DEFENCE
SCIENCE & TECHNOLOGY

Mathematical Models for a Missile Autopilot Design

Farhan A. Faruqi and
Thanh Lan Vu

DSTO-TN-0449

44

Mathematical Models for a Missile Autopilot Design

Farhan A. Faruqi and Thanh Lan Vu

Weapons Systems Division
Systems Sciences Laboratory

DSTO-TN-0449

ABSTRACT

This report considers the derivation of the mathematical model for a missile autopilot in state space form. The basic equations defining the airframe dynamics are non-linear, however, since the non-linearities are "structured" (in the sense that the states are of quadratic form) a novel approach of expressing this non-linear dynamics in state space form is given. This should provide a useful way to implement the equations in a computer simulation program and possibly for future application of non-linear analysis and synthesis techniques, particularly for autopilot design of missiles executing high g-manoeuvres.

This report also considers a locally linearised state space model that lends itself to better known linear techniques of the modern control theory. A coupled multi-input multi-output (MIMO) model is derived suitable for both the application of the modern control techniques as well as the classical time-domain and frequency domain techniques. This is validated by comparing the model with the other published results, and through both open and closed-loop systems simulations. The models developed are useful for further research on precision optimum guidance and control. It is hoped that the model will provide more accurate presentations of missile autopilot dynamics and will be used for adaptive and integrated guidance & control of agile missiles.

RELEASE LIMITATION

Approved for public release

Published by

DSTO Systems Sciences Laboratory
PO Box 1500
Edinburgh South Australia 5111 Australia

Telephone: (08) 8259 5555
Fax: (08) 8259 6567

AR-012-430
August 2002

APPROVED FOR PUBLIC RELEASE

Mathematical Models for a Missile Autopilot Design

Executive Summary

Requirements for next generation guided weapons, particularly with respect to their capability to engage high speed, highly agile targets and achieve precision end-game trajectory, has prompted a revision of the way in which the guidance and autopilot design is undertaken. This report considers the derivation of the mathematical models for a missile autopilot in state space form. The basic equations defining the airframe dynamics are non-linear, however, since the non-linearities are "structured" (in the sense that the states are of quadratic form) a novel approach of expressing this non-linear dynamics in state space form is given. This should provide a useful way to implement the equations in a computer simulation program and possibly for future application of non-linear analysis and synthesis techniques.

This report also considers a locally linearised state space model that lends itself to better known linear techniques of the modern control theory. A coupled multi-input multi-output (MIMO) model is derived suitable for both the application of the modern control techniques as well as the classical time-domain and frequency domain techniques. This is validated by comparing the model with the other published results, and through both open and closed-loop systems simulations. The models developed are useful for further research on precision optimum guidance and control. It is hoped that the model will provide more accurate presentations of missile auto-pilot dynamics and will be used for adaptive and integrated guidance & control of agile missiles.

Authors

Dr. Farhan A. Faruqi
Weapons Systems Division

Farhan A. Faruqi received B.Sc.(Hons) in Mechanical Engineering from the University of Surrey (UK), 1968; M.Sc. in Automatic Control from the University of Manchester Institute of Science and Technology (UK), 1970 and Ph.D from the Imperial College, London University (UK), 1973. He has over 20 years experience in the Aerospace and Defence Industry in UK, Europe and the USA. Prior to joining DSTO in January 1999 he was an Associate Professor at QUT (Australia) 1993-98. His research interests include: Missile Navigation, Guidance and Control, Target Tracking and Precision Pointing Systems, Strategic Defence Systems, Signal Processing, and Optoelectronics.

Thanh Lan Vu
Weapons Systems Division

Thanh Lan Vu joined the Guidance & Control group in the Weapons Systems Dvision in August 1999. She received a Master degree in Computing & Control Engineering from the Institute of Norwegian Technology, Trondheim, Norway, in 1991. She also received a PhD in Mechanical Engineering from the University of Western Australia in 1998. Her thesis presents on-line system identification method and active control of non-linear vibration. Her research interests include: Guidance and Control, Signal Processing, Analog and Digital Filters, Active Control of Vibration, and System Identification.

Contents

1. Introduction

Requirements for next generation guided weapons, particularly with respect to their capability to engage high speed, highly agile targets and achieve precision end-game trajectory, has prompted a revision of the way in which the guidance and autopilot design is undertaken. Integrating the guidance and control function is a synthesis approach that is being pursued as it allows the optimisation of the overall system performance. This approach requires a more complete representation of the airframe dynamics and the guidance system. The use of state space model allows the application of modern control techniques such as the optimal control and parameter estimation techniques to be utilised. In this report we derive the autopilot model that will serve as a basis for an adaptive autopilot design and allow further extension of this to integrated guidance and control system design.

Over the years a number of authors [1-3, 6] have considered modelling, analysis and design of autopilots for atmospheric flight vehicles including guided missiles. In the majority of the published work on autopilot analysis and design, locally linearised versions of the model with decoupled airframe dynamics has been considered. This latter simplification arises out of the assumption that the airframe and its mass distribution are symmetrical about the body axes, and that the yaw, pitch and roll motion about the equilibrium state remain "small". As a result, most of the autopilot analysis and design techniques, considered in open literature, use classical control approach, such as: single input/single output transfer-functions characterisation of the system dynamics and Bode, Nyquist, root-locus and transient response analysis and synthesis techniques [5,7]. These techniques are valid for a limited set of flight regimes and their extension to cover a wider set of flight regimes and airframe configurations requires autopilot gain and compensation network switching.

With the advent of fast processors it is now possible to take a more integrated approach to autopilot design. Modern optimal control techniques allow the designer to consider autopilots with high-order dynamics (large number of states) with multiple inputs/outputs and to synthesise controllers such that the error between the demanded and the achieved output is minimised. Moreover, with real-time mechanisation any changes in the airframe aerodynamics can be identified (parameter estimation) and compensated for by adaptively varying the optimum control gain matrix. This approach should lead to missile systems that are able to execute high g-manoeuvres (required by modern guided weapons), adaptively adjust control parameters (to cater for widely varying flight profiles) as well as account for non-symmetric airframe and mass distributions.

Typically, for a missile autopilot, the input is the demanded control surface deflection and outputs are the achieved airframe (lateral) accelerations and body rates measured about the body axes. Other input/output variables (such as: the flight path angle and angle rate or the body angles) can be derived directly from lateral accelerations and body rates.

This report considers the derivation of the mathematical model for a missile autopilot in state space form. The basic equations defining the airframe dynamics are non-linear, however, since the non-linearities are "structured" (in the sense that the states are of quadratic form) a novel approach of expressing this non-linear dynamics in state space form is given. This should provide a useful way to implement the equations in a computer simulation program and possibly for future application of non-linear analysis and synthesis techniques. Detailed consideration of the quadratic/bilinear type of dynamic systems is given in [4].

This report also considers a locally linearised state space model that lends itself to better known linear techniques of the modern control theory. A coupled multi-input multi-output (MIMO) model is derived suitable for both the application of the modern control techniques as well as the classical time-domain and frequency domain techniques. This is validated by comparing the model with others previously published and through simulation of a decoupled single-input single-output (SISO) system.

2. Rigid Body Dynamics

2.1 Notation and Convention

Conventions and notations for vehicle body axes systems as well as the forces, moments and other quantities are shown in Figure 2.1 and defined in Table 2.1.

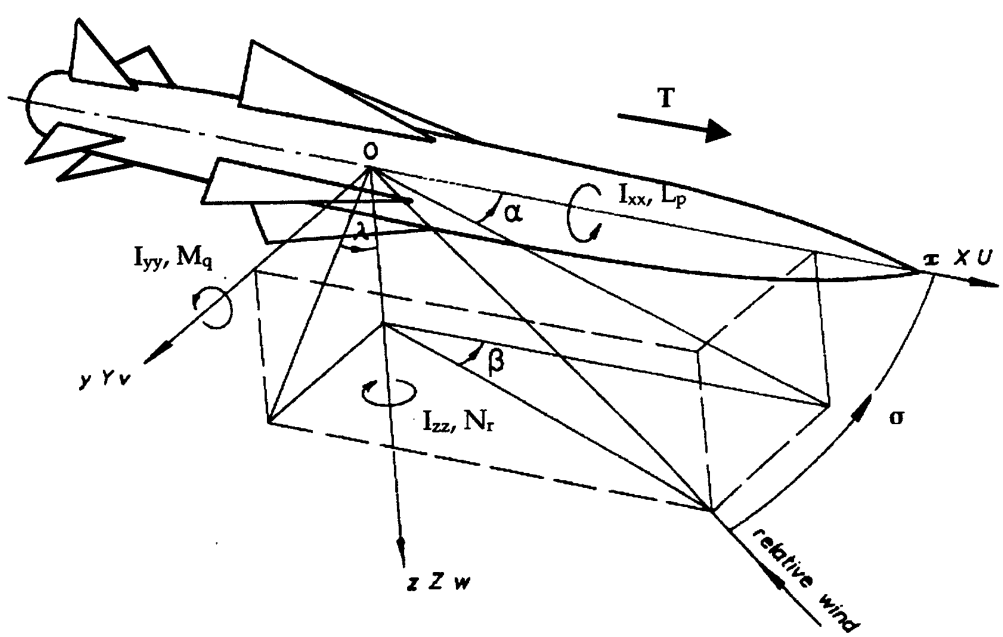

Figure 2.1 Motion variable notations

The variables shown in Figure 2.1 are defined as:
m - mass of a vehicle.
α - incidence in the pitch plane.
β - incidence in the yaw plane.
λ - incidence plane angle.
σ - total incidence, such that: $\tan \alpha = \tan \sigma \cos \lambda$, and $\tan \beta = \tan \sigma \sin \lambda$.
T – thrust.

Table 2.1: Motion variables

Vehicle Body Axes System	Roll axis X	Pitch axis Y	Yaw axis Z
Angular rates	p	q	r
Component of vehicle velocity along each axis	u	v	w
Component of aerodynamic forces acting on vehicle along each axis	X	Y	Z
Moments acting on vehicle about each axis	L	M	N
Moments of inertia about each axis	I_{xx}	I_{yy}	I_{zz}
Products of each inertia	I_{yz}	I_{zx}	I_{xy}
Longitudinal and lateral accelerations	a_x	a_y	a_z
Euler angles	ϕ	θ	ψ
Gravity along each axis	g_x	g_y	g_z
Vehicle thrust along the body axis	T		

ξ - aileron deflection.
η - elevator deflection.
ς - rudder deflection.

Figure 2.2 defines the control surface convention. Here the control surfaces are numbered as shown and the deflections $(\delta_1, \delta_2, \delta_3, \delta_4)$ are taken to be positive if clockwise, looking outwards along the individual hinge axis. Thus:

Aileron deflection: $\xi = \frac{1}{4}(\delta_1 + \delta_2 + \delta_3 + \delta_4)$, if all four control surfaces are active; or $\xi = \frac{1}{2}(\delta_1 + \delta_3)$, or $\xi = \frac{1}{2}(\delta_2 + \delta_4)$ if only two surfaces are active. Positive control defection (ξ) causes negative roll.

Elevator deflection: $\eta = \frac{1}{2}(\delta_1 - \delta_3)$. Positive control deflection (η) causes negative pitch.

Rudder deflection: $\zeta = \frac{1}{2}(\delta_2 - \delta_4)$. Positive control deflection (ζ) causes negative yaw.

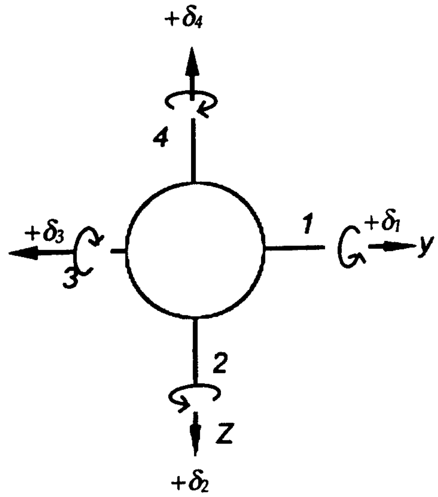

Figure 2.2 Control surfaces seen from the rear of a missile

2.2 Euler's Equations of Motion

The six equations of motion for a body with six degrees of freedom may be written as [1-3]:

$$m(\dot{u} + wq - vr) = X + T + g_x m \tag{2. 1}$$
$$m(\dot{v} + ur - wp) = Y + g_y m \tag{2. 2}$$
$$m(\dot{w} - uq + vp) = Z + g_x m \tag{2. 3}$$

$$I_{xx}\dot{p} - (I_{yy} - I_{zz})qr + I_{yz}(r^2 - q^2) - I_{zx}(pq + \dot{r}) + I_{xy}(rp - \dot{q}) = L \tag{2.4}$$
$$I_{yy}\dot{q} - (I_{zz} - I_{xx})rp + I_{zx}(p^2 - r^2) - I_{xy}(qr + \dot{p}) + I_{yz}(pq - \dot{r}) = M \tag{2.5}$$
$$I_{zz}\dot{r} - (I_{xx} - I_{yy})pq + I_{xy}(q^2 - p^2) - I_{yz}(rp + \dot{q}) + I_{zx}(qr - \dot{p}) = N. \tag{2.6}$$

Here $(\cdot) = \frac{d}{dt}$ - is the derivative operator.

