01 Jan 2003

Missile Longitudinal Autopilot Design Using a New Suboptimal Nonlinear Control Method

Ming Xin

S. N. Balakrishnan
Missouri University of Science and Technology, bala@mst.edu

Recommended Citation

M. Xin and S. N. Balakrishnan, "Missile Longitudinal Autopilot Design Using a New Suboptimal Nonlinear Control Method," *IEE Proceedings - Control Theory and Applications*, Institute of Electrical and Electronics Engineers (IEEE), Jan 2003.
The definitive version is available at https://doi.org/10.1049/ip-cta:20030966

Missile longitudinal autopilot design using a new suboptimal nonlinear control method

M. Xin and S.N. Balakrishnan

Abstract: A missile longitudinal autopilot is designed using a new nonlinear control synthesis technique called the θ–D approximation. The particular θ–D methodology used is referred to as the θ–D H_2 design. The technique can achieve suboptimal closed-form solutions to a class of nonlinear optimal control problems in the sense that it solves the Hamilton–Jacobi–Bellman equation approximately by adding perturbations to the cost function. An interesting feature of this method is that the expansion terms in the expression for suboptimal control are nothing but solutions to the state-dependent Riccati equations associated with this class of problems. The θ–D H_2 design has the same structure as that of the linear H_2 formulation except that the two Riccati equations are state dependent. Numerical simulations are presented that demonstrate the potential of this technique for use in an autopilot design. These results are compared with the recently popular SDRE H_2 method.

1 Introduction

Modern aircraft or missiles often operate in flight regimes where nonlinearities significantly affect dynamic response. For example, a high-performance missile must be quickly responsive to and follow accurately any guidance commands, so that it can intercept fast moving and agile targets. Many nonlinear control methods have been proposed for the missile autopilot design. One popular method of formulation has been the optimal control of nonlinear dynamics with respect to a mathematical index of performance [1]. A major difficulty in this line of approach is finding solutions to the resulting Hamilton–Jacobi–Bellman (HJB) equation.

Suboptimal solutions are found by power series expansion methods. Wernli and Cook [2] developed an approach by bringing the original system into an apparent linear form. Their suboptimal control involves finding the Taylor expansion of the solution to a state-dependent Riccati equation. But the convergence of this series is not guaranteed and the resulting control law leads to large control efforts when the initial states are large. Garrard [3, 4] formulated another approach that expanded both the optimal cost and the nonlinear dynamics as power series of the states and employed it in the high-angle-of-attack maneuverable aircraft. However, this method has to assume the structure of the optimal cost as a scalar polynomial with undetermined coefficients which contains all possible combinations of products of the elements of the state vector. As the system order increases, the complexity of determining these coefficients increases dramatically. The common problem with these methods is that they do not offer a way to ensure that the system is asymptotically stable in the large. Beard *et al.* [5] adopted the Galerkin approximation to solve the HJB equation. It was used to synthesise a nonlinear optimal control for a missile autopilot system [6]. The control laws are given as a series of basis functions. To find an admissible control to satisfy all the ten conditions proposed in that paper is not an easy task.

Another recently emerging technique that systematically solves the nonlinear regulator problem is the state-dependent Riccati equation (SDRE) method [7]. By turning the equations of motion into a linear-like structure, this approach permits the designer to employ linear optimal control methods such as the LQR methodology and the H_∞ design technique for the synthesis of nonlinear control systems. The SDRE method however, needs online computations of the algebraic Riccati equation at each sample time.

A new suboptimal nonlinear controller synthesis (θ–D approximation) technique based on approximate solution to the Hamilton–Jacobi–Bellman equation is proposed in this paper. By introducing an artificial variable θ, the costate λ can be expanded as a power series in terms of θ. This technique can overcome the problem of large-control-for-large-initial-states encountered by using the control law in [2]. By adjusting some perturbation parameters in the cost function, we are also able to modulate the transient performance of the system.

We extend the standard linear H_2 optimal control method to nonlinear problems using the θ–D technique. The linear H_2 control problem has been studied and implemented since 1960s [8]. It is used to find a proper controller that stabilises the system internally and minimise the H_2 norm of the transfer function from the exogenous input to the performance output. With output feedback, the H_2 design ends up with having to solve two Riccati equations. Some studies examined the use of H_2 controller design in nonlinear systems. In [9], the SDRE H_2 method was used to design a full-envelope pitch autopilot. However, solving two Riccati equations online is very timeconsuming. In this paper, θ–D H_2 design is proposed for the same problem as that in [9] that gives an approximate closed-form solution to the two state-dependent Riccati equations.

IEE Proceedings online no. 20030966

doi: 10.1049/ip-cta:20030966

Paper first received 23rd October 2002 and in revised form 29th July 2003

The authors are with the Department of Mechanical and Aerospace Engineering and Engineering Mechanics, University of Missouri-Rolla, Rolla, MO 65401, USA

2 Suboptimal control of a class of nonlinear systems

We consider systems described by

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{B}(\boldsymbol{x})\boldsymbol{u} \tag{1}$$

The problem is to find the control $\boldsymbol{u}(t)$ which minimises the cost J given by

$$J = \frac{1}{2}\int_0^\infty (\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{u}^T\boldsymbol{R}\boldsymbol{u})dt \tag{2}$$

where $\boldsymbol{x} \in \Omega \subset \boldsymbol{R}^n, \boldsymbol{f}:\Omega \to \boldsymbol{R}^n, \boldsymbol{B} \in \boldsymbol{R}^{n\times m}, \boldsymbol{u}:\Omega \to \boldsymbol{R}^m, \boldsymbol{Q} \in \boldsymbol{R}^{n\times n}$, $\boldsymbol{R} \in \boldsymbol{R}^{m\times m}$; Ω is a compact set in $\boldsymbol{R}^n$; $\boldsymbol{Q}$ is positive semidefinite matrix and $\boldsymbol{R}$ is positive definite matrix; $\boldsymbol{f}(0) = 0$. The solution to (2) is obtained by solving the Hamilton–Jacobi–Bellman partial differential equation [1]

$$\frac{\partial V^T}{\partial \boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x}) - \frac{1}{2}\frac{\partial V^T}{\partial \boldsymbol{x}}\boldsymbol{B}(\boldsymbol{x})\boldsymbol{R}^{-1}\boldsymbol{B}^T(\boldsymbol{x})\frac{\partial V}{\partial \boldsymbol{x}} + \frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} = 0 \tag{3}$$

with $V(0) = 0$. The optimal control is given by

$$\boldsymbol{u} = -\boldsymbol{R}^{-1}\boldsymbol{B}^T(\boldsymbol{x})\frac{\partial V}{\partial \boldsymbol{x}} \tag{4}$$

and $V(\boldsymbol{x})$ is the optimal cost, i.e.

