               213.
    6 Z(I)=Y(I)+A(I)/2.D0                                                214.
      CALL FCT1(N,Z,DERY,U)                                              215.
      DO 8 I=1,N                                                         216.
      B(I)=PRMT(3)*DERY(I)                                               217.
    8 Z(I)=Y(I)+B(I)/2.D0                                                218.
      CALL FCT1(N,Z,DERY,U)                                              219.
      DO 10 I=1,N                                                        220.
      C(I)=PRMT(3)*DERY(I)                                               221.
   10 Z(I)=Y(I)+C(I)                                                     222.
      CALL FCT1(N,Z,DERY,U)                                              223.
      DO 12 I=1,N                                                        224.
      D(I)=PRMT(3)*DERY(I)                                               225.
   12 Y(I)=Y(I)+(A(I)+2.D0*B(I)+2.D0*C(I)+D(I))/6.D0                     226.
      JJ=J+1                                                             227.
      DO 13 I=1,N                                                        228.
   13 STORX(I,JJ)=Y(I)                                                   229.
      RETURN                                                             230.
      END                                                                231.
C                                                                        232.
C*************************************************************         233.
C     THIS SUBROUTINE INTEGRATES THE ADJOINT DIFFERENTIAL    *         234.
C     EQUATIONS BACKWARDS IN TIME AND STORES THE RESULTING   *         235.
C      TRAJECTORIES.                                         *         236.
```

To integrate $\dot{x} = f(x,u)$ from $t_0 \rightarrow t_f$

```
      C*************************************************************
      C
      SUBROUTINE INTBAK(N,PRMT,XC,XCDOT,STORX,STORXC)
      REAL*8 PRMT(4),XC(N),XCDOT(N),STORX(N,101),STORXC(N,101)
      REAL*8 A(10),B(10),C(10),D(10),Z(10)
      M=PRMT(2)/PRMT(3)
      JJ=M+1
      DO 12 J=1,M
      CALL FCT2(N,XC,XCDOT,STORX,JJ)
      DO 1 I=1,N
    1 STORXC(I,JJ)=XC(I)
      JJ=M+1-J
      DO 6 I=1,N
      A(I)=-PRMT(3)*XCDOT(I)
    6 Z(I)=XC(I)+A(I)/2.D0
      CALL FCT2(N,Z,XCDOT,STORX,JJ)
      DO 8 I=1,N
      B(I)=-PRMT(3)*XCDOT(I)
    8 Z(I)=XC(I)+B(I)/2.D0
      CALL FCT2(N,Z,XCDOT,STORX,JJ)
      DO 10 I=1,N
      C(I)=-PRMT(3)*XCDOT(I)
   10 Z(I)=XC(I)+C(I)/2.D0
      CALL FCT2(N,Z,XCDOT,STORX,JJ)
      DO 12 I=1,N
      D(I)=-PRMT(3)*XCDOT(I)
   12 XC(I)=XC(I)+(A(I)+2.D0*B(I)+2.D0*C(I)+D(I))/6.D0
      DO 13 I=1,N
   13 STORXC(I,JJ)=XC(I)
      RETURN
      END
C
C*************************************************************
C     THIS SUBROUTINE INTEGRATES THE STATE DIFFERENTIAL    *
C     EQUATIONS AND PRINTS THE TRAJECTORIES.               *
C*************************************************************
C
      SUBROUTINE INTFR(N,PRMT,Y,DERY,STORU)
      REAL*8 PRMT(4),Y(N),DERY(N),STORU(3,101),U(3)
      REAL*8 A(10),B(10),C(10),D(10),Z(10),XX(10),T,H
      M=PRMT(2)/PRMT(3)
      T=0.D0
      DO 13 J=1,M
      DO 1 I=1,3
    1 U(I)=STORU(I,J)
      WRITE(6,100) T,(Y(I),I=1,7),(U(I),I=1,3)
      CALL FCT1(N,Y,DERY,U)
      DO 6 I=1,N
      A(I)=PRMT(3)*DERY(I)
    6 Z(I)=Y(I)+A(I)/2.D0
      CALL FCT1(N,Z,DERY,U)
      DO 8 I=1,N
      B(I)=PRMT(3)*DERY(I)
    8 Z(I)=Y(I)+B(I)/2.D0
      CALL FCT1(N,Z,DERY,U)
      DO 10 I=1,N
      C(I)=PRMT(3)*DERY(I)
   10 Z(I)=Y(I)+C(I)
```

To integrate $\dot{\xi} = -f_x'(x,u)\xi$ from $t_f \to t_0$

