Nonlinear Missile Autopilot Design with $\theta - D$ Technique

Ming Xin*, S.N. Balakrishnan**
University of Missouri-Rolla, Rolla, MO 65409

and

Donald T. Stansbery+  
Boeing Company, St. Louis, MO 63166

Ernest J. Ohlmeyer++
Naval Surface Weapon Center, Dahlgren, VA 22448-5000

Abstract

In this paper, a new nonlinear control method is used to design a full-envelope, hybrid bank-to-turn (BTT)/skid-to-turn (STT) autopilot for an air-breathing air-to-air missile. Through this new approach, called the $\theta - D$ method, we find approximate solutions to the Hamilton-Jacobi-Bellman (HJB) equation. As a result, the resulting nonlinear feedback law can be expressed in a closed form. In this paper, a $\theta - D$ outer-loop and inner-loop controller structure is used in an autopilot design. A hybrid BTT/STT autopilot command logic is used to convert the commanded accelerations from the guidance laws to reference angle commands for the autopilot. The outer-loop $\theta - D$ controller converts the angle-of-attack, sideslip, and bank angle commands to body rate commands for the inner loop. An inner-loop $\theta - D$ controller converts the body rate commands to fin commands. This design is evaluated using a detailed six-degrees-of-freedom simulation. Numerical results show that the new controllers achieve excellent tracking performance and exhibit insensitivity to parameter variations over a wide flight envelope.

Keywords: Nonlinear Systems, Optimal Control, Missile Autopilot, Bank-to-turn/Skid-to-Turn Missile

* Postdoctoral Research Fellow, Department of Mechanical and Aerospace Engineering and Engineering Mechanics, University of Missouri-Rolla, MO 65401. xin@umr.edu. Member AIAA.

** Professor, Department of Mechanical and Aerospace Engineering and Engineering Mechanics, University of Missouri-Rolla, MO 65401. bala@umr.edu. Associate Fellow AIAA.

+ Research Scientist, Phantom Works, Boeing Company at Saint Louis, P.O. Box 516, MO 63166.

Donald.t.stansbery@boeing.com. Senior Member AIAA.

++: Scientist, Naval Surface Weapon Center, Attn: G23, 17320 Dahlgren Road, Dahlgren, VA 22448-5000 Associate Fellow AIAA
Nomenclature

$\alpha$  Angle of attack

$\beta$  Side-slip angle

$\mu$  Bank angle about the velocity vector

$\gamma$  Flight path angle

$p,q,r$  Body-frame roll, pitch and yaw rate

$V$  Missile speed

$M$  Mach number

$m$  Missile mass

$q$  Dynamic pressure

$S$  Missile cross-sectional area

$d$  Missile Diameter

$g$  Gravity

$h$  Missile flight altitude

$I_x, I_y, I_z$  Moments of inertia about body-frame

$\delta p, \delta q, \delta r$  Aileron, elevator, and rudder fin deflections

$C_A$  Axial force coefficient

$C_Y$  Side force coefficient

$C_Z (-C_N)$  Normal force coefficient

$C_{N_0}, C_{\gamma_0}$  Normal and side force coefficients with zero fin deflections

$C_{N_{\delta p}}, C_{N_{\delta q}}, C_{N_{\delta r}}$  Normal force coefficients with respect to aileron, elevator and rudder fin respectively

$C_{Y_{\delta p}}, C_{Y_{\delta q}}, C_{Y_{\delta r}}$  Side force coefficients with respect to aileron, elevator and rudder fin respectively

$C_l$  Roll moment coefficient

$C_m$  Pitch moment coefficient
\( C_n \)  
Yaw moment coefficient

\( C_{l_0} \)  
Roll moment coefficient with zero fin deflections

\( C_{l_p} \)  
Roll moment coefficient with respect to roll rate

\( C_{l_p}, C_{l_q}, C_{l_r} \)  
Roll moment coefficient with respect to aileron, elevator and rudder fin respectively

\( C_{m_0} \)  
Pitch moment coefficient with zero fin deflections

\( C_{m_q} \)  
Pitch moment coefficient with respect to pitch rate

\( C_{m_q}, C_{m_q}, C_{m_r} \)  
Pitch moment coefficient with respect to aileron, elevator and rudder fin respectively