$$V(\boldsymbol{x}) = \min_{\boldsymbol{u}} \frac{1}{2}\int_0^\infty (\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{u}^T\boldsymbol{R}\boldsymbol{u})dt \tag{5}$$

The HJB equation is extremely difficult to solve in general; in this study we find approximate solutions. Add perturbations to the cost function as

$$J = \frac{1}{2}\int_0^\infty \left[\boldsymbol{x}^T\left(\boldsymbol{Q} + \sum_{i=1}^\infty \boldsymbol{D}_i\theta^i\right)\boldsymbol{x} + \boldsymbol{u}^T\boldsymbol{R}\boldsymbol{u}\right]dt \tag{6}$$

where θ and $\boldsymbol{D}_i$ are chosen such that $\boldsymbol{Q} + \sum_{i=1}^\infty \boldsymbol{D}_i\theta^i$ is positive semidefinite. For later use we rewrite the state equation as

$$\begin{aligned}\dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{B}(\boldsymbol{x})\boldsymbol{u}\\ &= \left[\boldsymbol{A}_0 + \theta\left(\frac{\boldsymbol{A}(\boldsymbol{x})}{\theta}\right)\right]\boldsymbol{x} + \left[\boldsymbol{g}_0 + \theta\left(\frac{\boldsymbol{g}(\boldsymbol{x})}{\theta}\right)\right]\boldsymbol{u}\end{aligned} \tag{7}$$

where $\boldsymbol{A}_0$ is a constant matrix such that $(\boldsymbol{A}_0, \boldsymbol{g}_0)$ is a stabilisable pair and $[(\boldsymbol{A}_0 + \boldsymbol{A}(\boldsymbol{x})), (\boldsymbol{g}_0 + \boldsymbol{g}(\boldsymbol{x}))]$ is pointwise controllable. Define

$$\lambda = \frac{\partial V}{\partial \boldsymbol{x}} \tag{8}$$

By using (8) in (3) we have

$$\lambda^T\boldsymbol{f}(\boldsymbol{x}) - \frac{1}{2}\lambda^T\boldsymbol{B}(\boldsymbol{x})\boldsymbol{R}^{-1}\boldsymbol{B}^T(\boldsymbol{x})\lambda + \frac{1}{2}\boldsymbol{x}^T\left(\boldsymbol{Q} + \sum_{i=1}^\infty \boldsymbol{D}_i\theta^i\right)\boldsymbol{x} = 0 \tag{9}$$

Assume a power series expansion of λ as

$$\lambda = \sum_{i=0}^\infty \boldsymbol{T}_i\theta^i\boldsymbol{x} \tag{10}$$

where $\boldsymbol{T}_i$ are to be determined and assumed to be symmetric. By substituting (10) into (3) and equating the coefficients of powers of θ to zero we get

$$\boldsymbol{T}_0\boldsymbol{A}_0 + \boldsymbol{A}_0^T\boldsymbol{T}_0 - \boldsymbol{T}_0\boldsymbol{g}_0\boldsymbol{R}^{-1}\boldsymbol{g}_0^T\boldsymbol{T}_0 + \boldsymbol{Q} = 0 \tag{11}$$

$$\begin{aligned}&\boldsymbol{T}_1(\boldsymbol{A}_0 - \boldsymbol{g}_0\boldsymbol{R}^{-1}\boldsymbol{g}_0^T\boldsymbol{T}_0) + (\boldsymbol{A}_0^T - \boldsymbol{T}_0\boldsymbol{g}_0\boldsymbol{R}^{-1}\boldsymbol{g}_0^T)\boldsymbol{T}_1\\ &\quad = -\frac{\boldsymbol{T}_0\boldsymbol{A}(\boldsymbol{x})}{\theta} - \frac{\boldsymbol{A}^T(\boldsymbol{x})\boldsymbol{T}_0}{\theta} + \boldsymbol{T}_0\boldsymbol{g}_0\boldsymbol{R}^{-1}\frac{\boldsymbol{g}^T}{\theta}\boldsymbol{T}_0\\ &\quad\quad + \boldsymbol{T}_0\frac{\boldsymbol{g}}{\theta}\boldsymbol{R}^{-1}\boldsymbol{g}_0^T\boldsymbol{T}_0 - \boldsymbol{D}_1\end{aligned} \tag{12}$$

$$\begin{aligned}&\boldsymbol{T}_2(\boldsymbol{A}_0 - \boldsymbol{g}_0\boldsymbol{R}^{-1}\boldsymbol{g}_0^T\boldsymbol{T}_0) + (\boldsymbol{A}_0^T - \boldsymbol{T}_0\boldsymbol{g}_0\boldsymbol{R}^{-1}\boldsymbol{g}_0^T)\boldsymbol{T}_2\\ &\quad = -\frac{\boldsymbol{T}_1\boldsymbol{A}(\boldsymbol{x})}{\theta} - \frac{\boldsymbol{A}^T(\boldsymbol{x})\boldsymbol{T}_1}{\theta} + \boldsymbol{T}_0\boldsymbol{g}_0\boldsymbol{R}^{-1}\frac{\boldsymbol{g}^T}{\theta}\boldsymbol{T}_1\\ &\quad\quad + \boldsymbol{T}_0\frac{\boldsymbol{g}}{\theta}\boldsymbol{R}^{-1}\boldsymbol{g}_0^T\boldsymbol{T}_1 + \boldsymbol{T}_0\frac{\boldsymbol{g}}{\theta}\boldsymbol{R}^{-1}\frac{\boldsymbol{g}^T}{\theta}\boldsymbol{T}_0 + \boldsymbol{T}_1\boldsymbol{g}_0\boldsymbol{R}^{-1}\boldsymbol{g}_0^T\boldsymbol{T}_1\\ &\quad\quad + \boldsymbol{T}_1\boldsymbol{g}_0\boldsymbol{R}^{-1}\frac{\boldsymbol{g}^T}{\theta}\boldsymbol{T}_0 + \boldsymbol{T}_1\frac{\boldsymbol{g}}{\theta}\boldsymbol{R}^{-1}\boldsymbol{g}_0^T\boldsymbol{T}_0 - \boldsymbol{D}_2\end{aligned} \tag{13}$$

$$\begin{aligned}&\boldsymbol{T}_n(\boldsymbol{A}_0 - \boldsymbol{g}_0\boldsymbol{R}^{-1}\boldsymbol{g}_0^T\boldsymbol{T}_0) + (\boldsymbol{A}_0^T - \boldsymbol{T}_0\boldsymbol{g}_0\boldsymbol{R}^{-1}\boldsymbol{g}_0^T)\boldsymbol{T}_n\\ &\quad = -\frac{\boldsymbol{T}_{n-1}\boldsymbol{A}(x)}{\theta} - \frac{\boldsymbol{A}^T(\boldsymbol{x})\boldsymbol{T}_{n-1}}{\theta}\\ &\quad\quad + \sum_{j=0}^{n-1}\boldsymbol{T}_j\left(\boldsymbol{g}_0\boldsymbol{R}^{-1}\frac{\boldsymbol{g}^T}{\theta} + \frac{\boldsymbol{g}}{\theta}\boldsymbol{R}^{-1}\boldsymbol{g}_0^T\right)\boldsymbol{T}_{n-1-j}\\ &\quad\quad + \sum_{j=0}^{n-2}\boldsymbol{T}_j\boldsymbol{g}\boldsymbol{R}^{-1}\boldsymbol{g}^T\boldsymbol{T}_{n-2-j} + \sum_{j=1}^{n-1}\boldsymbol{T}_j\boldsymbol{g}_0\boldsymbol{R}^{-1}\boldsymbol{g}_0^T\boldsymbol{T}_{n-j} - \boldsymbol{D}_n\end{aligned} \tag{14}$$