Equations (2.1 to 2.3) represent the force equations of a generalised rigid body and describe the translational motion of its centre of gravity (c.g) since the origin of the vehicle body axes is assumed to be co-located with the body c.g.

Equations (2.3 to 2.6) represent the moment equations of a generalised rigid body and describe the rotational motion about the body axes through its c.g.

Separating the derivative terms and after some algebraic manipulation, Equations (2.1 to 2.3) may be written in a vector form as:

$$\frac{d}{dt}\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & -1 \\ 0 & -1 & 0 & 0 & 1 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} uq \\ ur \\ vp \\ vr \\ wp \\ wq \end{bmatrix} + \begin{bmatrix} \tilde{X} + \tilde{T} \\ \tilde{Y} \\ \tilde{Z} \end{bmatrix} + \begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix} \tag{2.7}$$

where: $\tilde{X} = \frac{X}{m}; \quad \tilde{Y} = \frac{Y}{m}; \quad \tilde{Z} = \frac{Z}{m}; \quad \tilde{T} = \frac{T}{m}.$

Note that the states (u,v,w,p,q,r) appear as "quadratic terms (form)".

Equations (2.4 to 2.6) can be written as:

$$[\boldsymbol{A}]\frac{d}{dt}\begin{bmatrix} p \\ q \\ r \end{bmatrix} = [\boldsymbol{B}] \begin{bmatrix} p^2 \\ pq \\ pr \\ q^2 \\ qr \\ r^2 \end{bmatrix} + \begin{bmatrix} L \\ M \\ N \end{bmatrix} \tag{2.8}$$

where: matrices $[\boldsymbol{A}]$ and $[\boldsymbol{B}]$ are given by:

$$[A]=\begin{bmatrix} I_{xx} & -I_{xy} & -I_{zx} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{yz} & I_{zz} \end{bmatrix}$$

$$[B]=\begin{bmatrix} 0 & I_{zx} & -I_{xy} & I_{yz} & (I_{yy}-I_{zz}) & -I_{yz} \\ -I_{zx} & -I_{yz} & (I_{zz}-I_{xx}) & 0 & I_{xy} & I_{zx} \\ I_{xy} & (I_{xx}-I_{yy}) & I_{yz} & -I_{xy} & -I_{zx} & 0 \end{bmatrix}$$

Here again, the states (p,q,r) appear in "quadratic form".

Equation (2.8) may also be written as:

$$\frac{d}{dt}\begin{bmatrix} p \\ q \\ r \end{bmatrix}=\left[[A]^{-1}[B]\right]\begin{bmatrix} p^2 \\ pq \\ pr \\ q^2 \\ qr \\ r^2 \end{bmatrix}+\left[[A]^{-1}\right]\begin{bmatrix} L \\ M \\ N \end{bmatrix} \tag{2.9}$$

where the inverse $[A]^{-1}$ is given by (see Appendix A.3):

$$[A]^{-1}=\frac{1}{\Delta}\begin{bmatrix} (I_{yy}I_{zz}-I_{yz}^2) & (I_{zz}I_{xy}+I_{yz}I_{zx}) & (I_{yz}I_{xy}+I_{yy}I_{zx}) \\ (I_{zz}I_{xy}+I_{yz}I_{zx}) & (I_{xx}I_{zz}-I_{zx}^2) & (I_{xx}I_{yz}+I_{zx}I_{xy}) \\ (I_{yz}I_{xy}+I_{yy}I_{zx}) & (I_{xx}I_{yz}+I_{xy}I_{zx}) & (I_{xx}I_{yy}-I_{xy}^2) \end{bmatrix} \tag{2.9a}$$

and $\Delta=(I_{xx}I_{yy}I_{zz}-I_{xx}I_{yz}^2-I_{yy}I_{zx}^2-I_{zz}I_{xy}^2-2I_{yz}I_{zx}I_{xy})$.

The selection of the particular order of the terms in the "quadratic-state vectors" $[uq\ ur\ vp\ vr\ wp\ wq]^T$ of Equation (2.7) and $[p^2 pq\ pr\ q^2 qr\ r^2]^T$ of Equation (2.8) is discussed in Appendix A.1.

Combining Equations (2.7) and (2.9), we obtain the full 6th order rigid body dynamics state equations as:

$$\frac{d}{dt}\begin{bmatrix} \underline{x}_1^{[1]} \\ \hline \underline{x}_2^{[1]} \end{bmatrix} = \left[\begin{array}{c:c} [C] & [0] \\ \hline [0] & [[A]^{-1}[B]] \end{array}\right]\begin{bmatrix} \underline{x}_1^{[2]} \\ \hline \underline{x}_2^{[2]} \end{bmatrix} + \left[\begin{array}{c:c} [I] & [0] \\ \hline [0] & [[A]^{-1}] \end{array}\right]\begin{bmatrix} \underline{u}_1^{[1]} \\ \hline \underline{u}_2^{[1]} \end{bmatrix} + \begin{bmatrix} \underline{g} \\ \hline \underline{0} \end{bmatrix} \tag{2.10}$$

where $[\mathbf{C}] = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & -1 \\ 0 & -1 & 0 & 0 & 1 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 \end{bmatrix}$

$$\underline{x}_1^{[1]} = [u \quad v \quad w]^T,$$

$$\underline{x}_2^{[1]} = [p \quad q \quad r]^T,$$

$$\underline{x}_1^{[2]} = [uq \quad ur \quad vp \quad vr \quad wp \quad wq]^T,$$

$$\underline{x}_2^{[2]} = [p^2 \quad pq \quad pr \quad q^2 \quad qr \quad r^2]^T,$$

$$\underline{u}_1^{[1]} = [\tilde{X} + \tilde{T} \quad \tilde{Y} \quad \tilde{Z}]^T,$$

$$\underline{u}_2^{[1]} = [L \quad M \quad N]^T,$$

$$\underline{g} = [g_x \quad g_y \quad g_z]^T.$$

Equation (2.10) may be written in a compact form as:

$$\frac{d}{dt}\underline{x}^{[1]} = [F]\underline{x}^{[2]} + [G]\underline{u}^{[1]} + g^{[1]} \tag{2.11}$$

where

$[F] = \left[\begin{array}{c:c} [C] & [0] \\ \hline [0] & [[A]^{-1}[B]] \end{array}\right]$: is the 6x12 (quadratic) state coefficient matrix.

$[G] = \left[\begin{array}{c:c} [I] & [0] \\ \hline [0] & [[A]^{-1}] \end{array}\right]$: is the 6x6 coefficient matrix.

$\underline{x}^{[1]} = \left[\underline{x}_1^{[1]} \mid \underline{x}_2^{[1]}\right]^T = \left[u \quad v \quad w \mid p \quad q \quad r\right]^T$: is the 6x1 linear-state vector.

$\underline{x}^{[2]} = \left[\underline{x}_1^{[2]} \mid \underline{x}_2^{[2]}\right]^T = \left[uq \quad ur \quad vp \quad vr \quad wp \quad wq \mid p^2 \quad pq \quad pr \quad q^2 \quad qr \quad r^2\right]^T$: is the 12x1 quadratic-state vector.

$\underline{u}^{[1]} = \left[\underline{u}_1^{[1]} \mid \underline{u}_2^{[1]}\right]^T = \left[\tilde{X}+\tilde{T} \quad \tilde{Y} \quad \tilde{Z} \mid L \quad M \quad N\right]^T$: is 6x1 a vector function of control inputs, forces and moments.

and $\underline{g}^{[1]} = \left[\underline{g} \mid \underline{0}\right]^T = \left[g_x \quad g_y \quad g_z \mid 0 \quad 0 \quad 0\right]^T$: is the 6x1 gravity (or disturbance) vector.

Note that for a two-axis symmetrical airframe, $I_{yz} = I_{zx} = I_{xy} = 0$. Hence, in this case, the equation (2.9) can be reduced to:

$$\frac{d}{dt}\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 0 & 0 & \tilde{I}_{xx} \\ 0 & \tilde{I}_{yy} & 0 \\ \tilde{I}_{zz} & 0 & 0 \end{bmatrix}\begin{bmatrix} pq \\ pr \\ qr \end{bmatrix} + \begin{bmatrix} \tilde{L} \\ \tilde{M} \\ \tilde{N} \end{bmatrix} \tag{2.12}$$

where $\tilde{I}_{xx} = \dfrac{I_{yy}-I_{zz}}{I_{xx}}$, $\tilde{I}_{yy} = \dfrac{I_{zz}-I_{xx}}{I_{yy}}$, $\tilde{I}_{zz} = \dfrac{I_{xx}-I_{yy}}{I_{zz}}$,

$$\tilde{L} = \frac{L}{I_{xx}}, \quad \tilde{M} = \frac{M}{I_{yy}}, \quad \tilde{N} = \frac{N}{I_{zz}}.$$

As a result, the state equation for the system now becomes:

$$\frac{d}{dt}\begin{bmatrix} \underline{x}_1^{[1]} \\ \underline{x}_2^{[1]} \end{bmatrix} = \begin{bmatrix} [C] & [0] \\ [0] & [H] \end{bmatrix}\begin{bmatrix} \underline{x}_1^{[2]} \\ \underline{x}_2^{[2]} \end{bmatrix} + \begin{bmatrix} [I] & [0] \\ [0] & [I] \end{bmatrix}\begin{bmatrix} \underline{u}_1^{[1]} \\ \underline{u}_2^{[1]} \end{bmatrix} + \begin{bmatrix} \underline{g} \\ \underline{0} \end{bmatrix} \tag{2.13}$$

where: $[H] = \begin{bmatrix} 0 & 0 & \tilde{I}_{xx} \\ 0 & \tilde{I}_{yy} & 0 \\ \tilde{I}_{zz} & 0 & 0 \end{bmatrix}$, $\underline{x}_2^{[2]} = \left[pq \quad pr \quad qr\right]^T$, and

$$\underline{u}_2^{[l]} = \begin{bmatrix} \widetilde{L} & \widetilde{M} & \widetilde{N} \end{bmatrix}^T.$$

Remarks:

Equations (2.11) and (2.13) are complete non-linear description of the full 6-DOF autopilot model. In fact, these equations contain quadratic terms in states and will be classed as the quadratic dynamic model. This type of model is required when autopilot design is undertaken for a missile executing high g- or high angle of attack manoeuvres, and (u, v, w, p, q, r) are not small.

A more detailed consideration of the algebraic structure of this type of dynamic systems is given in [4].