\( C_{n_0} \)  
Yaw moment coefficient with zero fin deflections

\( C_{n_r} \)  
Yaw moment coefficient with respect to yaw rate

\( C_{n_r}, C_{n_q}, C_{n_r} \)  
Yaw moment coefficient with respect to aileron, elevator and rudder fin, respectively

\( a_y', a_z' \)  
Acceleration along the inertial y and z axis

\( T \)  
Thrust

1. Introduction:
Missile and aircraft autopilot designs have been mostly dominated by classical control techniques. They require a great deal of tuning and ad hoc modifications are often unavoidable. Cloutier et al. [1] surveyed on the classical designs as well as a variety of gain-scheduled modern control designs such as LQG/LTR and eigenstructure assignment in the context of bank-to-turn missile autopilot designs. Gain-scheduling has produced many highly reliable and effective control systems. The drawback of this method is that information about the actual nonlinear behavior is usually discarded. Williams et al. [2] proposed a gain-scheduled LQG controller to account for nonlinear kinematic coupling terms by scheduling the linear pitch/yaw channel gains
as a function of both dynamic pressure and body-axis roll rate. This approach achieved good performance over a range of operating conditions although unsuitable for high angle of attack maneuvering since it does not consider the aerodynamic nonlinearities. Krause and Stein [3] presented an adaptive longitudinal autopilot scheduled on the basis of an estimated pitch control derivative. Kamen et al. [4] synthesized an adaptive autopilot for a discrete-time linear-time-varying (LTV) missile model. Tan et al. [5] applied linear parameter-varying (LPV) control theory to the design of a gain-scheduled missile autopilot.

A number of nonlinear control methods have also been proposed for the missile autopilot design. Dynamic Inversion (DI) was used for the inner-loop design in the inner/outer loop control structure first proposed by Adams [6,7]. In the inner loop the plant dynamics are equalized across the flight envelope using DI. $\mu$-synthesis was employed to design the outer loop to achieve performance and robustness requirements. McFarland and D’Souza [8] also combined dynamic inversion control and $\mu$-synthesis in the missile autopilot design. Since DI replaces the set of existing dynamics with a designer selected set of dynamics, Georgie and Valasek [9] attempted to quantify the particular form of desired dynamics which produce the best closed-loop performance and robustness in a DI flight controller. Schumacher and Khargonekar [10] compared gain-scheduled $H_\infty$ control with dynamic inversion using linearized analysis and found that the latter lacked in robustness. Wise and Sedwick [11] applied nonlinear $H_\infty$ theory to a missile with coupled aerodynamic and thrust-vectoring control. However, they showed that the nonlinear design did not significantly improve the performance of a well-designed gain-scheduled linear autopilot. There are also examples of missile autopilot designs using sliding mode control [12-15].
Another recently emerging technique that systematically solves the nonlinear regulator problem is the State Dependent Riccati Equation (SDRE) method (Cloutier et al., 1996) [16]. 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 $H_\infty$ design technique for the synthesis of nonlinear control systems. It can be used for a broad class of nonlinear regulator problems. It has been employed to design advanced guidance algorithms in [17] and used in [18] for integrated missile guidance and control design. Wise and Sedwick [19] also reformulated their earlier nonlinear $H_\infty$ missile autopilot design [11] using the SDRE framework, which led to a simplified feedback control form for this nonlinear control application. The SDRE method however, needs online computation of the algebraic Riccati equation at each sample time. The $\theta-D$ method developed in this study though similar to the SDRE technique has a closed form solution.

In this paper, the $\theta-D$ approach is formulated as a way to find an approximate solution to the Hamilton-Jacobi-Bellman (HJB) equation. By introducing an intermediate variable $\theta$, the co-state $\lambda$ can be expanded as a power series in terms of $\theta$. The HJB equation is then reduced to a set of recursive algebraic equations. By adding perturbations to the cost function and manipulating these terms appropriately semi-global asymptotic stability can be achieved [20]. By adjusting the parameters in the perturbation terms, we are also able to modulate the transient performance of the system.

There are two basic modes of controlling the attitude of a missile to achieve the acceleration commanded by the guidance law: skid-to-turn (STT) and bank-to-turn (BTT). In the STT mode, the roll angle may be held constant or uncontrolled. The major advantage of STT is its faster response. A BTT missile allows only positive angles of attack while maintaining small sideslip
angles to prevent missile maneuvers from shading the inlet in order to increase engine efficiency and thereby maximize range. In this paper, we apply the $\theta-D$ technique to design a hybrid BTT/STT autopilot for an air-to-air missile to use their respective advantages.