```
      CALL FCT1(N,Z,DERY,U)                                              295.
      DO 12 I=1,N                                                        296.
      D(I)=PRMT(3)*DERY(I)                                               297.
   12 Y(I)=Y(I)+(A(I)+2.D0*B(I)+2.D0*C(I)+D(I))/6.D0                     298.
      T=T+PRMT(3)                                                        299.
   13 CONTINUE                                                           300.
      WRITE(6,100) T,(Y(I),I=1,7)                                        301.
      RETURN                                                             302.
  100 FORMAT(5X,F5.3,10F10.5)                                            303.
      END                                                                304.
C                                                                        305.
C*************************************************************         306.
C     THIS SUBROUTINE IS PART OF THE LINEAR SEARCH          *            307.
C     REQUIRED TO UPDATE THE CONTROLS.                      *            308.
C*************************************************************         309.
C                                                                        310.

      SUBROUTINE ALPHA(N,NC,M,PRMT,P,STORU,FUI)                          311.
      REAL*8 STORU(NC,M),STORX(7,101),ALPH,FUI,FAKM1,X(4),Z(4),ZMIN      312.
      REAL*8 PRMT(4),P(NC,M),DABS,A0,U1(3,101),Y(7),DERY(7)              313.
      REAL*8 TS,A1,A2,A3,B1,B2,C1,C2,C3,L1,L2,R                          314.
      REAL*8 AO                                                          315.
      COMMON TS,A1,A2,A3,B1,B2,C1,C2,C3,L1,L2,R                          316.
      COMMON /AA/ AO                                                     317.
      DO 6 I=1,3                                                         318.
      X(I)=0.D0                                                          319.
    6 Z(I)=0.D0                                                          320.
      X(4)=FUI                                                           321.
      Z(4)=0.D0                                                          322.
      MAX=20                                                             323.
      III=0                                                              324.
      ALPH=AO                                                            325.
      M1=M-1                                                             326.
      DO 20 I=1,3                                                        327.
      DO 20 J=1,M1                                                       328.
   20 U1(I,J)=STORU(I,J)+ALPH*P(I,J)                                     329.
      CALL INCON(N,Y)                                                    330.
      CALL INTFOR(N,NC,M,PRMT,Y,DERY,STORX,U1)                           331.
      CALL OUTP(N,PRMT(2),Y,DERY)                                        332.
      Z(3)=Z(4)                                                          333.
      Z(4)=ALPH                                                          334.
      X(3)=X(4)                                                          335.
      X(4)=Y(7)+.5D0*R*(Y(1)**2+Y(2)**2+Y(3)**2+                         336.
     $Y(4)**2+Y(5)**2+Y(6)**2)                                           337.
  101 FAKM1=X(4)                                                         338.
      III=III+1                                                          339.
      ALPH=2.D0*ALPH                                                     340.
      DO 102 I=1,3                                                       341.
      DO 102 J=1,M1                                                      342.
  102 U1(I,J)=STORU(I,J)+ALPH*P(I,J)                                     343.
      CALL INCON(N,Y)                                                    344.
      CALL INTFOR(N,NC,M,PRMT,Y,DERY,STORX,U1)                           345.
      CALL OUTP(N,PRMT(2),Y,DERY)                                        346.
      Z(1)=Z(2)                                                          347.
      Z(2)=Z(3)                                                          348.
      Z(3)=Z(4)                                                          349.
      Z(4)=ALPH                                                          350.
      X(1)=X(2)                                                          351.
      X(2)=X(3)                                                          352.
      X(3)=X(4)                                                          353.
```

[Perform a Linear search]