2.3 Linearised model for a two-axis symmetrical airframe

It is assumed that $\widetilde{X}$, $\widetilde{Y}$, $\widetilde{Z}$, $\widetilde{L}$, $\widetilde{M}$ and $\widetilde{N}$ are functions of u, v, w, p, q, r, ξ, η and ς. Using first order linearisation about the nominal values u_0, v_0, w_0, p_0, q_0, r_0, ξ_0, η_0 and ς_0, and defining the aerodynamic derivatives as:

$$\widetilde{X}_u = \frac{\partial \widetilde{X}}{\partial u},\ \widetilde{X}_v = \frac{\partial \widetilde{X}}{\partial v},\ \widetilde{X}_w = \frac{\partial \widetilde{X}}{\partial w},\ \widetilde{X}_p = \frac{\partial \widetilde{X}}{\partial p},\ \widetilde{X}_q = \frac{\partial \widetilde{X}}{\partial q},\ \widetilde{X}_r = \frac{\partial \widetilde{X}}{\partial r},$$
$$\widetilde{X}_\xi = \frac{\partial \widetilde{X}}{\partial \xi},\ \widetilde{X}_\eta = \frac{\partial \widetilde{X}}{\partial \eta},\ \widetilde{X}_\zeta = \frac{\partial \widetilde{X}}{\partial_\varsigma},$$

$$\widetilde{Y}_u = \frac{\partial \widetilde{Y}}{\partial u},\ \widetilde{Y}_v = \frac{\partial \widetilde{Y}}{\partial v},\ \widetilde{Y}_w = \frac{\partial \widetilde{Y}}{\partial w},\ \widetilde{Y}_p = \frac{\partial \widetilde{Y}}{\partial p},\ \widetilde{Y}_q = \frac{\partial \widetilde{Y}}{\partial q},\ \widetilde{Y}_r = \frac{\partial \widetilde{Y}}{\partial r},$$
$$\widetilde{Y}_\xi = \frac{\partial \widetilde{Y}}{\partial \xi},\ \widetilde{Y}_\eta = \frac{\partial \widetilde{Y}}{\partial \eta},\ \widetilde{Y}_\zeta = \frac{\partial \widetilde{Y}}{\partial \zeta},$$

$$\widetilde{Z}_u = \frac{\partial \widetilde{Z}}{\partial u},\ \widetilde{Z}_v = \frac{\partial \widetilde{Z}}{\partial v},\ \widetilde{Z}_w = \frac{\partial \widetilde{Z}}{\partial w},\ \widetilde{Z}_p = \frac{\partial \widetilde{Z}}{\partial p},\ \widetilde{Z}_q = \frac{\partial \widetilde{Z}}{\partial q},\ \widetilde{Z}_r = \frac{\partial \widetilde{Z}}{\partial r},$$
$$\widetilde{Z}_\xi = \frac{\partial \widetilde{Z}}{\partial \xi},\ \widetilde{Z}_\eta = \frac{\partial \widetilde{Z}}{\partial \eta},\ \widetilde{Z}_\zeta = \frac{\partial \widetilde{Z}}{\partial \zeta},$$

$$\widetilde{L}_u = \frac{\partial \widetilde{L}}{\partial u},\ \widetilde{L}_v = \frac{\partial \widetilde{L}}{\partial v},\ \widetilde{L}_w = \frac{\partial \widetilde{L}}{\partial w},\ \widetilde{L}_p = \frac{\partial \widetilde{L}}{\partial p},\ \widetilde{L}_q = \frac{\partial \widetilde{L}}{\partial q},\ \widetilde{L}_r = \frac{\partial \widetilde{L}}{\partial r},$$
$$\widetilde{L}_\xi = \frac{\partial \widetilde{L}}{\partial \xi},\ \widetilde{L}_\eta = \frac{\partial \widetilde{L}}{\partial \eta},\ \widetilde{L}_\zeta = \frac{\partial \widetilde{L}}{\partial \zeta},$$

$$\widetilde{M}_u = \frac{\partial \widetilde{M}}{\partial u},\ \widetilde{M}_v = \frac{\partial \widetilde{M}}{\partial v},\ \widetilde{M}_w = \frac{\partial \widetilde{M}}{\partial w},\ \widetilde{M}_p = \frac{\partial \widetilde{M}}{\partial p},\ \widetilde{M}_q = \frac{\partial \widetilde{M}}{\partial q},\ \widetilde{M}_r = \frac{\partial \widetilde{M}}{\partial r},$$
$$\widetilde{M}_\xi = \frac{\partial \widetilde{M}}{\partial \xi},\ \widetilde{M}_\eta = \frac{\partial \widetilde{M}}{\partial \eta},\ \widetilde{M}_\zeta = \frac{\partial \widetilde{M}}{\partial \zeta},$$

$$\widetilde{N}_u = \frac{\partial \widetilde{N}}{\partial u},\ \widetilde{N}_v = \frac{\partial \widetilde{N}}{\partial v},\ \widetilde{N}_w = \frac{\partial \widetilde{N}}{\partial w},\ \widetilde{N}_p = \frac{\partial \widetilde{N}}{\partial p},\ \widetilde{N}_q = \frac{\partial \widetilde{N}}{\partial q},\ \widetilde{N}_r = \frac{\partial \widetilde{N}}{\partial r},$$
$$\widetilde{N}_\xi = \frac{\partial \widetilde{N}}{\partial \xi},\ \widetilde{N}_\eta = \frac{\partial \widetilde{N}}{\partial \eta},\ \widetilde{N}_\zeta = \frac{\partial \widetilde{N}}{\partial \zeta}.$$

The six equations of motion of an airframe (using equation (2.13)) can thus be written as:

$$\begin{aligned}\Delta\dot{u} &= r_0\Delta v + v_0\Delta r - q_0\Delta w - w_0\Delta q \\ &+ \left(\widetilde{X}_u\Delta u + \widetilde{X}_v\Delta v + \widetilde{X}_w\Delta w + \widetilde{X}_p\Delta p + \widetilde{X}_q\Delta q + \widetilde{X}_r\Delta r + \widetilde{X}_\xi\Delta\xi + \widetilde{X}_\eta\Delta\eta + \widetilde{X}_\varsigma\Delta\varsigma\right) \\ &+ \Delta\widetilde{T} + \Delta g_x\end{aligned} \tag{2.14a}$$

$$\begin{aligned}\Delta\dot{v} &= p_0\Delta w + w_0\Delta p - r_0\Delta u - u_0\Delta r \\ &+ \left(\widetilde{Y}_u\Delta u + \widetilde{Y}_v\Delta v + \widetilde{Y}_w\Delta w + \widetilde{Y}_p\Delta p + \widetilde{Y}_q\Delta q + \widetilde{Y}_r\Delta r + \widetilde{Y}_\xi\Delta\xi + \widetilde{Y}_\eta\Delta\eta + \widetilde{Y}_\varsigma\Delta\varsigma\right) + \Delta g_y\end{aligned} \tag{2.14b}$$

$$\begin{aligned}\Delta\dot{w} &= q_0\Delta u + u_0\Delta q - p_0\Delta v - v_0\Delta p \\ &+ \left(\widetilde{Z}_u\Delta u + \widetilde{Z}_v\Delta v + \widetilde{Z}_w\Delta w + \widetilde{Z}_p\Delta p + \widetilde{Z}_q\Delta q + \widetilde{Z}_r\Delta r + \widetilde{Z}_\xi\Delta\xi + \widetilde{Z}_\eta\Delta\eta + \widetilde{Z}_\varsigma\Delta\varsigma\right) + \Delta g_z\end{aligned} \tag{2.14c}$$

$$\begin{aligned}\Delta\dot{p} &= \widetilde{I}_{xx}\left(q_0\Delta r + r_0\Delta q\right) \\ &+ \left(\widetilde{L}_u\Delta u + \widetilde{L}_v\Delta v + \widetilde{L}_w\Delta w + \widetilde{L}_p\Delta p + \widetilde{L}_q\Delta q + \widetilde{L}_r\Delta r + \widetilde{L}_\xi\Delta\xi + \widetilde{L}_\eta\Delta\eta + \widetilde{L}_\varsigma\Delta\varsigma\right)\end{aligned} \tag{2.14d}$$

$$\begin{aligned}\Delta\dot{q} &= \widetilde{I}_{yy}\left(r_0\Delta p + p_0\Delta r\right) \\ &+ \left(\widetilde{M}_u\Delta u + \widetilde{M}_v\Delta v + \widetilde{M}_w\Delta w + \widetilde{M}_p\Delta p + \widetilde{M}_q\Delta q + \widetilde{M}_r\Delta r + \widetilde{M}_\xi\Delta\xi + \widetilde{M}_\eta\Delta\eta + \widetilde{M}_\varsigma\Delta\varsigma\right)\end{aligned} \tag{2.14e}$$

$$\begin{aligned}\Delta\dot{r} &= \widetilde{I}_{zz}\left(p_0\Delta q + q_0\Delta p\right) \\ &+ \left(\widetilde{N}_u\Delta u + \widetilde{N}_v\Delta v + \widetilde{N}_w\Delta w + \widetilde{N}_p\Delta p + \widetilde{N}_q\Delta q + \widetilde{N}_r\Delta r + \widetilde{N}_\xi\Delta\xi + \widetilde{N}_\eta\Delta\eta + \widetilde{N}_\varsigma\Delta\varsigma\right)\end{aligned} \tag{2.14f}$$

Equations (2.14a to 2.14f) may be written in a matrix notation as:

$$\begin{bmatrix}\Delta\dot{u}\\ \Delta\dot{v}\\ \Delta\dot{w}\\ \Delta\dot{p}\\ \Delta\dot{q}\\ \Delta\dot{r}\end{bmatrix}=\begin{bmatrix}\tilde{X}_u & (r_0+\tilde{X}_v) & (-q_0+\tilde{X}_w) & \tilde{X}_p & (-w_0+\tilde{X}_q) & (v_0+\tilde{X}_r)\\ (-r_0+\tilde{Y}_u) & \tilde{Y}_v & (p_0+\tilde{Y}_w) & (w_0+\tilde{Y}_p) & \tilde{Y}_q & (-u_0+\tilde{Y}_r)\\ (q_0+\tilde{Z}_u) & (-p_0+\tilde{Z}_v) & \tilde{Z}_w & (-v_0+\tilde{Z}_p) & (u_0+\tilde{Z}_q) & \tilde{Z}_r\\ \tilde{L}_u & \tilde{L}_v & \tilde{L}_w & \tilde{L}_p & (\tilde{I}_{xx}r_0+\tilde{L}_q) & (\tilde{I}_{xx}q_0+\tilde{L}_r)\\ \tilde{M}_u & \tilde{M}_v & \tilde{M}_w & (\tilde{I}_{yy}r_0+\tilde{M}_p) & \tilde{M}_q & (\tilde{I}_{yy}p_0+\tilde{M}_r)\\ \tilde{N}_u & \tilde{N}_v & \tilde{N}_w & (\tilde{I}_{zz}q_0+\tilde{N}_p) & (\tilde{I}_{zz}p_0+\tilde{N}_q) & \tilde{N}_r\end{bmatrix}\begin{bmatrix}\Delta u\\ \Delta v\\ \Delta w\\ \Delta p\\ \Delta q\\ \Delta r\end{bmatrix}+$$

$$\begin{bmatrix}\tilde{X}_\xi & \tilde{X}_\eta & \tilde{X}_\varsigma\\ \tilde{Y}_\xi & \tilde{Y}_\eta & \tilde{Y}_\varsigma\\ \tilde{Z}_\xi & \tilde{Z}_\eta & \tilde{Z}_\varsigma\\ \tilde{L}_\xi & \tilde{L}_\eta & \tilde{L}_\varsigma\\ \tilde{M}_\xi & \tilde{M}_\eta & \tilde{M}_\varsigma\\ \tilde{N}_\xi & \tilde{N}_\eta & \tilde{N}_\varsigma\end{bmatrix}\begin{bmatrix}\Delta\xi\\ \Delta\eta\\ \Delta\varsigma\end{bmatrix}+\begin{bmatrix}\Delta\tilde{T}+\Delta g_x\\ \Delta g_y\\ \Delta g_z\\ 0\\ 0\\ 0\end{bmatrix}$$

Note that Equation (2.15) is represented in a state-space form as:

$$\frac{d}{dt}\Delta x=[F_l]\Delta x+[G_l]\Delta u_l+\Delta w_l \qquad (2.16)$$

where $\Delta x=\begin{bmatrix}\Delta u\\ \Delta v\\ \Delta w\\ \Delta p\\ \Delta q\\ \Delta r\end{bmatrix}$, $\Delta u_l=\begin{bmatrix}\Delta\xi\\ \Delta\eta\\ \Delta\varsigma\end{bmatrix}$, $\Delta w_l=\begin{bmatrix}\Delta\tilde{T}+\Delta g_x\\ \Delta g_y\\ \Delta g_z\\ 0\\ 0\\ 0\end{bmatrix}$,

$$[F_1]=\begin{bmatrix} \tilde{X}_u & (r_0+\tilde{X}_v) & -q_0+\tilde{X}_w & \tilde{X}_p & -w_0+\tilde{X}_q & v_0+\tilde{X}_r \\ -r_0+\tilde{Y}_u & \tilde{Y}_v & p_0+\tilde{Y}_w & w_0+\tilde{Y}_p & \tilde{Y}_q & -u_0+\tilde{Y}_r \\ q_0+\tilde{Z}_u & -p_0+\tilde{Z}_v & \tilde{Z}_w & -v_0+\tilde{Z}_p & u_0+\tilde{Z}_q & \tilde{Z}_r \\ \tilde{L}_u & \tilde{L}_v & \tilde{L}_w & \tilde{L}_p & \tilde{I}_{xx}r_0+\tilde{L}_q & \tilde{I}_{xx}q_0+\tilde{L}_r \\ \tilde{M}_u & \tilde{M}_v & \tilde{M}_w & \tilde{I}_{yy}r_0+\tilde{M}_p & \tilde{M}_q & \tilde{I}_{yy}p_0+\tilde{M}_r \\ \tilde{N}_u & \tilde{N}_v & \tilde{N}_w & \tilde{I}_{zz}q_0+\tilde{N}_p & \tilde{I}_{zz}p_0+\tilde{N}_q & \tilde{N}_r \end{bmatrix},$$

$$[G_1]=\begin{bmatrix} \tilde{X}_\xi & \tilde{X}_\eta & \tilde{X}_\varsigma \\ \tilde{Y}_\xi & \tilde{Y}_\eta & \tilde{Y}_\varsigma \\ \tilde{Z}_\xi & \tilde{Z}_\eta & \tilde{Z}_\varsigma \\ \tilde{L}_\xi & \tilde{L}_\eta & \tilde{L}_\varsigma \\ \tilde{M}_\xi & \tilde{M}_\eta & \tilde{M}_\varsigma \\ \tilde{N}_\xi & \tilde{N}_\eta & \tilde{N}_\varsigma \end{bmatrix},$$

$\tilde{T}=\tilde{T}_0+\Delta\tilde{T}$, $g_x=g_{x0}+\Delta g_x$, $g_y=g_{y0}+\Delta g_y$, $g_z=g_{z0}+\Delta g_z$.