It is an accepted result in flight mechanics that the flight control problem is structured in two layers. The motion of the center of gravity is addressed in the outer loop while the angular motion around the center of gravity is taken care of by the inner loop. Likewise, the control structure can also be structured into an outer-loop control which provide the servo tracking of prescribed flight angle command and the inner-loop control providing servo tracking of angular rates. Tournes and Johnson [21] developed an inner-loop controller for a combat aircraft based on Subspace Stabilization Control Theory to track the attitude rates. The outer-loop controller was designed with the Linear Adaptive technique to follow the flight path and ground track angles. Good results were obtained in both tracking performance and robustness. The SDRE technique is used in [22] to design the inner/outer loop, full-envelope autopilot for Bank-to-Turn/Skid-to-Turn missiles from an optimal control point of view.

This paper is organized as follows: The $\theta$-D approximation method is formulated in Section 2. In Section 3, the missile dynamics are described. The $\theta-D$ inner/outer loop autopilot design is presented in Section 4. In Section 5, the BTT/STT command logic is presented. Simulation results and analysis are given in Section 6. Conclusions are made in Section 7.

2. $\theta-D$ Suboptimal Control Method

In this paper we consider optimal control of systems of the form

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

with a cost function given by

$$J = \frac{1}{2} \int_0^\infty (x^T Q(x)x + u^T Ru)dt$$
where \( x \in \mathbb{R}^n, f \in \mathbb{R}^n, B \in \mathbb{R}^{m \times n}, u \in \mathbb{R}^m, Q \in \mathbb{R}^{m \times m}, R \in \mathbb{R}^{m \times m} \); Assume that \( x \in \Omega \) and \( \Omega \) is a compact set in \( \mathbb{R}^n \); \( Q(x) \) is semi-positive definite and \( R \) is a positive definite constant matrix; It is assumed that \( f(0)=0 \);

To ensure that the control problem is well posed we assume that a solution to the optimal control problem (1), (2) exists. We also assume that \( f(x) \) is of class \( C^1 \) in \( x \) on a compact set \( \Omega \) and zero state observable through \( Q \).

The optimal solution of the infinite-horizon nonlinear regulator problem can be obtained by solving the Hamilton-Jacobi-Bellman (HJB) partial differential equation [23]:

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

where \( V(x) \) is the optimal cost, i.e.

\[
V(x) = \min_{u} \int_0^\infty (x^T Q(x)x + u^T Ru) dt \tag{4}
\]

We assume that \( V(x) \) is continuously differentiable and \( V(x) > 0 \) with \( V(0)=0 \).

Optimal control is given by

\[
u = -R^{-1} B^T(x) \frac{\partial V}{\partial x} \tag{5}
\]

The HJB equation is extremely difficult to solve in general, rendering optimal control techniques of limited use for nonlinear systems.

Now consider perturbations added to the cost function:

\[
J = \frac{1}{2} \int_0^\infty [x^T (Q(x) + \sum_{i=1}^\infty D_i \theta^i) x + u^T Ru] dt \tag{6}
\]

where \( \theta \) and \( D_i \) are chosen such that \( Q(x) + \sum_{i=1}^\infty D_i \theta^i \) is semi-positive definite.

Write the original state equation as:
\[ \dot{x} = f(x) + B(x)u = \left[ A_0 + \theta \left( \frac{A(x)}{\theta} \right) \right] x + \left[ g_0 + \theta \left( \frac{g(x)}{\theta} \right) \right] u \] (7)

where \( A_0 \) and \( g_0 \) are constant matrices such that \((A_0, g_0)\) is a stabilizable pair and \([(A_0 + A(x)), (g_0 + g(x))]\) is pointwise controllable.

Also write the perturbed cost function as

\[ J = \frac{1}{2} \int_0^T \left[ Q_0 + \theta \frac{Q(x)}{\theta} + \sum_{i=1}^{n} D_i \theta^i \right] x + u^T R u \] \[ + \lambda^T (x) x + \lambda^T (x) u + \lambda^T (x) R u dt \]

such that \( Q(x) = Q_0 + Q_i (x) \) and \( Q_0 \) is a constant matrix.