Since the right-hand side of (11)–(14) involve $\boldsymbol{x}$ and θ, $T_i = \boldsymbol{T}_i(\boldsymbol{x}, \theta)$. The expression for control can be obtained in a power series as

$$\boldsymbol{u} = -\boldsymbol{R}^{-1}\boldsymbol{B}^T(\boldsymbol{x})\lambda = -\boldsymbol{R}^{-1}\boldsymbol{B}^T(\boldsymbol{x})\sum_{i=0}^\infty \boldsymbol{T}_i(\boldsymbol{x}, \theta)\theta^i\boldsymbol{x} \tag{15}$$

Note that (11) is an algebraic Riccati equation. The rest of equations are linear Lyapunov equations. In the rest of this paper we call this method the θ–D approximation technique. The algorithm in [2] results in the 'θ-approximation' (without the $\boldsymbol{D}_i$ terms) although through a different approach. A problem with the θ-approximation is that large initial conditions may give rise to large control or even instability. We avoid it by constructing $\boldsymbol{D}_i$ as

$$\boldsymbol{D}_1 = k_1e^{-l_1t}\left[-\frac{\boldsymbol{T}_0\boldsymbol{A}(\boldsymbol{x})}{\theta} - \frac{\boldsymbol{A}^T(\boldsymbol{x})\boldsymbol{T}_0}{\theta}\right] \tag{16}$$

$$\boldsymbol{D}_2 = k_2e^{-l_2t}\left[-\frac{\boldsymbol{T}_1\boldsymbol{A}(\boldsymbol{x})}{\theta} - \frac{\boldsymbol{A}^T(\boldsymbol{x})\boldsymbol{T}_1}{\theta}\right] \tag{17}$$

$$\vdots$$

$$\boldsymbol{D}_n = k_ne^{-l_nt}\left[-\frac{\boldsymbol{T}_{n-1}\boldsymbol{A}(\boldsymbol{x})}{\theta} - \frac{\boldsymbol{A}^T(\boldsymbol{x})\boldsymbol{T}_{n-1}}{\theta}\right] \tag{18}$$

where k_i and $l_i > 0$, $i = 1, \cdots n$ are constants. The motivation for this kind of $\boldsymbol{D}_i$ construction is to offset the large control results from the state dependent term $\boldsymbol{A}(\boldsymbol{x})$ in (11)–(14). For example, when $\boldsymbol{A}(\boldsymbol{x})$ includes a cubic term, a higher initial state will result in higher initial $\boldsymbol{T}_i$ and consequently higher initial control. So if we choose $\boldsymbol{D}_i$ such that

$$-\frac{T_{i-1}A(x)}{\theta}-\frac{A^T(x)T_{i-1}}{\theta}-D_i$$
$$=\varepsilon_i(t)\left[-\frac{T_{i-1}A(x)}{\theta}-\frac{A^T(x)T_{i-1}}{\theta}\right] \tag{19}$$

where $\varepsilon_i(t)=1-k_ie^{-l_it}$ and $\varepsilon_i(t)$ is a small number, ε_i can be used to suppress this large value to propagate in (12)–(14). $\varepsilon_i(t)$ is chosen to satisfy some conditions required in the proof of convergence and stability of the algorithm [10] while e^{-l_it} with $l_i>0$ is used to let the perturbation terms in the cost function and HJB equation diminish with time. (k_i, l_i) are design parameters which can be tuned to adjust system transient performance.

Remark 2.1: Solutions to (11)–(14) are carried out offline from top to bottom. Equation (11) is a standard algebraic Riccati equation. The rest of (12)–(14) are linear equations in terms of $T_2,\cdots,T_n$ with constant coefficients $A_0-g_0R^{-1}g_0^TT_0$ and $A_0^T-T_0g_0R^{-1}g_0^T$. So we get the closed-form solutions for $T_2,\cdots,T_n$ with just one matrix inverse operation after some algebra.

Remark 2.2: θ is just an intermediate variable. It turns out to be cancelled by the choice of D_i matrices (see (16)–(18)). In the simulation, it is set to one. Theoretical work on convergence of series expansion of $\sum_{i=0}^{\infty}T_i(x,\theta)\theta^i$, semi-globally asymptotic stability of the $\theta-D$ method etc. can be found in [10].

3 Missile longitudinal autopilot design

3.1 Formulation of $\theta-D$ H_2 problem

Consider the general nonlinear system

$$\dot{x}=f(x)+B_w(x)w+B_u(x)u \tag{20}$$

$$z=c_z(x)+D_{zu}(x)u \tag{21}$$

$$y=c_y(x)+D_{yw}(x)w \tag{22}$$

where w is the exogenous input including tracking command and noises injected into the system; u is the control, z is the performance output and y is the measurement output.

The nonlinear dynamic is rewritten to have a linear-like structure as

$$\dot{x}=A(x)x+B_w(x)w+B_u(x)u \tag{23}$$

$$z=C_z(x)x+D_{zu}(x)u \tag{24}$$

$$y=C_y(x)x+D_{yw}(x)w \tag{25}$$

Then the following formulation is similar to the standard linear H_2 problem except that the coefficent matrices of x, u and w are state-dependent. This has the same formulation as SDRE H_2 at this point [9].