2.4 Incorporation of accelerometer and gyro measurement model

Generally, not all state variables in the state equation are accessible or measurable. The common measurement variables, in most missiles or airplanes, are the angular rate components (roll rate, p, pitch rate, q, and yaw rate, r) and the acceleration components (a_x, a_y, a_z).

Assuming that the gyros provide ideal readings of the angular rates, we get:

$$p_m = p \tag{2.17a}$$

$$q_m = q \tag{2.17b}$$

$$r_m = r \tag{2.17c}$$

where p_m, q_m and r_m are the measured body rates. Normally, errors due to drifts and noise are included. These appear as additional additive terms in equations (2.17a) to (2.17c).

In contrast to the readings of the angular rate components, the readings of the acceleration components are dependent on the location of the accelerometers, w.r.t. the c.g. of the body.

The acceleration components measured at point O (where O is at a distance of d_x, d_y and d_z from the central point of gravity, c.g., along x-, y- and z-axis, respectively), may be written as:

$$a_x = \dot{u} + qw - rv - d_x(q^2 + r^2) + d_y(pq - \dot{r}) + d_z(pr + \dot{q}) \tag{2.18a}$$

$$a_y = \dot{v} + ru - pw + d_x(pq + \dot{r}) - d_y(p^2 + r^2) + d_z(qr - \dot{p}) \tag{2.18b}$$

$$a_z = \dot{w} + pv - qu + d_x(pr - \dot{q}) + d_y(qr + \dot{p}) - d_z(p^2 + q^2) \tag{2.18c}$$

If the accelerometers are mounted along the x-axis (ie. $dy = dz = 0$) which is usually the case, then equations (2.18a-c) reduce to:

$$a_x = \dot{u} + qw - rv - d_x(q^2 + r^2) = \tilde{X} + \tilde{T} + g_x - d_x(q^2 + r^2) \tag{2.19a}$$

$$a_y = \dot{v} + ru - pw + d_x(pq + \dot{r}) = \tilde{Y} + g_y + d_x(pq + \dot{r}) \tag{2.19b}$$

$$a_z = \dot{w} + pv - qu + d_x(pr - \dot{q}) = \tilde{Z} + g_z + d_x(pr - \dot{q}) \tag{2.19c}$$

Note that the right hand side of Equations (2.19a) to (2.19c) come directly from Equations (2.1) to (2.3).

Linearising Equations (2.19a) to (2.19c), and using the relationship (2.15) gives us:

$$\Delta y(t) = [H_l]\Delta x(t) + [J_l]\Delta u(t) + \Delta v(t) \tag{2.20}$$

where: $\Delta y(t) = [\Delta p \quad \Delta q \quad \Delta r \quad \Delta a_x \quad \Delta a_y \quad \Delta a_z]^T$: is the output vector,

$$[H_l] = \begin{bmatrix} 0 & 0 & 0 & \tilde{X}_u & \tilde{Y}_u + \tilde{N}_u d_x & \tilde{Z}_u - \tilde{M}_u d_x \\ 0 & 0 & 0 & \tilde{X}_v & \tilde{Y}_v + \tilde{N}_v d_x & \tilde{Z}_v - \tilde{M}_v d_x \\ 0 & 0 & 0 & \tilde{X}_w & \tilde{Y}_w + \tilde{N}_w d_x & \tilde{Z}_w - \tilde{M}_w d_x \\ 1 & 0 & 0 & \tilde{X}_p & \left[\tilde{Y}_p + (q_0 + \tilde{I}_{zz} q_0 + \tilde{N}_p) d_x\right] & \left[\tilde{Z}_p + (r_0 - \tilde{I}_{yy} r_0 - \tilde{M}_p) d_x\right] \\ 0 & 1 & 0 & \tilde{X}_q - 2q_0 d_x & \left[\tilde{Y}_q + (p_0 + \tilde{I}_{zz} p_0 + \tilde{N}_q) d_x\right] & \tilde{Z}_q - \tilde{M}_q d_x \\ 0 & 0 & 1 & \tilde{X}_r - 2r_0 d_x & \tilde{Y}_r + \tilde{N}_r d_x & \left[\tilde{Z}_r + (p_0 - \tilde{I}_{yy} p_0 - \tilde{M}_r) d_x\right] \end{bmatrix}^T$$: is

the state output matrix,

$$[J_l]=\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \tilde{X}_\xi & \tilde{X}_\eta & \tilde{X}_\varsigma \\ \tilde{Y}_\xi+\tilde{N}_\xi d_x & \tilde{Y}_\eta+\tilde{N}_\eta d_x & \tilde{Y}_\varsigma+\tilde{N}_\varsigma d_x \\ \tilde{Z}_\xi-\tilde{M}_\xi d_x & \tilde{Z}_\eta-\tilde{M}_\eta d_x & \tilde{Z}_\varsigma-\tilde{M}_\varsigma d_x \end{bmatrix}$$: is the matrix related to inputs in the measurement matrix,

$\Delta u_l(t)=[\Delta\xi \quad \Delta\eta \quad \Delta\varsigma]^T$: is the input vector, and

$\Delta v_l(t)=\left[0 \quad 0 \quad 0 \quad \dfrac{\Delta T}{m}+\Delta g_x \quad \Delta g_y \quad \Delta g_z\right]^T$: is the disturbance vector.

2.5 Linearised model of the airframe including fin servos

Assuming that the servo dynamics for the aileron, elevator and rudder can be described adequately by a second order lag as:

$$\frac{\Delta\xi}{\Delta\xi_d}=\frac{k_{s\xi}}{\dfrac{s^2}{\omega_{s\xi}^{2}}+\dfrac{2\mu_{s\xi}s}{\omega_{s\xi}}+1}, \tag{2.21a}$$

$$\frac{\Delta\eta}{\Delta\eta_d}=\frac{k_{s\eta}}{\dfrac{s^2}{\omega_{s\eta}^{2}}+\dfrac{2\mu_{s\eta}s}{\omega_{s\eta}}+1}, \tag{2.21b}$$

$$\frac{\Delta\zeta}{\Delta\zeta_d}=\frac{k_{s\zeta}}{\dfrac{s^2}{\omega_{s\zeta}^{2}}+\dfrac{2\mu_{s\zeta}s}{\omega_{s\zeta}}+1}, \tag{2.21c}$$

where $\Delta\xi_d$, $\Delta\eta_d$ and $\Delta\zeta_d$ are the demand aileron, elevator and rudder deflection, respectively.

$k_{s\xi}$, $k_{s\eta}$, and $k_{s\zeta}$ are the servo gain for the aileron, elevator and rudder, respectively.

$\mu_{s\xi}$, $\mu_{s\eta}$, and $\mu_{s\zeta}$ are the damping factor for the aileron, elevator and rudder, respectively.

$\omega_{s\xi}$, $\omega_{s\eta}$, and $\omega_{s\zeta}$ are the natural frequency for the aileron, elevator and rudder, respectively.

Equations (2.21a) to (2.21c) can be converted into differential equations as follows:

$$\Delta\ddot{\xi} = -\omega_{s\xi}^{2}\Delta\xi - 2\mu_{s\xi}\omega_{s\xi}\Delta\dot{\xi} + k_{s\xi}\omega_{s\xi}^{2}\Delta\xi_{d}, \tag{2.22a}$$

$$\Delta\ddot{\eta} = -\omega_{s\eta}^{2}\Delta\eta - 2\mu_{s\eta}\omega_{s\eta}\Delta\dot{\eta} + k_{s\eta}\omega_{s\eta}^{2}\Delta\eta_{d}, \tag{2.22b}$$

$$\Delta\ddot{\zeta} = -\omega_{s\zeta}^{2}\Delta\zeta - 2\mu_{s\zeta}\omega_{s\zeta}\Delta\dot{\zeta} + k_{s\zeta}\omega_{s\zeta}^{2}\Delta\zeta_{d}. \tag{2.22c}$$

Hence, the state-space model for the autopilot of a missile including the servos and airframe is:

$$\Delta\dot{x}_2(t) = [A_2]\Delta x_2(t) + [B_2]\Delta u_2(t) + \Delta w_2(t), \tag{2.23}$$

where

$$\Delta x_2(t) = \begin{bmatrix}\Delta u & \Delta v & \Delta w & \Delta p & \Delta q & \Delta r & \Delta\xi & \Delta\eta & \Delta\zeta & \Delta\dot{\xi} & \Delta\dot{\eta} & \Delta\dot{\zeta}\end{bmatrix}^T,$$

Note that the aileron, elevator and rudder deflection now become state variables. Hence, the dimension of the state vector is increased to [12 × 1].

$\Delta u_2(t) = \begin{bmatrix}\Delta\xi_d & \Delta\eta_d & \Delta\zeta_d\end{bmatrix}^T,$

The inputs are now the ***demanded*** aileron, elevator and rudder deflection.

$$\Delta w_2(t) = \begin{bmatrix}\dfrac{\Delta T}{m} + \Delta g_x & \Delta g_y & \Delta g_z & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T,$$

$$[A_2]=\left[\begin{array}{cccccc:ccc:ccc} & & [F_1] & & & & & [G_1] & & 0 & 0 & 0 \\ & & & & & & & & & 0 & 0 & 0 \\ & & & & & & & & & 0 & 0 & 0 \\ & & & & & & & & & 0 & 0 & 0 \\ & & & & & & & & & 0 & 0 & 0 \\ & & & & & & & & & 0 & 0 & 0 \\ \hdashline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\omega_{s\xi}^{2} & 0 & 0 & -2\mu_{s\xi}\omega_{s\xi} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\omega_{s\eta}^{2} & 0 & 0 & -2\mu_{s\eta}\omega_{s\eta} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\omega_{s\zeta}^{2} & 0 & 0 & -2\mu_{s\zeta}\omega_{s\zeta} \end{array}\right]$$

(Here matrices F_1 and G_1 are the same as in Equation (2.16)).

$$[B_2]=\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ k_{s\xi}\omega_{s\xi}^{2} & 0 & 0 \\ 0 & k_{s\eta}\omega_{s\eta}^{2} & 0 \\ 0 & 0 & k_{s\zeta}\omega_{s\zeta}^{2} \end{bmatrix}.$$

The output vector (or the measurement equation) is given by:

$$\Delta y_2(t) = H_2 \Delta x_2(t) + \Delta v_2(t), \tag{2.24}$$

where $\Delta y_2(t) = \begin{bmatrix}\Delta p_m & \Delta q_m & \Delta r_m & \Delta a_{xm} & \Delta a_{ym} & \Delta a_{zm}\end{bmatrix}^T$, (Note that gyro drift and noise and the accelerometer bias may be added to the right hand side of Equation (2.24)).

$$[H_2] = \left[\begin{array}{c:c:ccc} & & 0 & 0 & 0 \\ & & 0 & 0 & 0 \\ & & 0 & 0 & 0 \\ [H_1] & [J_1] & 0 & 0 & 0 \\ & & 0 & 0 & 0 \\ & & 0 & 0 & 0 \end{array} \right],$$

(Here matrices H_1 and J_1 are the same as those in Equation (2.20)).

$$\Delta v_2(t) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dfrac{\Delta T}{m} + \Delta g_x & \Delta g_y & \Delta g_z \end{bmatrix}^T .$$

Figure 2.3 shows the block diagram of an open-loop autopilot which contains the fin servos and airframe.