Define

\[ \lambda = \frac{\partial V}{\partial x} \] (9)

By using (8) and (9) in HJB equation (3) we have the perturbed HJB equation:

\[ \lambda^T f(x) - \frac{1}{2} \lambda^T B(x) R^{-1} B^T (x) \lambda + \frac{1}{2} x^T \left[ Q_0 + \theta \frac{Q(x)}{\theta} + \sum_{i=1}^{n} D_i \theta^i \right] x = 0 \] (10)

Assume a power series expansion of \( \lambda \) in terms of \( \theta \)

\[ \lambda = \frac{\partial V}{\partial x} = \sum_{i=0}^{\infty} T_i \theta^i x \] (11)

where \( T_i \) are assumed to be symmetric and to be determined.

Substitute (11) into the HJB equation (10) and equate the coefficients of powers of \( \theta \) to zero to get the following equations:

\[ T_0 A_0 + A_0^T T_0 - T_0 g_0 R^{-1} g_0^T T_0 + Q_0 = 0 \] (12)

\[ T_1 (A_0 - g_0 R^{-1} g_0^T T_0) + (A_0^T - T_0 g_0 R^{-1} g_0^T T_0) T_0 = -\frac{T_0 A(x)}{\theta} \left[ \frac{A(x)}{\theta} \right] T_0 + T_0 g_0 R^{-1} g_0^T T_0 + T_0 g_0^T T_0 - D_1 - \frac{Q(x)}{\theta} \] (13)

\[ T_2 (A_0 - g_0 R^{-1} g_0^T T_0) + (A_0^T - T_0 g_0 R^{-1} g_0^T T_0) T_2 = -\frac{T_0 A(x)}{\theta} \left[ \frac{A(x)}{\theta} \right] T_2 + T_0 g_0 R^{-1} g_0^T T_0 + T_0 g_0^T T_0 - D_2 \] (14)
\[ T_u(A_0 - g_0 R^{-1} g_0^T T_u) + (A_0^T - T_0 g_0 R^{-1} g_0^T T_u) = - \frac{T_u A(x) T_u}{\theta} - \frac{A^T(x) T_u}{\theta} - D_n + \sum_{j=0}^{n-1} T_j (g_0 R^{-1} g_0^T T_{n-j}) + \sum_{j=0}^{n-1} T_j g_0 R^{-1} g_0^T T_{n-j} \]

Since the right hand side of equations (12)-(15) involve \( x \) and \( \theta \), \( T_i \) would be a function of \( x \) and \( \theta \). Thus we denote it as \( T_i(x, \theta) \). The expression for control can be obtained in terms of the power series:

\[ u = -R^{-1}B^T(x)\frac{\partial V}{\partial x} = -R^{-1}B^T(x)\sum_{i=0}^{\infty} T_i(x, \theta)\theta^i x \]

It is easy to see that the equation (12) is an algebraic Riccati equation. The rest of equations are Lyapunov equations that are \textit{linear} in terms of \( T_i \) \( (i = 1 \cdots n) \).

We construct the following expression for \( D_i, i = 1, \cdots n \): 

\[ D_i = k_i e^{-l_i} \left[ - \frac{T_u A(x) T_u}{\theta} - \frac{A^T(x) T_u}{\theta} + T_0 g_0 R^{-1} g_0^T T_0 + T_0 \frac{g}{\theta} R^{-1} g_0^T T_0 - \frac{Q_0(x)}{\theta} \right] \]

\[ D_i = k_i e^{-l_i} \left[ - \frac{T_u A(x) T_u}{\theta} - \frac{A^T(x) T_u}{\theta} + T_0 g_0 R^{-1} g_0^T T_0 + T_0 \frac{g}{\theta} R^{-1} g_0^T T_0 + T_0 \frac{g}{\theta} R^{-1} g_0^T T_0 + T_0 g_0 R^{-1} g_0^T T_0 + T_0 g_0 R^{-1} g_0^T T_0 \right] \]

\[ \vdots \]

\[ D_n = k_n e^{-l_n} \left[ - \frac{T_u A(x) T_u}{\theta} - \frac{A^T(x) T_u}{\theta} + \sum_{j=0}^{n-1} T_j (g_0 R^{-1} g_0^T T_{n-j}) T_{n-j} + \sum_{j=0}^{n} T_j \frac{g}{\theta} R^{-1} g_0^T T_{n-j} + \sum_{j=0}^{n} T_j g_0 R^{-1} g_0^T T_{n-j} \right] \]

where \( k_i \) and \( l_i > 0, i = 1, \cdots n \) are adjustable design parameters.