The linear H_2 problem leads to solving two Riccati equations given in terms of their hamiltonians

$$\begin{bmatrix} A-B_uR_2^{-1}R_{12}^T & -B_uR_2^{-1}B_u^T \\ -R_1+R_{12}R_2^{-1}R_{12}^T & -(A-B_uR_2^{-1}R_{12}^T)^T \end{bmatrix} \tag{26}$$

$$\begin{bmatrix} (A-V_{12}V_2^{-1}C_y)^T & -C_y^TV_2^{-1}C_y \\ -V_1+V_{12}V_2^{-1}V_{12}^T & -(A-V_{12}V_2^{-1}C_y) \end{bmatrix} \tag{27}$$

where

$$V_1=B_wB_w^T \quad V_{12}=B_wD_{yw}^T \quad V_2=D_{yw}D_{yw}^T$$
$$R_1=C_z^TC_z \quad R_{12}=C_z^TD_{zu} \quad R_2=D_{zu}^TD_{zu} \tag{28}$$

Assume that the solutions to (26) and (27) are P_1 and P_2. If we rewrite (20)–(22) as linear-like systems (23)–(25), the nonlinear H_2 problem needs to solve the state-dependent Riccati equation (26) and (27) where the argument x has been omitted for brevity. Construct the nonlinear feedback controller via

$$\frac{d\hat{x}}{dt}=A_c\hat{x}+B_cy \tag{29}$$

$$u=C_c\hat{x} \tag{30}$$

where A_c, B_c, and C_c are

$$A_c=A+B_uC_c-B_cC_y \tag{31}$$

$$B_c=[P_2C_y^T+V_{12}]V_2^{-1} \tag{32}$$

$$C_c=-R_2^{-1}[B_u^TP_1+R_{12}^T] \tag{33}$$

It is interesting to note that solving the state-dependent Riccati equation (26) is equivalent to solving the following nonlinear optimal control problem:

Find $u(t)$ to minimise J where

$$J=\int_0^{\infty}x^T[(R_1-R_{12}R_2^{-1}R_{12}^T)x+u^TR_2^{-1}u]dt \tag{34}$$

subject to

$$\dot{x}=[A(x)-B_u(x)R_2^{-1}R_{12}^T]x+B_u(x)u=f(x)+B_u(x)u \tag{35}$$

This class of nonlinear optimal control problem can be solved by using the $\theta-D$ technique. The same is true for the second state-dependent Riccati equation (27).

3.2 Missile longitudinal dynamics

The missile model used in this paper is taken from [9]; it assumes constant mass, post burnout, no roll rate, zero roll angle, no sideslip, and no yaw rate. The rigid body equations of motion reduce to two force equations, one moment equation, and one kinematic equation

$$\dot{U}+qW=\frac{\sum F_{B_X}}{m} \tag{36}$$

$$\dot{W}-qU=\frac{\sum F_{B_Z}}{m} \tag{37}$$

$$\dot{q}=\frac{\sum M_Y}{I_Y} \tag{38}$$

$$\dot{\theta}=q \tag{39}$$

where U and W are components of velocity vector $\vec{V}_T$ along the body-fixed x- and z-axes; θ is the pitch angle; q is the pitch rate about the body y-axis; m is the missile mass. The forces along the body–fixed co-ordinates and moments

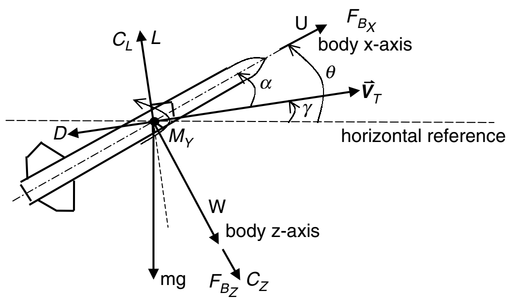

Fig. 1 *Longitudinal forces and moment acting on missile*

about the centre of gravity are shown in Fig. 1. The force and moments about the centre of gravity are

$$\sum F_{B_X} = L \sin\alpha - D\cos\alpha - mg\sin\theta \tag{40}$$

$$\sum F_{B_Z} = -L\cos\alpha - D\sin\alpha + mg\cos\theta \tag{41}$$

$$\sum M_Y = \bar{M} \tag{42}$$

where α is angle of attack; L denotes lift; D denotes drag and $\bar{M}$ is the pitching moment.

$$L = \frac{1}{2}\rho V^2 S C_L,\ D = \frac{1}{2}\rho V^2 S C_D,\ \bar{M} = \frac{1}{2}\rho V^2 S d C_m \tag{43}$$

The normal force coefficient C_Z is used to calculate the lift and drag coefficients

$$C_L = -C_Z\cos\alpha, \quad C_D = C_{D_0} - C_Z\sin\alpha \tag{44}$$

where C_{D_0} is the drag coefficient at the zero angle of attack.

The nondimensional aerodynamic coefficients at 6096m altitude are:

$$C_Z = a_n\alpha^3 + b_n\alpha|\alpha| + c_n\left(2 - \frac{M}{3}\right)\alpha + d_n\delta \tag{45}$$

$$C_m = a_m\alpha^3 + b_m\alpha|\alpha| + c_m\left(-7 + \frac{8M}{3}\right)\alpha + d_m\delta + e_m q \tag{46}$$

In this paper we adopt Mach number M, angle of attack α, flight path angle γ, and pitch rate q as the states since they appear in the aerodynamic coefficients. Note that

$$\tan\alpha = \frac{W}{U},\ V^2 = U^2 + W^2,\ M = \frac{V}{a},\ \gamma = \theta - \alpha \tag{47}$$

and

$$\dot{M} = \frac{\dot{V}}{a},\ \dot{V} = \frac{\dot{U}U + \dot{W}W}{V} \tag{48}$$

Table 1: Aerodynamic Coefficients

Force	Moment
$a_n = 19.373$	$a_m = 40.440$
$b_n = -31.023$	$b_m = -64.015$
$c_n = -9.717$	$c_m = 2.922$
$d_n = -1.948$	$d_m = -11.803$
$C_{D_0} = 0.300$	$e_m = -1.719$

Table 2: Physical Parameters

Symbol	Name	Value
P_0	Static Pressure	973.3 lb/ft^2
I_Y	Moment of Inertia	182.5 $slug - ft^2$
S	Reference Area	0.44 ft^2
d	Reference Distance	0.75 ft
m	Mass	13.98 $slug$
a	Speed of Sound	1036.4 ft/ sec
g	Gravity	32.2 ft/ sec^2

The numerical values for the coefficients in (45) and (46) are given in Table 1 and the physical parameters associated with this missile are given in Table 2.