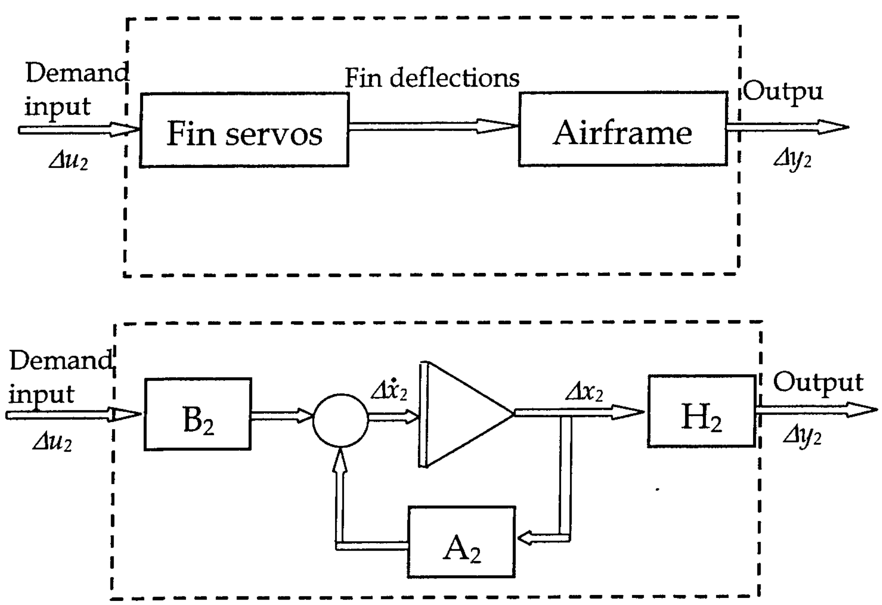

Figure 2.3 A block diagram of an open-loop autopilot system.

2.6 Lateral Auto-pilot design

For the case of small perturbation, we may assume that (u_0, v_0, w_0, p_0, q_0, r_0) are identically zero. In this case, the airframe model decouples into two lateral dynamics (pitch and yaw) and one roll dynamics. We will consider the lateral autopilot dynamics to validate the model derived in this report.

Figure 2.4 shows the block diagram of a closed-loop autopilot system.

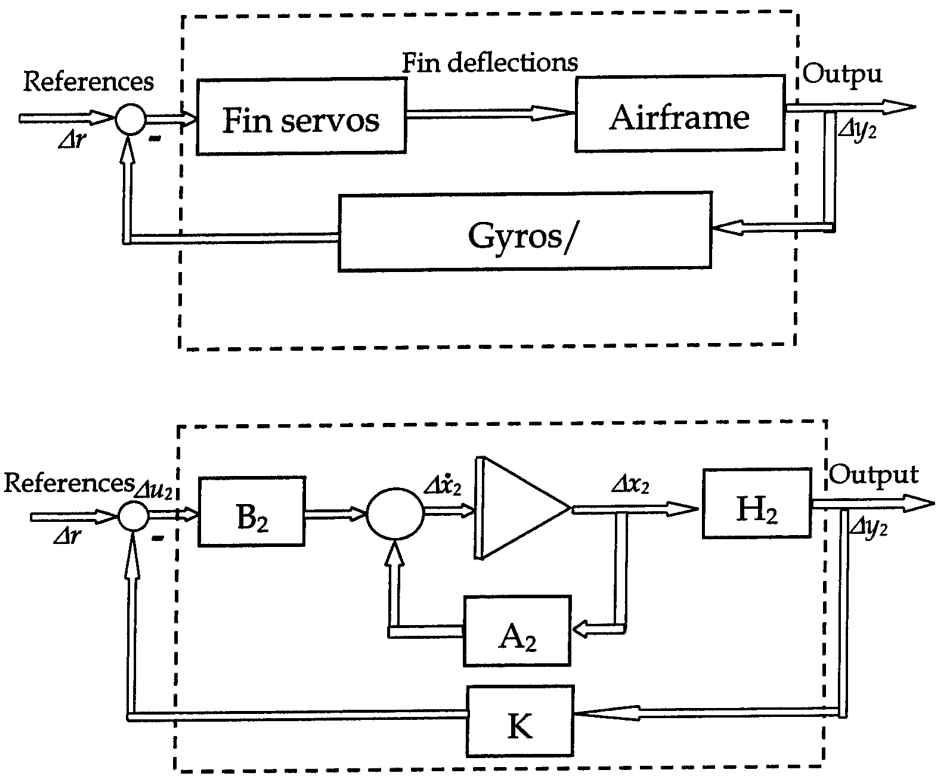

Figure 2.4 A block diagram of a closed-loop auto-pilot system.

Ignoring the instrument (gyro, accelerometer) dynamics, the measured roll, pitch and yaw angular rates (the gyro outputs) can be expressed as inputs to the gyros multiplied the gyro gains, K_{gr}, K_{gp} and K_{gy}, respectively. Similarly, the measured longitudinal acceleration, a_x, and lateral accelerations, a_y and a_z, are inputs to the accelerometers multiplied accelerometer gains, K_{ax}, K_{ay} and K_{az}, respectively. The accelerometer gains affect the steady state response and may be set to 1 for transient tests. Rescaling accelerometer gains, after selecting gyro gains, allows a unity gain autopilot to be designed.

The reference signals, generally used for testing the transient time response of the autopilot, are the desired accelerations in yaw direction, a_{yd}, the pitch direction, a_{zd}, and roll rate, p_d. The reference roll rate is kept at zero to assess the missile dynamics in roll against spurious disturbances. Hence, the reference vector, Δr, can be written as:

$$\Delta r = \begin{bmatrix} \Delta p_d & \Delta a_{zd} & \Delta a_{yd} \end{bmatrix}^T,$$

For a case of lateral directional control, the control input signal for the fin servos can be derived as follows:

$$\Delta\eta_d = \Delta a_{zd} - K_{az}\Delta a_z - K_{gq}\Delta q \tag{2.25a}$$

$$\Delta\varsigma_d = \Delta a_{yd} - K_{ay}\Delta a_y - K_{gr}\Delta r \tag{2.25b}$$

For sake of simplicity, p_d is set to zero since this case only considers the lateral directional control. As a result, the control input vector can be written as:

$$\Delta u_2(t) = \Delta r(t) - K\Delta y_2(t), \tag{2.26}$$

where K is the feedback matrix as follows:

$$K = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & K_{gq} & 0 & 0 & 0 & K_{az} \\ 0 & 0 & K_{gr} & 0 & K_{ay} & 0 \end{bmatrix}.$$

3. Verification of the Developed Model

In order to verify the developed model, the state-space model (Equations (2.23, 2.24 and 2.26)) was converted into transfer-function form using Matlab symbolic toolbox (see Appendix B and Appendix C.1) for comparison with the results already published [4].

Consider the following derivatives and variables to be non-zero:

$$\widetilde{L}_u, \widetilde{L}_p, \widetilde{L}_\xi, \widetilde{M}_q, \widetilde{M}_w, \widetilde{M}_\eta, \widetilde{N}_r, \widetilde{N}_v, \widetilde{N}_\zeta, \widetilde{X}_u, \widetilde{X}_p, \widetilde{X}_\xi, \widetilde{Y}_r, \widetilde{Y}_v, \widetilde{Y}_\zeta, \widetilde{Z}_w, \widetilde{Z}_\eta, \widetilde{Z}_q, u_0, d_x,$$

we obtain the transfer-function between the roll rate and the aileron deflection as:

$$\frac{p(s)}{\xi(s)} = \frac{\widetilde{L}_\xi s + (\widetilde{L}_u + \widetilde{X}_\xi - \widetilde{X}_u\widetilde{L}_\xi)}{s^2 - (\widetilde{L}_p + \widetilde{X}_u)s + (\widetilde{X}_u\widetilde{L}_p - \widetilde{X}_p\widetilde{L}_u)}, \tag{2.27}$$

The transfer-function between the pitch rate and the elevator deflection as:

$$\frac{q(s)}{\eta(s)}=\frac{\widetilde{M}_{\eta}s-\widetilde{M}_{\eta}\widetilde{Z}_{w}+\widetilde{M}_{w}\widetilde{Z}_{\eta}}{s^{2}-(\widetilde{M}_{q}+\widetilde{Z}_{w})s+\widetilde{M}_{q}\widetilde{Z}_{w}-\widetilde{M}_{w}\widetilde{Z}_{q}-\widetilde{M}_{w}u_{0}}, \tag{2.28}$$

The transfer-function between the yaw rate and the rudder deflection as:

$$\frac{r(s)}{\zeta(s)}=\frac{\widetilde{N}_{\zeta}s+\widetilde{N}_{v}\widetilde{Y}_{\zeta}-\widetilde{N}_{\zeta}\widetilde{Y}_{v}}{s^{2}-(\widetilde{N}_{r}+\widetilde{Y}_{v})s+\widetilde{N}_{v}u_{0}+\widetilde{N}_{r}\widetilde{Y}_{v}-\widetilde{N}_{v}\widetilde{Y}_{r}}, \tag{2.29}$$

The transfer-function between longitudinal acceleration and the aileron deflection as:

$$\frac{a_{x}(s)}{\xi(s)}=\frac{(\widetilde{X}_{u}\widetilde{X}_{\xi}+\widetilde{X}_{p}\widetilde{X}_{\xi}\widetilde{L}_{\xi})s-\widetilde{X}_{u}\widetilde{X}_{\xi}\widetilde{L}_{p}+\widetilde{X}_{p}\widetilde{X}_{\xi}\widetilde{L}_{u}}{s^{2}-(\widetilde{X}_{u}+\widetilde{L}_{p})s+\widetilde{X}_{u}\widetilde{L}_{p}-\widetilde{X}_{p}\widetilde{L}_{u}}, \tag{230}$$

The transfer-function between lateral acceleration a_y and the rudder deflection as:

$$\frac{a_{y}(s)}{\zeta(s)}=\frac{(\widetilde{Y}_{\zeta}+\widetilde{N}_{\zeta}dx)s^{2}+(\widetilde{N}_{v}\widetilde{Y}_{\zeta}dx-\widetilde{N}_{\zeta}\widetilde{Y}_{v}dx+\widetilde{N}_{\zeta}\widetilde{Y}_{r}-\widetilde{N}_{r}\widetilde{Y}_{\zeta})s+\widetilde{N}_{v}\widetilde{Y}_{\zeta}u_{0}-\widetilde{N}_{\zeta}\widetilde{Y}_{v}u_{0}}{s^{2}-(\widetilde{N}_{r}+\widetilde{Y}_{v})s+\widetilde{N}_{v}u_{0}+\widetilde{N}_{r}\widetilde{Y}_{v}-\widetilde{N}_{v}\widetilde{Y}_{r}}, \tag{2.31}$$

And the transfer-function between lateral acceleration a_z and the elevator deflection as:

$$\frac{a_{z}(s)}{\eta(s)}=\frac{(\widetilde{Z}_{\eta}+\widetilde{M}_{\eta}dx)s^{2}+(\widetilde{M}_{w}\widetilde{Z}_{\eta}dx-\widetilde{M}_{\eta}\widetilde{Z}_{w}dx+\widetilde{M}_{\eta}\widetilde{Z}_{q}-\widetilde{M}_{q}\widetilde{Z}_{\eta})s+\widetilde{M}_{\eta}\widetilde{Z}_{w}u_{0}-\widetilde{M}_{w}\widetilde{Z}_{\eta}u_{0}}{s^{2}-(\widetilde{M}_{q}+\widetilde{Z}_{w})s-\widetilde{M}_{w}u_{0}-\widetilde{M}_{w}\widetilde{Z}_{q}+\widetilde{M}_{q}\widetilde{Z}_{w}} \tag{2.32}$$

Furthermore, if it is assumed that $\widetilde{X}_{u}=\widetilde{X}_{\xi}=\widetilde{Y}_{r}=\widetilde{L}_{u}=0$, the transfer-function between the roll rate and the aileron deflection may be simplified to:

$$\frac{p(s)}{\xi(s)}=\frac{\widetilde{L}_{\xi}}{s-\widetilde{L}_{p}}, \tag{2.33}$$

The transfer-function between the yaw rate and the rudder deflection as:

$$\frac{r(s)}{\zeta(s)} = \frac{\widetilde{N}_\zeta s + \widetilde{N}_v \widetilde{Y}_\zeta - \widetilde{N}_\zeta \widetilde{Y}_v}{s^2 - (\widetilde{N}_r + \widetilde{Y}_v)s + \widetilde{N}_v u_0 + \widetilde{N}_r \widetilde{Y}_v}, \tag{2.34}$$

And the transfer-function between the lateral acceleration in yaw axis, a_y, and the rudder deflection measured at the c.g. can be rewritten as:

$$\frac{a_y(s)}{\zeta(s)} = \frac{\widetilde{Y}_\zeta s^2 - \widetilde{N}_r \widetilde{Y}_\zeta s + \widetilde{N}_v \widetilde{Y}_\zeta u_0 - \widetilde{N}_\zeta \widetilde{Y}_v u_0}{s^2 - (\widetilde{N}_r + \widetilde{Y}_v)s + \widetilde{N}_v u_0 + \widetilde{N}_r \widetilde{Y}_v}. \tag{2.35}$$

Equations (2.33) to (2.35) are identical to those of the transfer-functions presented in [P.Garnell and D.J.East [Equations (4.6-6), (4.6-8) and (4.6-7)].