The idea in constructing \( D_i \) in this manner is based on the observation that large initial states may give rise to large initial control due to the state dependent term \( A(x) \) on the right-hand side of the equations (13)-(15). It happens when there are some terms in \( A(x) \) which could grow to a
high magnitude as \( x \) is large. To see this, for example, when \( A(x) \) includes a cubic term, its magnitude could be large if \( x \) is large. This large value will be reflected into the solution for \( T_i \), i.e. the left-hand side of equations (13)-(15). Since \( T_i \) will be used in the next equation to solve for \( T_{i+1} \), this large value will be propagated and amplified and consequently cause higher control or even instability. So if we choose \( D_i \) such that

\[
-T_i A(x) - \frac{A'(x) T_i}{\theta} + \sum_{j=0}^{L-1} T_j \left( Q_j R^{-1} g^T_j + g_j R^{-1} g^T_j T_i \right) + \sum_{j=0}^{L-1} T_j \frac{g_j}{\theta} R^{-1} g^T_j T_{i-2-j} + \sum_{j=0}^{L-1} T_j g_j R^{-1} g^T_j T_{i-j} - D_i
\]

\[
= \varepsilon_i(t) \left[ -T_i A(x) - \frac{A'(x) T_i}{\theta} + \sum_{j=0}^{L-1} T_j \left( Q_j R^{-1} g^T_j + g_j R^{-1} g^T_j T_i \right) + \sum_{j=0}^{L-1} T_j \frac{g_j}{\theta} R^{-1} g^T_j T_{i-2-j} + \sum_{j=0}^{L-1} T_j g_j R^{-1} g^T_j T_{i-j} \right]
\]

(20)

where

\[
\varepsilon_i(t) = 1 - k_i e^{-l_i t}
\]

(21)

is a small number, \( \varepsilon_i \) can be used to suppress this large value from propagating in the equations (13) through (15). \( \varepsilon_i(t) \) is chosen to satisfy some conditions required in the proof of convergence and stability of the above algorithm [20]. On the other hand, the exponential term \( e^{-l_i t} \) with \( l_i > 0 \) is used to let the perturbation terms in the cost function and HJB equation diminish as time evolves.

**Remark 2.1**

Solving equations (12)-(15) is carried out offline successively from top to bottom, i.e. \( T_u \) can be solved from \( T_{n-1} \). Equation (12) is a standard algebraic Riccati equation. Once \( A_0, g_0, Q_0 \) and \( R \) are determined, \( T_0 \) can be solved to be a constant matrix. The rest of the equations (13)-(15) are linear equations in terms of \( T_2, \ldots, T_u \) with constant coefficients \((A_0 - g_0 R^{-1} g^T_0 T_0)\) and \((A_0^T - T_0 g_0 R^{-1} g^T_0 T_0^T)\) once we obtain the solution for \( T_0 \) from Eq. (12). With some linear algebra,
the equations (13)-(15) can be rearranged in the form of \( \hat{A}_o T_i = Q_i(x, \theta, t) \). As a result, \( T_i \) can be written in a closed form, \( T_i = \hat{A}_o^{-1} Q_i(x, \theta, t) \) where \( \hat{A}_o \) is a constant matrix. So we can get closed form solutions to \( T_2, \cdots, T_n \) with just one matrix inverse operation. The expression of \( Q_i(x, \theta, t) \) on the right hand side of the equations is already known and only needs simple matrix multiplications and additions.

**Remark 2.2:**

The construction of \( D_i \) in (17)-(20) serves three functions. The first is to suppress the large control if it happens. The second is to provide an appropriate \( \epsilon_i \) to guarantee the convergence of power series expansion \( \sum_{i=0}^{\infty} T_i(x, \theta) \theta^i \) and stability of the closed loop system [20]. The third function is to allow flexibility to modulate the system performance by tuning the parameters of \( k_i \) and \( l_i \) in the \( D_i \). This will be shown in the autopilot design.

**Remark 2.3:**

\( \theta \) is just an intermediate variable and its value can be kept as unity.