The state equations can now be written as

$$\dot{M} = \frac{-0.7P_0S}{ma}[M^2(C_{D_0} - C_Z\sin\alpha)] - \frac{g}{a}\sin\gamma \tag{49}$$

$$\dot{\alpha} = \frac{0.7P_0S}{ma}MC_Z\cos\alpha + \frac{g}{aM}\cos\gamma + q \tag{50}$$

$$\dot{\gamma} = -\frac{0.7P_0S}{ma}MC_Z\cos\alpha - \frac{g}{aM}\cos\gamma \tag{51}$$

$$\dot{q} = \frac{0.7P_0Sd}{I_Y}M^2C_m \tag{52}$$

By substituting the aerodynamic data, (49)–(52) become

$$\begin{aligned}\dot{M} &= 0.4008M^2\alpha^3\sin\alpha - 0.6419M^2|\alpha|\alpha\sin\alpha \\ &\quad - 0.2010M^2\left(2 - \frac{M}{3}\right)\alpha\sin\alpha - 0.0062M^2 \\ &\quad - 0.0403M^2\sin\alpha\delta - 0.0311\sin\gamma\end{aligned} \tag{53}$$

$$\begin{aligned}\dot{\alpha} &= 0.4008M\alpha^3\cos\alpha - 0.6419M|\alpha|\alpha\cos\alpha \\ &\quad - 0.2010M\left(2 - \frac{M}{3}\right)\alpha\cos\alpha \\ &\quad - 0.0403M\cos\alpha\delta - 0.0311\frac{\cos\gamma}{M} + q\end{aligned} \tag{54}$$

$$\begin{aligned}\dot{\gamma} = &-0.4008M\alpha^3\cos\alpha + 0.6419M|\alpha|\alpha\cos\alpha \\ &+ 0.2010M\left(2 - \frac{M}{3}\right)\alpha\cos\alpha \\ &+ 0.0403M\cos\alpha\delta + 0.0311\frac{\cos\gamma}{M}\end{aligned} \tag{55}$$

$$\begin{aligned}\dot{q} &= 49.82M^2\alpha^3 - 78.86M^2|\alpha|\alpha \\ &\quad + 3.60M^2\left(-7 + \frac{8M}{3}\right)\alpha \\ &\quad - 14.54M^2\delta - 2.12M^2q\end{aligned} \tag{56}$$

Actuator dynamics are incorporated with the following dynamics:

$$\begin{bmatrix}\dot{\delta} \\ \ddot{\delta}\end{bmatrix} = \begin{bmatrix}0 & 1 \\ -\omega_a^2 & -2\zeta\omega_a\end{bmatrix}\begin{bmatrix}\delta \\ \dot{\delta}\end{bmatrix} + \begin{bmatrix}0 \\ \omega_a^2\end{bmatrix}\delta_c \tag{57}$$

where $\zeta = 0.7$ and $\omega_a = 50$. The normal acceleration (in gs) is described by

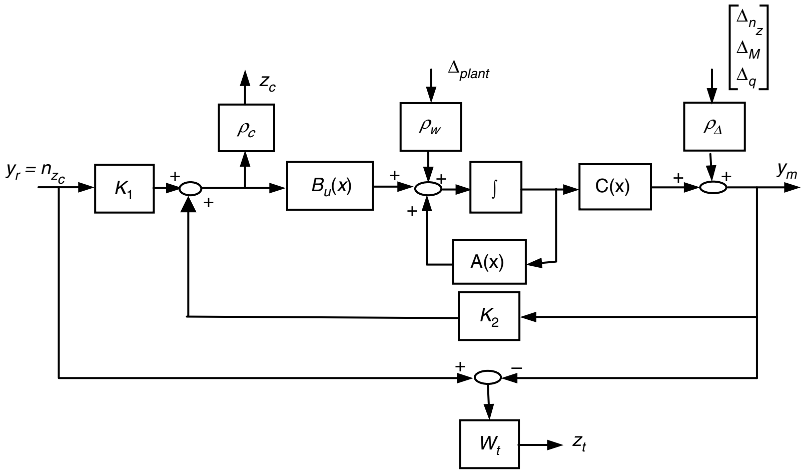

Fig. 2 *H_2 tracking block diagram*

$$n_z = \frac{\sum F_{B_Z}}{mg} + \cos\theta = \frac{0.7P_0 S}{mg} M^2 C_Z + \cos(\gamma + \alpha) \quad (58)$$

In terms of the flight conditions at 6096m,

$$\begin{aligned} n_z = {} & 12.901 M^2 \alpha^3 - 20.659 M^2 |\alpha| \alpha \\ & - 6.471 M^2 \left(2 - \frac{M}{3}\right)\alpha \\ & - 1.297 M^2 \delta + \cos(\gamma + \alpha) \end{aligned} \quad (59)$$

3.3 $\theta - D$ controller design

The controller objective is to drive the system to track the commanded normal acceleration (in gs). The tracking block diagram is shown in Fig. 2. The Kalman gain $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$ are the solutions of the dynamic feedback controller (29)–(33). The control weight is ρ_c. The plant and output disturbance weights are ρ_w and ρ_Δ. The performance weighting function for tracking error $y_r - y_m$ is chosen to be [9]

$$W_t(s) = \frac{1}{s + 0.001} \quad (60)$$

or in state-space form

$$W_t = \left[\begin{array}{c|c} A_t & B_t \\ \hline C_t & 0 \end{array}\right] = \begin{bmatrix} -0.001 & 1 \\ 1 & 0 \end{bmatrix} \quad (61)$$

Performance output is

$$\boldsymbol{z} = [z_t \quad z_c]^T \quad (62)$$

The augmented state-space $\boldsymbol{x}$ is given as

$$\boldsymbol{x} = [M, \alpha, \gamma, q, \delta, \dot{\delta}, x_t]^T \quad (63)$$

The control variable is the fin deflection

$$u = \delta_c \quad (64)$$

The measurement vector is

$$\boldsymbol{y}_m = [n_z \quad M \quad q]^T \quad (65)$$

The acceleration command

$$y_r = n_{z_c} \quad (66)$$

where n_{z_c} is the normal acceleration command. So the output vector in the controller design is

$$\boldsymbol{y} = [y_r \quad \boldsymbol{y}_m]^T = [n_{z_c} \quad \boldsymbol{y}_m]^T = [n_{z_c} \quad n_z \quad M \quad q]^T \quad (67)$$

The exogenous input is

$$\boldsymbol{w} = [n_{z_c} \quad \Delta_{plant} \quad \Delta_{n_z} \quad \Delta_M \quad \Delta_q]^T \quad (68)$$

where Δ_{plant} is the process noise and $[\Delta_{n_Z} \quad \Delta_M \quad \Delta_q]^T$ the measurement noise. In the simulation they are assumed gaussian with unit variance.

The plant noise weights are chosen to be:

$$\rho_w = [0.2 \quad 0.01 \quad 0.01 \quad 0.2 \quad 0.01 \quad 0.01]^T \quad (69)$$

The measurement noise weights on n_z, M and q are, respectively, the diagonal elements of

$$\rho_\Delta = \begin{bmatrix} 0.01 & 0 & 0 \\ 0 & 0.001 & 0 \\ 0 & 0 & 0.01 \end{bmatrix} \quad (70)$$

To avoid overflow in the numerical simulations, $\sin\gamma/\gamma$ is set to 1 when γ is less than 10^{-4} radian.