The state-space model was used for simulation of open-loop and closed-loop responses for a typical missile, using the same values as those used in [4] (see Appendices C and D).

Figures 2.5 and 2.6 show the lateral accelerations of the missile due to a step input to the rudder and elevator, respectively, for an open loop simulation. As can be seen from these figures, there are large steady state errors. However, the steady state errors can be reduced with a feedback loop as can been seen in Figures 2.7 and 2.8. These simulation results are similar to the results presented in [4].

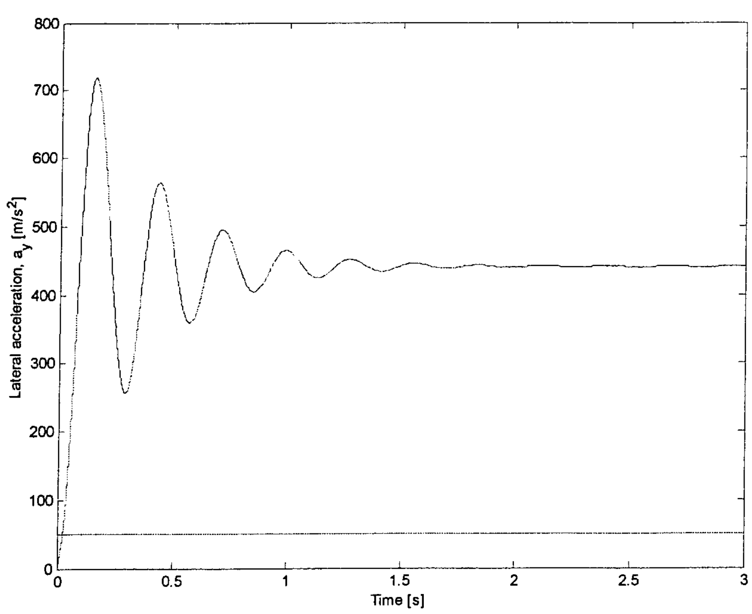

Figure 2.5 Open loop simulation: Lateral autopilot response to a step demand acceleration

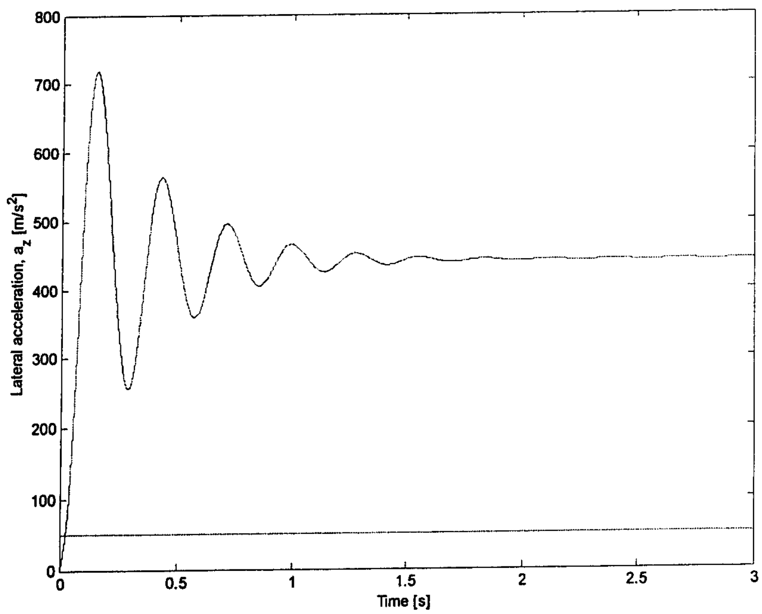

Figure 2.6 Open loop simulation: Lateral autopilot response to a step demand acceleration.

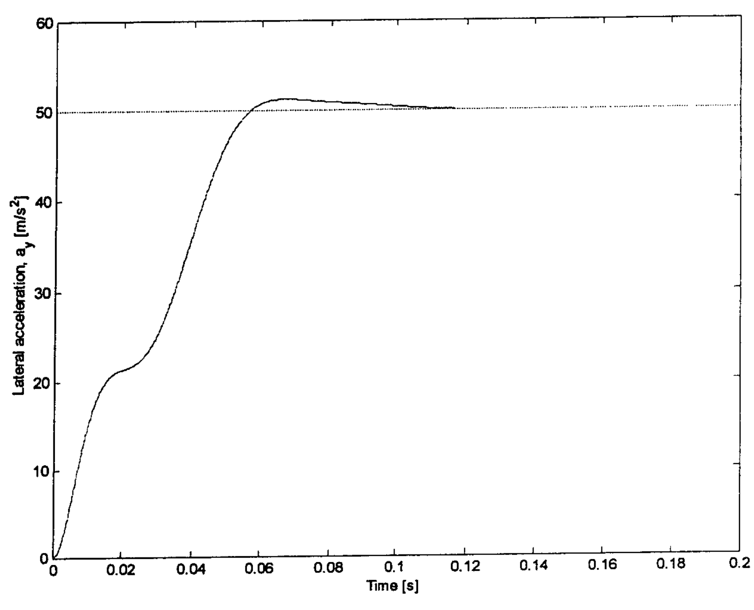

Figure 2.7 Closed loop simulation: Lateral autopilot response to a step demand acceleration.

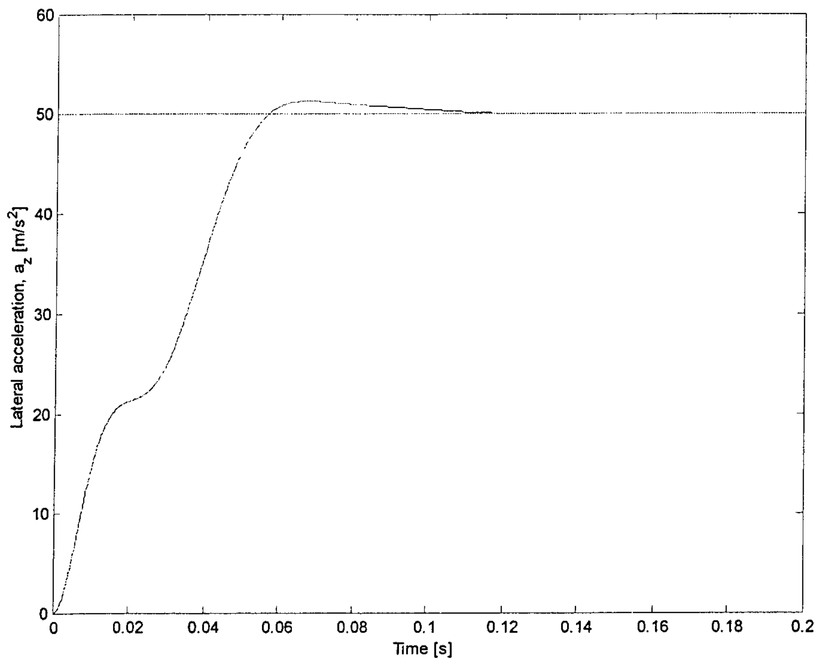

Figure 2.8 Closed loop simulation: Lateral autopilot response to a step demand acceleration.

4. Conclusions

Both the non-linear and linearised autopilot models have been derived in this report. The state-space model of a missile autopilot was validated by comparing the model with the other published results, and through both open and closed-loop systems simulation. The non-linear dynamics model presented as structural quadratic algebraic system is novel and will be used for developed non-linear control techniques suitable for missile systems high g- manoeuvres and operating of a range of aerodynamics conditions. The models developed in this report are useful for further research on precision optimum guidance and control. It is hoped that the higher order model with motion and inertial coupling will provide more accurate representation of missile autopilot dynamics and should be used for adaptive and integrated guidance and control of agile missiles.

5. References

1. Babister, A.W., Aircraft Dynamic Stability and Response, Pergamon, 1980.
2. Blakelock, J.H., *Automatic Control of Aircraft and Missiles,* John Wiley & Sons, Inc., 1965.
3. Cook, M.V., *Flight Dynamics Principles,* Arnold, 1997.

4. Faruqi, F.A., *On the Algebraic Structure of a Class of Nonlinear Dynamical Systems*, DSTO report (to be sent for publication).
5. Garnell, P. and East, D.J., *Guided Weapon Control Systems*, Pergamon Press, 1977.
6. Kuo, B.C., Automatic Control Systems, 6th Edition, Prentice-Hall International Editions, 1991.
7. Pallett, E.H.J., *Automatic Flight Control*, 3rd Edition, BSP Professional Books, 1987.

Appendix A: State Equation

A.1. Quadratic State Vector

Since the state equations (Equations (2.7) and (2.9)) include the quadratic terms, which are comprised of all the combinations of state variables, we define two separate state vectors, one linear state vector, and one quadratic state vector.

Let define the linear-state vector as:

$$\underline{x}_1^{[1]} = [u \quad v \quad w]^T, \tag{A1.1}$$

$$\underline{x}_2^{[1]} = [p \quad q \quad r]^T, \tag{A1.2}$$

$$\underline{x}^{[1]} = \left[\underline{x}_1^{[1]} \,\vdots\, \underline{x}_2^{[1]}\right]^T = [u \quad v \quad w \,\vdots\, p \quad q \quad r]^T, \tag{A1.3}$$

We shall consider the quadratic-state vector $\underline{x}^{[2]}$corresponding to $\underline{x}^{[1]}$. The quadratic-state vector will be defined as a vector whose elements are components of a homogeneous quadratic polynomial of these states taken in the same lexicographic order. That is, the quadratic-state vector may be written as:

$$\underline{x}^{[2]} = \left[u^2 uv\ uw\ up\ uq\ ur \,\vdots\, v^2 vw\ vp\ vq\ vr \,\vdots\, w^2 wp\ wq\ wr \,\vdots\, p^2 pq\ pr \,\vdots\, q^2 qr \,\vdots\, r^2\right]^T \tag{A1.4}$$

There are 21 terms in this quadratic state vector. Note that the dimension of the quadratic state vector is $\dfrac{n(n+1)}{2}$ when the dimension of the linear-state vector is n. This type of representation has been used by other authors [3, 4] when describing high order state combinations of dynamical systems. In the rigid body dynamic equations (2.7) and (2.9), coefficients of a number of these terms are zero. For the sake of simplicity (to avoid setting large number elements in the matrices to zero), only those quadratic states that are associated with non-zero terms are retained. That is, the quadratic-state vector $\underline{x}^{[2]}$and its partitioned form may be written as:

$$\underline{x}^{[2]} = \left[\underline{x}_1^{[2]} \;\vdots\; \underline{x}_2^{[2]}\right] \triangleq \left[uq \quad ur \quad vp \quad vr \quad wp \quad wq \;\vdots\; p^2 \quad pq \quad pr \quad q^2 \quad qr \quad r^2\right]^T \quad \text{(A1.5)}$$

A.2. Linearisation of the Quadratic Vector

Given the quadratic-state vector (A1.4), the first-order locally linearised vector is given by:

$$\underline{\Delta x}^{[2]} = \left[\underline{\Delta x}_1^{[2]} \quad \underline{\Delta x}_2^{[2]}\right] \triangleq \begin{bmatrix} (u_0\Delta q + q_0\Delta u) \\ (u_0\Delta r + r_0\Delta u) \\ (v_0\Delta p + p_0\Delta v) \\ (v_0\Delta r + r_0\Delta v) \\ (w_0\Delta p + p_0\Delta w) \\ (w_0\Delta q + q_0\Delta w) \\ 2p_0\Delta p \\ (p_0\Delta q + q_0\Delta p) \\ (p_0\Delta r + r_0\Delta p) \\ 2q_0\Delta q \\ (q_0\Delta r + r_0\Delta q) \\ 2q_0\Delta q \end{bmatrix} = \begin{bmatrix} q_0 & 0 & 0 & 0 & u_0 & 0 \\ r_0 & 0 & 0 & 0 & 0 & u_0 \\ 0 & p_0 & 0 & v_0 & 0 & 0 \\ 0 & r_0 & 0 & 0 & 0 & v_0 \\ 0 & 0 & p_0 & w_0 & 0 & 0 \\ 0 & 0 & q_0 & 0 & w_0 & 0 \\ 0 & 0 & 0 & 2p_0 & 0 & 0 \\ 0 & 0 & 0 & q_0 & p_0 & 0 \\ 0 & 0 & 0 & r_0 & 0 & p_0 \\ 0 & 0 & 0 & 0 & 2q_0 & 0 \\ 0 & 0 & 0 & 0 & r_0 & q_0 \\ 0 & 0 & 0 & 0 & 0 & 2r_0 \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta v \\ \Delta w \\ \Delta p \\ \Delta q \\ \Delta r \end{bmatrix} \quad \text{(A2.1)}$$

Note that (u_0, v_0, w_0, p_0, q_0, r_0) are local operating states.