3. Missile Dynamics

Equations of motion of a generic air-to-air missile dynamics are given below [22] in terms of missile’s speed \( V \), angle of attack \( \alpha \), sideslip \( \beta \), bank angle \( \mu \), roll rate \( p \), pitch rate \( q \), and yaw rate \( r \):

\[
\dot{V} = -\frac{\bar{q}S}{m} C_A \cos \alpha \cos \beta + \frac{\bar{q}S}{m} C_{y_V} \sin \beta - \frac{\bar{q}S}{m} C_{N_k} \sin \alpha \cos \beta - g \sin \gamma + \frac{\cos \alpha \cos \beta}{m} T \\
+ \frac{\bar{q}S}{m} \left[ (C_{y_K} \sin \beta - C_{n_{y_K}} \sin \alpha \cos \beta) \delta p - (C_{n_{y_K}} \sin \beta - C_{N_{y_K}} \sin \alpha \cos \beta) \delta q - (C_{y_R} \sin \beta - C_{n_{y_R}} \sin \alpha \cos \beta) \delta r \right] \tag{22}
\]

\[
\dot{\alpha} = q - \tan \beta \cos \alpha p - \tan \beta \sin \alpha r + \frac{g}{V \cos \beta} \cos \gamma \cos \mu - \frac{\bar{q}S}{m \cos \beta} C_{N_k} \cos \alpha + \frac{\bar{q}S}{m \cos \beta} C_A \sin \alpha
\]
\[
\dot{\beta} = \sin\alpha p - \cos\alpha r + \frac{g}{V} \cos\gamma \sin\mu + \frac{\bar{q}S}{mV} C_{n_y} \sin\alpha \sin\beta + \frac{\bar{q}S}{mV} C_{\alpha} \cos\alpha \sin\beta + \frac{\bar{q}S}{mV} C_{\gamma} \cos\beta - \frac{\cos\alpha \sin\beta}{mV} T
\]

\[
+ \frac{\bar{q}S}{mV} [(C_{n_y} \cos\beta + C_{n_y} \sin\alpha \sin\beta) \delta p - (C_{n_y} \cos\beta - C_{n_y} \sin\alpha \sin\beta) \delta q - (C_{n_y} \cos\beta - C_{n_y} \sin\alpha \sin\beta) \delta r]
\]

\[
\dot{\mu} = p \frac{\cos\alpha}{\cos\beta} + r \frac{\sin\alpha}{\cos\beta} + \frac{T}{mV} [\sin\alpha \sin\mu \tan\gamma - \cos\alpha \sin\beta \cos\mu \tan\gamma + \sin\alpha \tan\beta]
\]

\[
+ \frac{\bar{q}S}{mV} [-C_{\alpha} \sin\alpha (\sin\mu \tan\gamma + \tan\beta) + C_{\alpha} \cos\alpha \cos\mu \sin\beta \tan\gamma + C_{n_y} \sin\alpha \cos\mu \sin\beta \tan\gamma + C_{n_y} \cos\mu \cos\beta \tan\gamma]
\]

\[
- \frac{g}{V} \cos\gamma \cos\mu \tan\beta + \frac{\bar{q}S}{mV} [C_{n_y} \delta p - C_{n_y} \delta q - C_{n_y} \delta r] [\cos\alpha (\sin\mu \tan\gamma + \tan\beta) + \sin\alpha \cos\mu \tan\beta \gamma]
\]

\[
+ \frac{\bar{q}S}{mV} [C_{n_y} \delta p - C_{n_y} \delta q - C_{n_y} \delta r] \cos\mu \cos\beta \tan\gamma
\]

\[
p = \frac{I_z - I_x}{I_y} \frac{q^2}{2I_x} (C_{p_y} \cos\alpha p - C_{n_y} \sin\alpha r) + \frac{\bar{q}Sd}{I_y} (C_{\gamma} \cos\alpha - C_{n_y} \sin\alpha)
\]

\[
+ \frac{\bar{q}Sd}{I_y} [(C_{p_y} \cos\alpha - C_{n_y} \sin\alpha) \delta p - (C_{p_y} \cos\alpha - C_{n_y} \sin\alpha) \delta q - (C_{p_y} \cos\alpha - C_{n_y} \sin\alpha) \delta r]
\]

\[
q = \frac{I_z - I_x}{I_y} \frac{r^2}{2I_y} (C_{m_y} + C_{m_y} \delta p - C_{m_y} \delta q - C_{m_y} \delta r)
\]

\[
+ \frac{\bar{q}Sd}{I_y} (C_{p_y} \sin\alpha p + C_{n_y} \cos\alpha r) + \frac{\bar{q}Sd}{I_z} (C_{\gamma} \sin\alpha \cos\alpha + C_{n_y} \cos\alpha) + \frac{\bar{q}Sd}{I_z} [(C_{p_y} \sin\alpha + C_{n_y} \cos\alpha) \delta p
\]

\[
- (C_{n_y} \sin\alpha + C_{n_y} \cos\alpha) \delta q - (C_{n_y} \sin\alpha + C_{n_y} \cos\alpha) \delta r]
\]