```
259           X(4)=Y(7)+.5D0*R*(Y(1)**2+Y(2)**2+Y(3)**2+                    354.
             $Y(4)**2+Y(5)**2+Y(6)**2)                                      355.
260           IF(X(4).GT.FAKM1) GOTO 150                                    356.
261           IF(III.GT.MAX) GOTO 350                                       357.
262           GOTO 101                                                      358.
263       150 ALPH=(ALPH+ALPH/2.D0)/2.D0                                    359.
264       255 DO 251 I=1,3                                                  360.
265           DO 251 J=1,M1                                                 361.
266       251 U1(I,J)=STORU(I,J)+ALPH*P(I,J)                                362.
267           CALL INCON(N,Y)                                               363.
268           CALL INTFOR(N,NC,M,PRMT,Y,DERY,STORX,U1)                      364.
269           CALL OUTP(N,PRMT(2),Y,DERY)                                   365.
270           X(1)=X(2)                                                     366.
271           X(2)=X(3)                                                     367.
272           X(3)=Y(7)+.5D0*R*(Y(1)**2+Y(2)**2+Y(3)**2+                    368.
             $Y(4)**2+Y(5)**2+Y(6)**2)                                      369.
273           Z(1)=Z(2)                                                     370.
274           Z(2)=Z(3)                                                     371.
275           Z(3)=(Z(3)+Z(4))/2.D0                                         372.
276           WRITE(6,500) (X(I),I=1,4)                                     373.
277           WRITE(6,500) (Z(I),I=1,4)                                     374.
278           CALL POLY(X,Z,ZMIN)                                           375.
279           WRITE(6,501) ZMIN                                             376.
280           DO 325 I=1,3                                                  377.
281           DO 325 J=1,M1                                                 378.
282       325 STORU(I,J)=STORU(I,J)+ZMIN*P(I,J)                             379.
283           RETURN                                                        380.
284       350 WRITE(6,400)                                                  381.
285       400 FORMAT(5X,'I IS TOO LARGE.',///)                              382.
286       500 FORMAT(5X,4F12.8,/)                                           383.
287       501 FORMAT(5X,'ZMIN=',F12.8,/)                                    384.
288           STOP                                                          385.
289           END                                                           386.
          C                                                                 387.
          C*****************************************************************388.
          C     THIS SUBROUTINE PERFORMS A THIRD DEGREE POLYNOMIAL  *       389.
          C     INTERPOLATION AND FINDS THE VALUE OF THE            *       390.
          C     MINIMIZING ARGUMENT.                                *       391.
          C*****************************************************************392.
          C                                                                 393.

290           SUBROUTINE POLY(X,Z,ZMIN)                                     394.
291           REAL*8 X(4),X1,X2,X3,X4,STEP                                  395.
292           REAL*8 Z(4),Z1,Z2,Z3,Z4,ALPH,Y                                396.
293           REAL*8 L1,L2,L3,L4,P,ZMIN                                     397.
294           L1(ALPH)=((ALPH-Z2)*(ALPH-Z3)*(ALPH-Z4))/                     398.
             #((Z1-Z2)*(Z1-Z3)*(Z1-Z4))                                     399.
295           L2(ALPH)=((ALPH-Z1)*(ALPH-Z3)*(ALPH-Z4))/                     400.
             #((Z2-Z1)*(Z2-Z3)*(Z2-Z4))                                     401.
296           L3(ALPH)=((ALPH-Z1)*(ALPH-Z2)*(ALPH-Z4))/                     402.
             #((Z3-Z1)*(Z3-Z2)*(Z3-Z4))                                     403.
297           L4(ALPH)=((ALPH-Z1)*(ALPH-Z2)*(ALPH-Z3))/                     404.
             #((Z4-Z1)*(Z4-Z2)*(Z4-Z3))                                     405.
298           P(Y)=X1*L1(Y)+X2*L2(Y)+X3*L3(Y)+X4*L4(Y)                      406.
299        20 Z1=Z(1)                                                       407.
300           Z2=Z(2)                                                       408.
301           Z3=Z(3)                                                       409.
302           Z4=Z(4)                                                       410.
303           X1=X(1)                                                       411.
304           X2=X(2)                                                       412.
```

```
305           X3=X(3)                                                          413.
306           X4=X(4)                                                          414.
307        30 CONTINUE                                                         415.
308           STEP=(Z4-Z1)/200.D0                                              416.
309           ZMIN=Z1                                                          417.
310           DO 40 I=1,200                                                    418.
311           Y=Z1+FLOAT(I)*STEP                                               419.
312           IF(P(Y).LT.P(ZMIN)) ZMIN=Y                                       420.
313        40 CONTINUE                                                         421.
314           RETURN                                                           422.
315           END                                                              423.
         C                                                                     424.
         C*****************************************************************  425.
         C       THIS SUBROUTINE PRINTS RESULTS.                           *  426.
         C*****************************************************************  427.
         C                                                                     428.

316           SUBROUTINE OUTP(N,X,Y,DERY)                                      429.
317           REAL*8 X,Y(N),DERY(N),JCOST                                      430.
318           REAL*8 TS,A1,A2,A3,B1,B2,C1,C2,C3,L1,L2,R                        431.
319           COMMON TS,A1,A2,A3,B1,B2,C1,C2,C3,L1,L2,R                        432.
320           JCOST=Y(7)+.5D0*R*(Y(1)**2+Y(2)**2+Y(3)**2+Y(4)**2+Y(5)**2+      433.
             &Y(6)**2)                                                        434.
321           WRITE(6,1) X,(Y(I),I=1,6),JCOST                                  435.
322        1  FORMAT(5X,8F15.4)                                                436.
323           RETURN                                                           437.
324           END                                                              438.