We shall write this relationship in a compact form as:

$$\underline{\Delta x}^{[2]} = \left[\underline{\Delta x}_1^{[2]} \quad \underline{\Delta x}_2^{[2]}\right] = \left[X_0\right] \underline{\Delta x}^{[1]} \quad \text{(A2.2)}$$

The matrix $\left[X_0\right]$ is defined via the equation (A2.1).

A.3. Calculation of Inverse Matrix

Given a matrix $[G]$ defined as:

$$[G] = \begin{bmatrix} A & -F & -E \\ -F & B & -D \\ -E & -D & C \end{bmatrix}$$

Its inverse $[G]^{-1}$ is given by:

$$[G]^{-1} = \frac{1}{\Delta}\begin{bmatrix} (BC - D^2) & (CF + DE) & (DF + BE) \\ (CF + DE) & (AC - E^2) & (AD + EF) \\ (DF + BE) & (AD + EF) & (AB - F^2) \end{bmatrix},$$

where $\Delta = (ABC - AD^2 - BE^2 - CF^2 - 2DEF)$.

Appendix B: From State-Space Form into Transfer-function

Consider a system given as:

$$\dot{x}(t) = Ax(t) + Bu(t), \tag{B.1}$$

and

$$y(t) = Cx(t) + Du(t). \tag{B.2}$$

Laplace transformation of Equations (B.1) and (B.2) yields:

$$sX(s) = AX(s) + BU(s), \tag{B.3}$$

and

$$Y(s) = CX(s) + DU(s). \tag{B.4}$$

Equation (B.3) can be rearranged as:

$$X(s) = (sI - A)^{-1}BU(s). \tag{B.5}$$

Substituting (B.5) into (B.4), we obtain:

$$Y(s) = [C(sI - A)^{-1}B + D]U(s) = H(s)U(s). \tag{B.6}$$

Hence, the transfer-function of the system is:

$$H(s) = \frac{Y(s)}{U(s)} = C(sI - A)^{-1}B + D. \tag{B.7}$$

Appendix C: Matlab Codes

C.1. Converting the state-space model into the transfer- functions

The following m.file converts the state-space model into transfer-functions

```
clear all;

% Define the deravatives and parameters
syms Xu Xp Xxi f;
syms Yv Yr Yxi Yzeta f;
syms Zw Zq Zeta f;
syms Lu Lp Lxi f;
syms Mw Mq Meta f;
syms Nv Nr Nxi Neta Nzeta f;
syms u v w p q r f;
syms xi eta zeta xi_dot eta_dot zeta_dot f;
syms xi_d eta_d zeta_d f;
syms ks ws mus f;
syms ax ay az f;
syms dx u0 f;
syms s;

% Define the elements of the matrix A
a11=Xu; a12=0; a13=0; a14=Xp; a15=0; a16=0;
a21=0; a22=Yv; a23=0; a24=0; a25=0; a26=-u0+Yr;
a31=0; a32=0; a33=Zw; a34=0; a35=u0+Zq; a36=0;
a41=Lu; a42=0; a43=0; a44=Lp; a45=0; a46=0;
a51=0; a52=0; a53=Mw; a54=0; a55=Mq; a56=0;
a61=0; a62=Nv; a63=0; a64=0; a65=0; a66=Nr;

% Define the elements of the matrix B
b11=Xxi; b12=0; b13=0;
b21=Yxi; b22=0; b23=Yzeta;
b31=0; b32=Zeta; b33=0;
b41=Lxi; b42=0; b43=0;
b51=0; b52=Meta; b53=0;
b61=Nxi; b62=Neta; b63=Nzeta;

% Define the elements of the matrix C
c11=0; c12=0; c13=0; c14=1; c15=0; c16=0;
c21=0; c22=0; c23=0; c24=0; c25=1; c26=0;
c31=0; c32=0; c33=0; c34=0; c35=0; c36=1;
c41=Xu; c42=0; c43=0; c44=Xp; c45=0; c46=0;
c51=0; c52=Yv+Nv*dx; c53=0; c54=0; c55=0; c56=Yr+Nr*dx;
c61=0; c62=0; c63=Zw+Mw*dx; c64=0; c65=Zq-Mq*dx; c66=Yr;
```

```
% Define the elements of the matrix D
d11=0; d12=0; d13=0;
d21=0; d22=0; d23=0;
d31=0; d32=0; d33=0;
d41=Xxi; d42=0; d43=0;
d51=Yxi+Nxi*dx; d52=0; d53=Yzeta+Nzeta*dx;
d61=0; d62=Zeta-Meta*dx; d63=0;

A=[a11,a12,a13,a14,a15,a16;
   a21,a22,a23,a24,a25,a26;
   a31,a32,a33,a34,a35,a36;
   a41,a42,a43,a44,a45,a46;
   a51,a52,a53,a54,a55,a56;
   a61,a62,a63,a64,a65,a66];

B=[b11,b12,b13;
   b21,b22,b23;
   b31,b32,b33;
   b41,b42,b43;
   b51,b52,b53;
   b61,b62,b63];


C=[c11,c12,c13,c14,c15,c16;
   c21,c22,c23,c24,c25,c26;
   c31,c32,c33,c34,c35,c36;
   c41,c42,c43,c44,c45,c46;
   c51,c52,c53,c54,c55,c56;
   c61,c62,c63,c64,c65,c66];

D=[d11,d12,d13;
   d21,d22,d23;
   d31,d32,d33;
   d41,d42,d43;
   d51,d52,d53;
   d61,d62,d63];

U=[xi;eta;zeta];

S=[s,0,0,0,0,0;
   0,s,0,0,0,0;
   0,0,s,0,0,0;
   0,0,0,s,0,0;
   0,0,0,0,s,0;
   0,0,0,0,0,s];

% The transfer-function of the system is:
```

% $H(s) = \frac{y(s)}{u(s)} = C(SI - A)^{-1}B + D$

```
% Calculate (SI − A)^-1
SIAinv= inv(S-A);

% Calculate the transfer-function
H=C*SIAinv*B +D;

% Display the transfer-function between the lateral
% acceleration, a_y, and the rudder deflection,ζ.
H(5,3)
```