In the $\theta - D$ formulation we choose the partition of (7) as

$$\begin{aligned} \dot{\boldsymbol{x}} = {} & \left[\boldsymbol{A}(\boldsymbol{x}_0) + \theta\left(\frac{\boldsymbol{A}(\boldsymbol{x}) - \boldsymbol{A}(\boldsymbol{x}_0)}{\theta}\right)\right]\boldsymbol{x} \\ & + \left[\boldsymbol{B}(\boldsymbol{x}_0) + \theta\left(\frac{\boldsymbol{B}(\boldsymbol{x}) - \boldsymbol{B}(\boldsymbol{x}_0)}{\theta}\right)\right]\boldsymbol{u} \end{aligned} \quad (71)$$

The advantage of choosing this partition is that in the $\theta - D$ formulation $\boldsymbol{T}_0$ is solved from $\boldsymbol{A}_0$ and $\boldsymbol{g}_0$ in (7) and (11). If $\boldsymbol{A}_0 = \boldsymbol{A}(\boldsymbol{x}_0)$ and $\boldsymbol{g}_0 = \boldsymbol{B}(\boldsymbol{x}_0)$ are selected, one would have a good starting point for $\boldsymbol{T}_0$ because $\boldsymbol{A}(\boldsymbol{x}_0)$ and $\boldsymbol{B}(\boldsymbol{x}_0)$ keep much more system information than an arbitrary choice of $\boldsymbol{A}_0$ and $\boldsymbol{g}_0$.

4 Numerical results and analysis

The simulation scenario is to initially command a zero-g normal acceleration, a square wave of magnitude $10g$ at one second, returning to zero at three seconds. The initial state space is $\boldsymbol{x}_0 = [2.5 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T$. The simulation is run at 100 Hz. In solving the two state-dependent Riccati equations (26) and (27), we use $\boldsymbol{T}_0$, $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$ terms in the λ expansion (10). Three terms have been found to be sufficient. For comparison we also use the SDRE H_2 method.

The results are presented in Figs. 3–9. Figure 3 shows the commanded and achieved normal acceleration