         $ENTRY                                                               439.
```

Peferences

[1] D.C.McLellan, 'Anticipated Developments In Communications Satellite Technology.', Communications Satellite Systems, M.I.T. Press 1974, pp145-171.

[2] R.H.Canon Jr., 'Some Basic Resporse Relations For Reaction-Wheel Attitude Control', ARS Journal 1962.

[3] A.F.Guibord, N.U.Ahmed, 'A Note On Satellite Dynamics', IEEE Transactions On Aerospace And Electronic Systems, Nov 1976, pp 819-822.

[4] A.L.Greensite, 'Analysis And Design Of Space Vehicle Flight Control Systems.', New-York: Spartan Book Company, 1970.

[5] H.Goldstein, 'Classical Mechanics.' Addison-Wesley Publishing Company Inc. 1950.

[6] M.R.Spiegel, 'Theoretical Mechanics With An Introduction To Lagrange's Equations And Hamiltonian Theory'. McGraw-Hill Book Company, 1967.

[7] W.V.Discenza, J.J.Griffin, and S.Millman, 'Simulation And Test Results Of A Three-Axis Attitude Control System For Synchronous Orbit Communication Satellites.'

IAF-76-235.

[8] F.H.Cannon Jr., 'Some Basic Response Relations For Reaction-Wheel Attitude Control.', AIAS Journal 1962 pp. 145-171.

[9] L.S.Pontryagin, V.G.Boltyanski, R.V.Gamkrelidge, and E.F.Mishchenko. 'The Mathematical Theory Of Optimal Processes.', Interscience Publishers, 1962.

[10] H.B.Keller, 'Numerical Methods For Two-Point Bourdary Value Problems', Blaisdell, Waltham, Mass., 1968.

[11] R.Bellman and R.Kalaba, 'Quasilinearization and Nonlinear Boundary Value Problems', American Elsevier, New-York, 1965.

[12] S.M.Roberts and J.S.Shipman, 'Two-Point Bourdary Value Problems: Shooting Methods', American Elsevier, New-York, 1972.

[13] E.Polak, 'Computational Methods In Optimization', Academic Press, 1971.

[14] L.G.Birta, 'The TEF/Davidon-Fletcher- Powell Method In The Computation Of Optimal Controls', Joint Automatic Control Conference, 1969.

[15] L.G.Birta, 'A Parameter Optimization Module For CSSL Based Simulation Software', Simulation, Vol. 28, No.4, April 1977, pp. 113-121.

[16] R.Fletcher and M.J.D.Powell, 'A Rapidly Corvergent Descent Method For Minimization', Computer Journal, July 1963.

[17] W.W.Claycombe, W.G.Sullivan, 'Fourdations Of Mathematical Programming', Reston Publishing Company Inc., 1975.

[18] A.Chatterjee, 'Imbedding Methods In The Numerical Solution Of Optimization Problems', M.A.Sc. Thesis, University of Ottawa,1974.

[19] L.A.Zadeh and C.A.Desoer, Linear System Theory —The State Space Approach', McGraw-Hill New-York, 1963.

[20] R.E.Bellman, 'Dynamic Programming', Princeton University Press, Princeton, N.J., 1959.

[21] L.Hasdorff, 'Gradient Optimization And Nonlinear Control', Wiley-Interscience, 1976.

[22] L.Hasdorff, 'The Gradient Of The Cost Functional For Several Common Types Of Control Inputs', Proc. SWIEECO,

Dallas, April 1970.

[23] D.R.Smith, 'Variational Methods In Optimization', Prentice-Hall, Inc. Englewood Cliffs, N.J., 1974.

[24] R.B.Ash, 'Measure Integration And Functional Analysis', Academic Press, 1972.

[25] A.P.Sage, 'Optimum Systems Control', Prentice-Hall, 1977.

[26] A.K.Sen, N.U.Ahmed, 'Fuel Savings In The Whecon Attitude Control System Using An Asymetric Controller',IEEE Transactions On Aerospace And Electronic Systems,May 1976, pp 349-354.

[27] T.A.Heppenheimer, 'Colonies In Space', Stackpole Publishing Company, 1977.

VITA

NAME: Arthur François Guibord

BORN: 23-10-1952, Ottawa, Canada

EDUCATED:

Primary: Cumberland Public School

Secondary: Eastview High School

College: Algonquin College of Applied Arts and Technology, Ottawa

Dipl. T.(Electronics), 1973.

University: University of Ottawa, Ottawa
B.A.Sc. in Electrical Engineering, 1976.