C.1. Open-loop and closed-loop simulation

```
%**********************************************************
%This main program simulates the response of the missile
%for a step input at the elevator and rudder.

clear all;

% Call the M-file called values.m
values;

% Call the M-file called init.m
init;

% Call the M-file called ssmodel.m
ssmodel;
for t=0:ts:5
   REF=[theta_d;az_d;ay_d];
   U=REF - K*Y;
   X_dot_prev=X_dot;
   X_dot=A*X+B*U;
   X= X+(X_dot_prev + X_dot)/2*ts;
   Y=C*X;
   accy=[accy,Y(5,1)];
   accz=[accz,Y(6,1)];
   time = [time,t];
   ay_d=50;
   az_d= 50;
   theta_d = 0;
end;
plot(time,acc1,time,acc3);
xlabel('Time [s]');
ylabel('Lateral acceleration, a_y [m/s^2]');
title('Closed loop response for a step input of 50 m/s^2 at the
rudder');
figure;
plot(time,acc2,time,acc3);
xlabel('Time [s]');
```

```
ylabel('Lateral acceleration, a_z [m/s^2]');
title('Closed loop response for a step input of 50 m/s^2 at the
elevator');

%***************** Values.m *********************************
% Set values of the parameters
Xu=0.0; Xp=0.0; Xxi=0.0; Xeta=0.0; Xzeta=0.0;

Yv=-3; Yp=0; Yr=0; Yxi=0.0; Yeta=0; Yzeta=180;

Lu=0.0; Lp=0.0; Lxi=0.0;

Mu=0.0; Mv=0.0; Mw=-1.0; Mq=-3.0; Meta=-500.0; Mzeta=0.0;

Nv= 1; Np=0.0; Nw=0.0; Nr=-3; Nxi=0.0; Neta=0.0;
Nzeta=-500.0;

Zw=-3; Zq=Yr; Zeta =-Yzeta;

ks=0.0068; ksz=ks
mus=0.7; musz=mus;
ws=180;
dx =0.5;
u0=500;

% For the case of open-loop simulation, Kgr=Kay=Kgq=Kaz=0
Kgr= 30.75; Kay = 0.825;
Kgq=-Kgr; Kaz = Kay;

%************************ init.m *************************
%Initialise the parameters
u=0; v=0; w=0; p=0; q=0; r=0;
xi=0; eta=0; zeta=0;
xi_dot=0; eta_dot=0; zeta_dot=0;
X=[u;v;w;p;q;r;xi;eta;zeta;xi_dot;eta_dot;zeta_dot];
X_dot=X;
Y=[0;0;0;0;0;0];
i = 1;
ay_d=0;
az_d=0;
theta_d=0;
xi_d(i)=0;
eta_d(i)=0;
zeta_d(i)=0;

ts=0.001;
accy=0;
accz=0;
time=0;
```

```
% ********************* ssmodel.m **********************
% Define the state-space model
a1_1=Xu;a1_2=0;a1_3=0;a1_4=Xp;a1_5=0;a1_6=0;
a1_7=Xxi;a1_8=0;a1_9=0;a1_10=0;a1_11=0;a1_12=0;

a2_1=0;a2_2=Yv;a2_3=0;a2_4=0;a2_5=0;a2_6=-u0+Yr;
a2_7=Yxi;a2_8=0;a2_9=Yzeta;a2_10=0;a2_11=0;a2_12=0

a3_1=0;a3_2=0;a3_3=Zw;a3_4=0;a3_5=u0+Zq;a3_6=0;
a3_7=0;a3_8=Zeta;a3_9=0;a3_10=0;a3_11=0;a3_12=0;

a4_1=Lu;a4_2=0;a4_3=0;a4_4=Lp;a4_5=0;a4_6=0;
a4_7=Lxi;a4_8=0;a4_9=0;a4_10=0;a4_11=0;a4_12=0;

a5_1=0;a5_2=0;a5_3=Mw;a5_4=0;a5_5=Mq;a5_6=0;
a5_7=0;a5_8=Meta;a5_9=0;a5_10=0;a5_11=0;a5_12=0;

a6_1=0;a6_2=Nv;a6_3=0;a6_4=0;a6_5=0;a6_6=Nr;
a6_7=Nxi;a6_8=Neta;a6_9=Nzeta;a6_10=0;a6_11=0;a6_12=0;

a7_1=0;a7_2=0;a7_3=0;a7_4=0;a7_5=0;a7_6=0;
a7_7=0;a7_8=0;a7_9=0;a7_10=1;a7_11=0;a7_12=0;

a8_1=0;a8_2=0;a8_3=0;a8_4=0;a8_5=0;a8_6=0;
a8_7=0;a8_8=0;a8_9=0;a8_10=0;a8_11=1;a8_12=0;

a9_1=0;a9_2=0;a9_3=0;a9_4=0;a9_5=0;a9_6=0;
a9_7=0;a9_8=0;a9_9=0;a9_10=0;a9_11=0;a9_12=1;

a10_1=0;a10_2=0;a10_3=0;a10_4=0;a10_5=0;a10_6=0;
a10_7=-ws^2;a10_8=0;a10_9=0;a10_10=-2*mus*ws;a10_11=0;
a10_12=0;

a11_1=0;a11_2=0;a11_3=0;a11_4=0;a11_5=0;a11_6=0;
a11_7=0;a11_8=-ws^2;a11_9=0;a11_10=0;a11_11=-2*musz*ws;
a11_12=0;

a12_1=0;a12_2=0;a12_3=0;a12_4=0;a12_5=0;a12_6=0;
a12_7=0;a12_8=0;a12_9=-ws^2;a12_10=0;a12_11=0;
a12_12=-2*mus*ws;

b1_1=0;b1_2=0;b1_3=0;
b2_1=0;b2_2=0;b2_3=0;
b3_1=0;b3_2=0;b3_3=0;
b4_1=0;b4_2=0;b4_3=0;
b5_1=0;b5_2=0;b5_3=0;
b6_1=0;b6_2=0;b6_3=0;
b7_1=0;b7_2=0;b7_3=0;
b8_1=0;b8_2=0;b8_3=0;
b9_1=0;b9_2=0;b9_3=0;
```

```
b10_1=-ks*ws^2;b10_2=0;b10_3=0;
b11_1=0;b11_2=-ksz*ws^2;b11_3=0;
b12_1=0;b12_2=0;b12_3=-ks*ws^2;

c1_1=0;c1_2=0;c1_3=0;c1_4=1;c1_5=0;c1_6=0;
c1_7=0;c1_8=0;c1_9=0;c1_10=0;c1_11=0;c1_12=0;

c2_1=0;c2_2=0;c2_3=0;c2_4=0;c2_5=1;c2_6=0;
c2_7=0;c2_8=0;c2_9=0;c2_10=0;c2_11=0;c2_12=0;

c3_1=0;c3_2=0;c3_3=0;c3_4=0;c3_5=0;c3_6=1;
c3_7=0;c3_8=0;c3_9=0;c3_10=0;c3_11=0;c3_12=0;

c4_1=Xu;c4_2=0;c4_3=0;c4_4=Xp;c4_5=0;c4_6=0;
c4_7=Xxi;c4_8=Xeta;c4_9=Xzeta;c4_10=0;c4_11=0;c4_12=0;

c5_1=0;c5_2=Yv+Nv*dx;c5_3=0;c5_4=0;c5_5=0;c5_6=Yr+Nr*dx;
c5_7=Yxi+Nxi*dx;c5_8=Yeta+Neta*dx;c5_9=Yzeta+Nzeta*dx;
c5_10=0;c5_11=0;c5_12=0;

c6_1=0;c6_2=0;c6_3=Zw-Mw*dx;c6_4=0;c6_5=Zq-Mq*dx;c6_6=0;
c6_7=0;c6_8=Zeta-Meta*dx;c6_9=0;c6_10=0;c6_11=0;c6_12=0;

A=[a1_1,a1_2,a1_3,a1_4,a1_5,a1_6,a1_7,a1_8,a1_9,a1_10,a1_11,a1_12;
   a2_1,a2_2,a2_3,a2_4,a2_5,a2_6,a2_7,a2_8,a2_9,a2_10,a2_11,a2_12;
   a3_1,a3_2,a3_3,a3_4,a3_5,a3_6,a3_7,a3_8,a3_9,a3_10,a3_11,a3_12;
   a4_1,a4_2,a4_3,a4_4,a4_5,a4_6,a4_7,a4_8,a4_9,a4_10,a4_11,a4_12;
   a5_1,a5_2,a5_3,a5_4,a5_5,a5_6,a5_7,a5_8,a5_9,a5_10,a5_11,a5_12;
   a6_1,a6_2,a6_3,a6_4,a6_5,a6_6,a6_7,a6_8,a6_9,a6_10,a6_11,a6_12;
   a7_1,a7_2,a7_3,a7_4,a7_5,a7_6,a7_7,a7_8,a7_9,a7_10,a7_11,a7_12;
   a8_1,a8_2,a8_3,a8_4,a8_5,a8_6,a8_7,a8_8,a8_9,a8_10,a8_11,a8_12;
 a9_1,a9_2,a9_3,a9_4,a9_5,a9_6,a9_7,a9_8,a9_9,a9_10,a9_11,a9_12;
        a10_1,a10_2,a10_3,a10_4,a10_5,a10_6,a10_7,a10_8,a10_9,a10_10,a10_11,
a10_12;  a11_1,a11_2,a11_3,a11_4,a11_5,a11_6,a11_7,a11_8,a11_9,a11_10,a11_11,
a11_12;  a12_1,a12_2,a12_3,a12_4,a12_5,a12_6,a12_7,a12_8,a12_9,a12_10,a12_11,
a12_12];

B=[b1_1,b1_2,b1_3;
   b2_1,b2_2,b2_3;
   b3_1,b3_2,b3_3;
   b4_1,b4_2,b4_3;
   b5_1,b5_2,b5_3;
   b6_1,b6_2,b6_3;
   b7_1,b7_2,b7_3;
   b8_1,b8_2,b8_3;
   b9_1,b9_2,b9_3;
   b10_1,b10_2,b10_3;
   b11_1,b11_2,b11_3;
   b12_1,b12_2,b12_3];
```

```
C=[c1_1,c1_2,c1_3,c1_4,c1_5,c1_6,c1_7,c1_8,c1_9,c1_10,c1_11,c1_12;
   c2_1,c2_2,c2_3,c2_4,c2_5,c2_6,c2_7,c2_8,c2_9,c2_10,c2_11,c2_12;
   c3_1,c3_2,c3_3,c3_4,c3_5,c3_6,c3_7,c3_8,c3_9,c3_10,c3_11,c3_12;
   c4_1,c4_2,c4_3,c4_4,c4_5,c4_6,c4_7,c4_8,c4_9,c4_10,c4_11,c4_12;
   c5_1,c5_2,c5_3,c5_4,c5_5,c5_6,c5_7,c5_8,c5_9,c5_10,c5_11,c5_12;
   c6_1,c6_2,c6_3,c6_4,c6_5,c6_6,c6_7,c6_8,c6_9,c6_10,c6_11,c6_12];


D= [0,0,0;
    0,0,0;
    0,0,0;
    0,0,0;
    0,0,0;
    0,0,0];

K = [0,0,0,0,0,0;
     0,Kgq,0,0,0,Kaz;
     0,0,Kgr,0,Kay,0];
```

Appendix D: Values of the Non-Zero Derivatives and Parameters used in the simulation

$\widetilde{Y}_v = -3, \widetilde{Y}_\zeta = 180, \widetilde{N}_v = 1, \widetilde{N}_r = -3, \widetilde{N}_\zeta = -500,$

$\widetilde{Z}_w = -3, \widetilde{Z}_\eta = -180, \widetilde{M}_w = -1, \widetilde{M}_q = -3,, \widetilde{M}_\eta = -500,$

$k_{s\xi} = 0.0068, k_{s\eta} = 0.0068, k_{s\zeta} = 0.0068,$

$\mu_{s\xi} = 0.7, \mu_{s\eta} = 0.7, \mu_{s\zeta} = 0.7,$

$\omega_{s\xi} = 180, \omega_{s\eta} = 180, \omega_{s\zeta} = 180,$

$u_0 = 500, dx = 0.5$

Feedback gains:

Kgr= 30.75; Kay = 0.825;
Kgq=-Kgr; Kaz = Kay;

DISTRIBUTION LIST

Mathematical Models for a Missile Autopilot Design

Farhan A. Faruqi and Thanh Lan Vu

AUSTRALIA

DEFENCE ORGANISATION

S&T Program
Chief Defence Scientist
FAS Science Policy
AS Science Corporate Management
Director General Science Policy Development
shared copy
Counsellor Defence Science, London (Doc Data Sheet)
Counsellor Defence Science, Washington (Doc Data Sheet)
Scientific Adviser to MRDC Thailand (Doc Data Sheet)
Scientific Adviser Joint (Doc Data Sheet and distribution list only)
Navy Scientific Adviser (Doc Data Sheet and distribution list only)

Scientific Adviser - Army (Doc Data Sheet and distribution list only)

Air Force Scientific Adviser
Director Trials

Systems Sciences Laboratory
Chief of Weapon Systems Division
RLAWS
RLMWS
RLLWS
RLEWT
Guidance and Control Group, WSD (6 Copies)
Farhan A. Faruqi
Thanh Lan Vu

DSTO Library and Archives
Library Edinburgh 2 copies
Australian Archives

Capability Systems Staff
Director General Maritime Development (Doc Data Sheet only)
Director General Aerospace Development (Doc Data Sheet only)

Knowledge Staff
Director General Command, Control, Communications and Computers (DGC4)
(Doc Data Sheet only)

Army

ABCA National Standardisation Officer, Land Warfare Development Sector, Puckapunyal (4 copies)

Intelligence Program

DGSTA Defence Intelligence Organisation
Manager, Information Centre, Defence Intelligence Organisation

Defence Libraries

Library Manager, DLS-Canberra
Library Manager, DLS - Sydney West (Doc Data Sheet Only)

UNIVERSITIES AND COLLEGES

Australian Defence Force Academy
Library
Head of Aerospace and Mechanical Engineering
Hargrave Library, Monash University (Doc Data Sheet only)
Librarian, Flinders University

OTHER ORGANISATIONS

National Library of Australia
NASA (Canberra)
AusInfo

OUTSIDE AUSTRALIA

INTERNATIONAL DEFENCE INFORMATION CENTRES

US Defense Technical Information Center, 2 copies
UK Defence Research Information Centre, 2 copies
Canada Defence Scientific Information Service, 1 copy
NZ Defence Information Centre, 1 copy

ABSTRACTING AND INFORMATION ORGANISATIONS

Library, Chemical Abstracts Reference Service
Engineering Societies Library, US
Materials Information, Cambridge Scientific Abstracts, US
Documents Librarian, The Center for Research Libraries, US

INFORMATION EXCHANGE AGREEMENT PARTNERS

Acquisitions Unit, Science Reference and Information Service, UK
Library - Exchange Desk, National Institute of Standards and Technology, US
National Aerospace Laboratory, Japan
National Aerospace Laboratory, Netherlands

SPARES (5 copies)

Total number of copies: **51**

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION
DOCUMENT CONTROL DATA

1. PRIVACY MARKING/CAVEAT (OF DOCUMENT)

2. TITLE Mathematical Models for a Missile Autopilot Design	3. SECURITY CLASSIFICATION (FOR UNCLASSIFIED REPORTS THAT ARE LIMITED RELEASE USE (L) NEXT TO DOCUMENT CLASSIFICATION) Document (U) Title (U) Abstract (U)
4. AUTHOR(S) Farhan A. Faruqi and Thanh Lan Vu	5. CORPORATE AUTHOR Systems Sciences Laboratory PO Box 1500 Edinburgh South Australia 5111 Australia

6a. DSTO NUMBER	6b. AR NUMBER	6c. TYPE OF REPORT	7. DOCUMENT DATE
DSTO-TN-0449	AR- 012-430	Technical Note	August, 2002

8. FILE NUMBER	9. TASK NUMBER	10. TASK SPONSOR	11. NO. OF PAGES	12. NO. OF REFERENCES
J9505-21-207	01/219	DST	31	7

13. URL on the World Wide Web	14. RELEASE AUTHORITY
http://www.dsto.defence.gov.au/corporate/reports/DSTO-TN-0449.pdf	Chief, Weapons Systems Division

15. SECONDARY RELEASE STATEMENT OF THIS DOCUMENT

Approved for public release

OVERSEAS ENQUIRIES OUTSIDE STATED LIMITATIONS SHOULD BE REFERRED THROUGH DOCUMENT EXCHANGE, PO BOX 1500, EDINBURGH, SA 5111

16. DELIBERATE ANNOUNCEMENT

No Limitations

17. CITATION IN OTHER DOCUMENTS Yes

18. DEFTEST DESCRIPTORS

Mathematical models, Automatic pilots, Missile guidance, Design criteria

19. ABSTRACT
This report considers the derivation of the mathematical model for a missile autopilot in state space form. The basic equations defining the airframe dynamics are non-linear, however, since the non-linearities are "structured" (in the sense that the states are of quadratic form) a novel approach of expressing this non-linear dynamics in state space form is given. This should provide a useful way to implement the equations in a computer simulation program and possibly for future application of non-linear analysis and synthesis techniques, particularly for autopilot design of missiles executing high g-manoeuvres.

This report also considers a locally linearised state space model that lends itself to better known linear techniques of the modern control theory. A coupled multi-input multi-output (MIMO) model is derived suitable for both the application of the modern control techniques as well as the classical time-domain and frequency domain techniques. This is validated by comparing the model with the other published results, and through both open and closed-loop systems simulations. The models developed are useful for further research on precision optimum guidance and control. It is hoped that the model will provide more accurate presentations of missile autopilot dynamics and will be used for adaptive and integrated guidance & control of agile missiles.

TECHNICAL NOTE DSTO-TN-0449 AR-012-430 AUGUST 2002

DEFENCE
SCIENCE & TECHNOLOGY

SYSTEMS SCIENCES LABORATORY
PO BOX 1500 EDINBURGH SOUTH AUSTRALIA, 5111 AUSTRALIA
TELEPHONE (08) 8259 5555