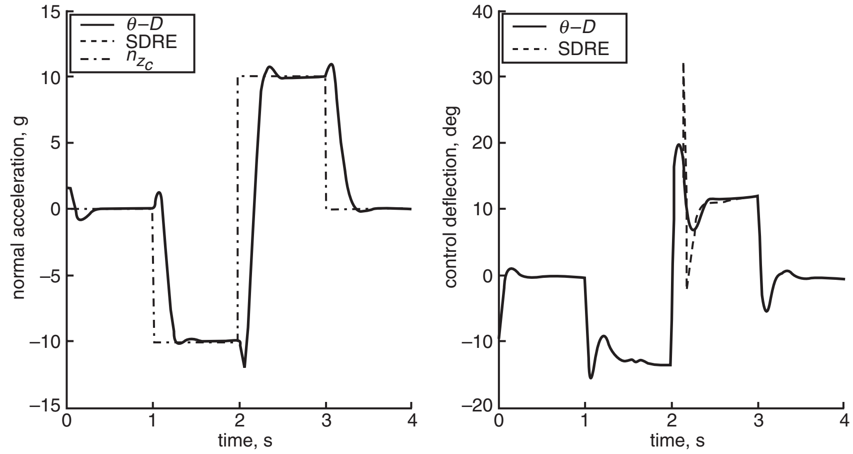

Fig. 3 *Normal acceleration tracking and control usage when $\rho_c = 1$*

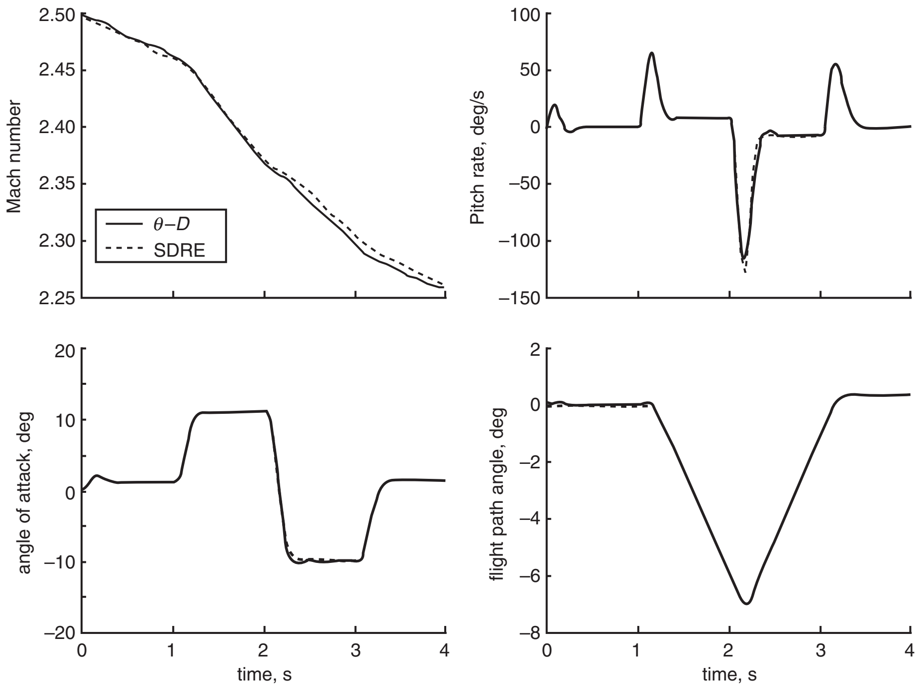

Fig. 4 *Trajectories of M, q, α and γ response when $\rho_c = 1$*

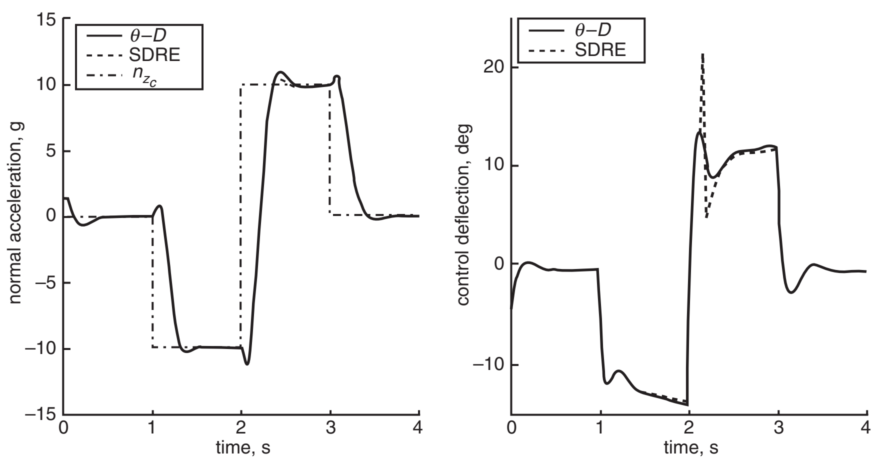

Fig. 5 *Normal acceleration tracking and control usage when $\rho_c = 2$*

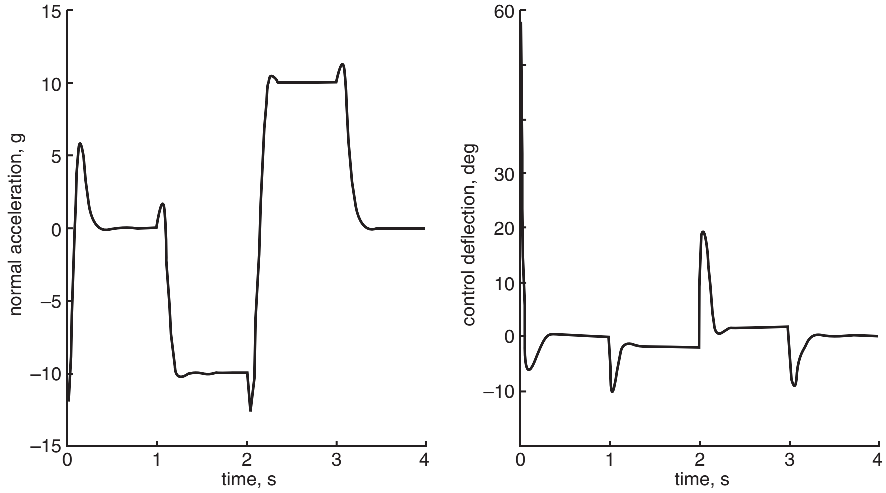

Fig. 6 *Normal acceleration tracking and control usage with* $\rho_c = 1$ *and without* $\boldsymbol{D}_i$

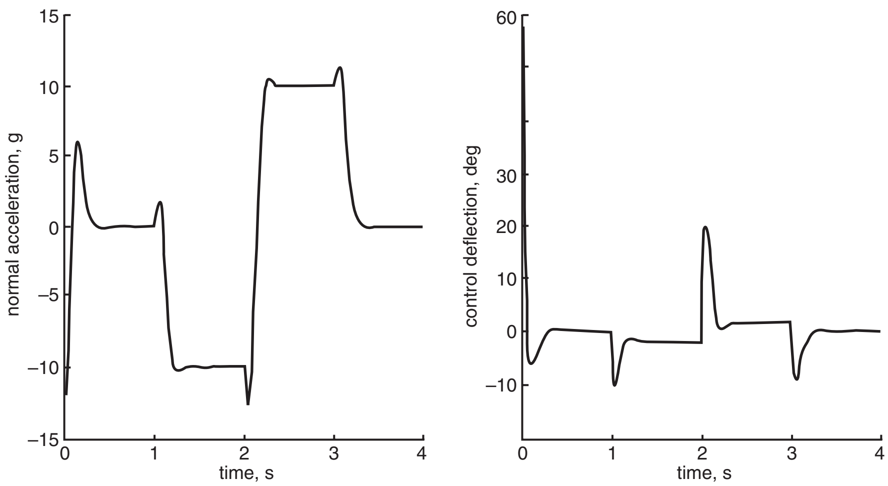

Fig. 7 *Normal acceleration tracking and control usage with* $\rho_c = 1$ *and with* $\boldsymbol{D}_i$ *added*

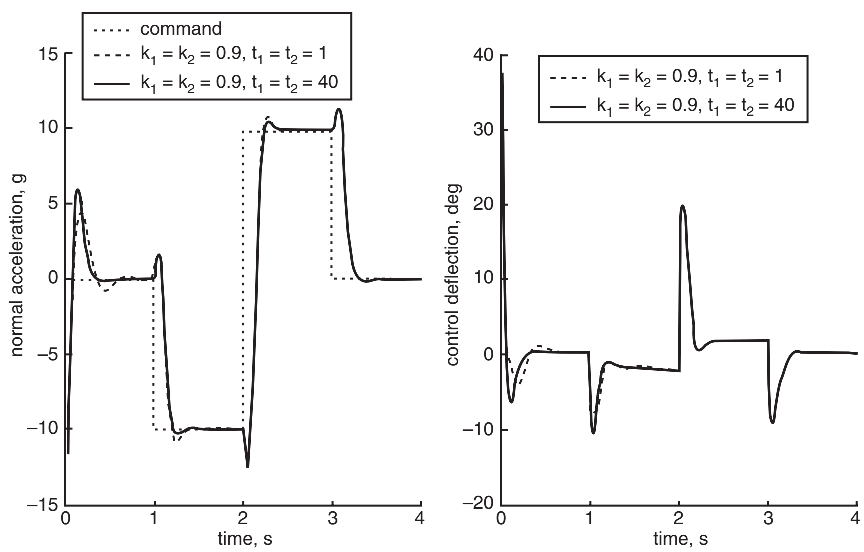

Fig. 8 *Normal acceleration tracking and control usage with* $\rho_c = 1$ *and with different* (k_i, l_i)

and the control usage when the weight ρ_c of control is 1. Both the SDRE method and the θ–D method track very well and have reasonable transient responses. Figure 4 shows that the state histories are similar for both methods. The normal acceleration tracking for SDRE has no overshoot and a little faster response. However, it needs considerable control effort at this jump as seen from Fig. 4. Figure 5 represents the effects of increase in the control weight ρ_c to 2. While

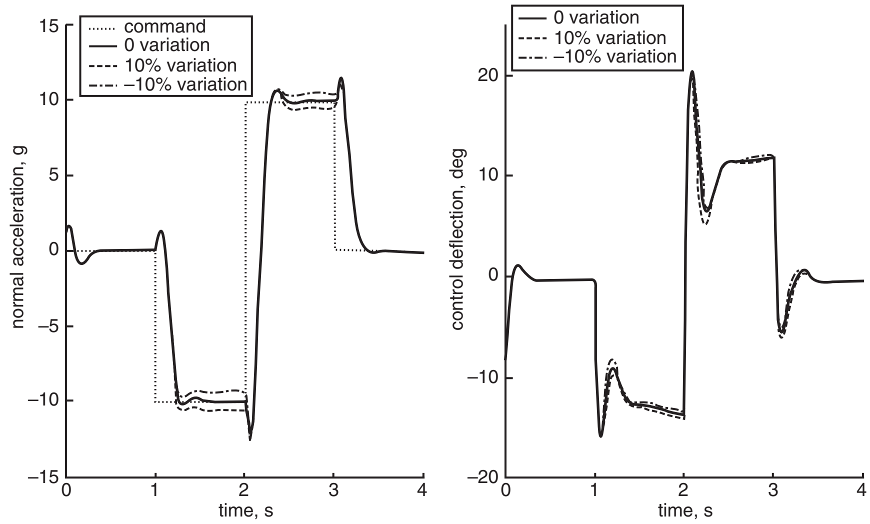

Fig. 9 *Normal acceleration tracking and control usage with parameter variations*

the control usage is reduced, the normal acceleration tracking shows more lag and overshoot.

As discussed in Section 2, the construction of perturbation matrices $\boldsymbol{D}_i$ in (16)–(19) is used to overcome the initial large control problem which is induced by the propagation of large initial states through $\theta-D$ algorithm (11)–(14). The $\theta-D$ design parameters are chosen as

$$\boldsymbol{D}_1 = e^{-40t}\left[-\frac{\boldsymbol{T}_0\boldsymbol{A}(\boldsymbol{x})}{\theta} - \frac{\boldsymbol{A}^T(\boldsymbol{x})\boldsymbol{T}_0}{\theta}\right],$$
$$\boldsymbol{D}_2 = e^{-40t}\left[-\frac{\boldsymbol{T}_1\boldsymbol{A}(\boldsymbol{x})}{\theta} - \frac{\boldsymbol{A}^T(\boldsymbol{x})\boldsymbol{T}_1}{\theta}\right] \quad (72)$$

These parameters are selected based on many initial conditions of interest. For the initial state $\boldsymbol{x}_0 = [2.5\ 0\ 0\ 0\ 0\ 0\ 0]^T$, the tracking is good and the control usage is reasonable even without $\boldsymbol{D}_1$ and $\boldsymbol{D}_2$. To demonstrate the function of $\boldsymbol{D}_1$ and $\boldsymbol{D}_2$, the results from a different initial state $\boldsymbol{x}_0 = [3.5\ 5^0\ 5^0\ 10^0/\mathrm{s}\ 0\ 0\ 0]^T$ is given in Figs. 6 and 7. As can be seen from Fig. 6, the initial maximum control is about 58^0 without $\boldsymbol{D}_1$ and $\boldsymbol{D}_2$ but is reduced to 29^0 with $\boldsymbol{D}_1$ and $\boldsymbol{D}_2$ in Fig. 7. The selection of (k_i, l_i) in $\boldsymbol{D}_i$ terms is problem dependent. A large exponential parameter is chosen in this particular problem because we found that large control only happens at the very early stage. It may not be the case for other problems in which (k_i, l_i) could be small values. These are design parameters that need tuning. Numerical experiments with these parameters show that the system performance is not sensitive to the variations around the selected values. To show this, Fig. 8 presents the results with two other sets of parameters $k_1 = k_2 = 0.9, l_1 = l_2 = 40$ and $k_1 = k_2 = 0.9, l_1 = l_2 = 1$. As can be seen, the tracking performance does not change significantly and the maximum control effort for both cases is about 37^0. Compared with large $l_1 = l_2 = 40$, the transient response in the first second with $l_1 = l_2 = 1$ is only a little worse.

We have further investigated performance robustness of both controllers to parameter variations. All aerodynamic coefficients are then changed by $\pm 10\%$ in the missile model while keeping elevator coefficients unaltered. As can be seen from Fig. 9, the performance and control usage for both methods do not change significantly with these parameter variations.

As far implementation considerations, though, the $\theta-D$ algorithm needs just one matrix inverse operation offline when solving the linear Lyapunov equations (12)–(14) and solution to the first algebraic Riccati equation (11) only one time, offline. That is to say, when solving (12)–(14), we only need to rearrange the left-hand side of the equations such that they form a linear matrix equation $\hat{\boldsymbol{A}}_0\boldsymbol{T}_i = \boldsymbol{Q}_i(\boldsymbol{x}, \theta, t)$ and then $\boldsymbol{T}_i = \hat{\boldsymbol{A}}_0^{-1}\boldsymbol{Q}_i(\boldsymbol{x}, \theta, t)$, where $\hat{\boldsymbol{A}}_0$ is a constant matrix and $\boldsymbol{Q}_i(\boldsymbol{x}, \theta, t)$ is the right-hand side of (12)–(14). When implemented online, this method involves only two 7×7 matrix multiplications and three 7×7 matrix additions if we take three terms. However, in comparison, SDRE needs computation of the 7×7 algebraic Riccati equation at *each* sample time.

5 Conclusions

A new suboptimal nonlinear control synthesis technique has been applied to the missile longitudinal autopilot design. The new nonlinear $\theta-D$ H_2 design extends the applicability of the linear H_2 design. Compared with the SDRE H_2 design, this approach does not need the intensive online solutions of the Riccati equation.

6 Acknowledgment

A grant from the Naval Surface Warfare Center, Dahlgren Division, in support of this study is gratefully acknowledged.

7 References

1 Bryson, A.E. Jr., and Ho, Y-C.: 'Applied optimal control' (Hemisphere, Washington DC, 1975)
2 Wernli, A., and Cook, G.: 'Suboptimal control for the nonlinear quadratic regulator problem', *Automatica*, 1975, **11**, pp. 75–84
3 Garrard, W.L.: 'Design of nonlinear automatic flight control systems', *Automatica*, 1977, **13**, pp. 497–505
4 Garrard, W.L., Enns, D.F., and Snell, S.A.: 'Nonlinear feedback control of highly manoeuverable aircraft', *Int. J. Control*, 1992, **56**, pp. 799–812
5 Beard, R.W., Saridis, G.N., and Wen, J.T.: 'Galerkin approximations of the generalized Hamilton–Jacobi-Bellman equation', *Automatica*, 1997, **33**, (12), pp. 2156–2177
6 McLain, T.W., and Beard, R.W.: 'Nonlinear optimal control design of a missile autopilot'. Presented at the AIAA Conference on Guidance, Navigation and Control, Boston, MA, 10–12 August 1998
7 Cloutier, J.R., D'Souza, C.N., and Mracek, C.P.: 'Nonlinear regulation and nonlinear H_∞ control via the state-dependent Riccati equation technique'. Presented at the 1st Int. Conf. on Nonlinear Problems in Aviation and Aerospace, Daytona Beach, FL, May 1996
8 Zhou, K., and Doyle, J.C.: 'Essentials of robust control' (Prentice-Hall, Upper Saddle River, NJ, 1998)
9 Mracek, C.P., and Cloutier, J.R.: 'Missile Longitudinal Autopilot Design Using the State-Dependent Riccati Equation Method'. Presented at the Conf. on Nonlinear Problems in Aviation and Aerospace, Daytona Beach, FL, May, 1996
10 Xin, M., and Balakrishnan, S.N.: 'A new method for suboptimal control of a class of nonlinear systems'. Presented at the IEEE Conf. on Decision and Control, Las Vegas, Nevada, 10–13 Dec. 2002