4. \(\theta-D\) Autopilot Design

4.1 Outer-loop and Inner-loop Tracking Structure

The autopilot design is performed in a two-loop structure shown in Figure 1. Acceleration commands from the missile guidance law are converted to \(\alpha_e, \beta_e\) and \(\mu_e\) which are angle-of-attack, sideslip, and bank angle commands respectively by using the BTT/STT logic which will
be presented in Section 5. The outer-loop converts $\alpha_c, \beta_c$ and $\mu_c$ to roll-rate, pitch-rate, and yaw-rate commands $p_c, q_c$ and $r_c$ respectively for the inner loop. The inner loop then converts $p_c, q_c$ and $r_c$ to fin commands $\delta_{p_c}, \delta_{q_c}$ and $\delta_{r_c}$ for the actuators.

Figure 1: $\theta - D$ Outer-loop and Inner-loop Autopilot Structure

In order to design the autopilot to perform command following, the $\theta - D$ controller is implemented as an integral servo-mechanism. This is accomplished as follows. First, the state $x$ is decomposed as

$$x = \begin{bmatrix} x_T \\ x_N \end{bmatrix}$$

(29)

where $x_T$ consists of the tracked states which should track a reference command $r_c$. $x_N$ consists of the nontracked states. The state vector $x$ is then augmented with $x_I$, the integral states of $x_T$:

$$\tilde{x} = \begin{bmatrix} x_I \\ x_T \\ x_N \end{bmatrix}$$

(30)

The augmented system is given by

$$\dot{\tilde{x}} = \tilde{f}(\tilde{x}) + \tilde{B}(\tilde{x})u$$

(31)

where

$$\tilde{B}(\tilde{x}) = \begin{bmatrix} 0 \\ B(x) \end{bmatrix}$$

(32)

So the problem addressed here is to find an optimal controller to minimize the cost functional

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

(33)
The $\theta-D$ integral controller is given by

$$u = -\tilde{R}(\tilde{x})^{-1}\tilde{B}(\tilde{x})^T\sum_{n=0}^{\infty} T_n(\tilde{x}, \theta)\theta^n \left[ x_f - \int r_c dt \right]$$

where $r_c$ is the command that it is desired to track. In the $\theta-D$ controller design, use of the first three terms in $\sum_{n=0}^{\infty} T_n(\tilde{x}, \theta)\theta^n$ has been found to be accurate enough for this problem (as well as some others that have been solved).

In the $\theta-D$ formulation, we choose the factorization of nonlinear equation (7) in this way:

$$\dot{\tilde{x}} = \tilde{A}(\tilde{x}_0) + \theta \left( \frac{\tilde{A}(\tilde{x}) - \tilde{A}(\tilde{x}_0)}{\theta} \right) \tilde{x} + \theta \left( \frac{\tilde{B}(\tilde{x}) - \tilde{B}(\tilde{x}_0)}{\theta} \right) u$$

(35)

The advantage of choosing this factorization is that in the $\theta-D$ formulation $T_0$ is solved from $A_0$ and $g_0$ in Eq. (7) and Eq. (12). If we select $A_0 = \tilde{A}(\tilde{x}_0)$ and $g_0 = \tilde{B}(\tilde{x}_0)$, we would have a good starting point for $T_0$ because $\tilde{A}(\tilde{x}_0)$ and $\tilde{B}(\tilde{x}_0)$ keep much more system information than an arbitrary choice of $A_0$ and $g_0$.

4.2 $\theta-D$ Outer-loop Design

For the outer loop, the state space is chosen to be:

$$\tilde{x} = [V \alpha_t \alpha \beta_t \beta \mu_t \mu]^T$$

(36)

with control vector $u$ given by:

$$u = [p, q, r]^T$$

(37)

The tracking command is:

$$r_c = [\alpha_c \beta_c \mu_c]^T$$

(38)

This command is obtained from the guidance law by using the BTT/STT Logic which will be presented in Section 5.
The outer-loop state-dependent coefficient factorization $\tilde{A}_{OL}(\tilde{x})$ is chosen as:

$$
\tilde{A}_{OL}(\tilde{x}) = 
\begin{bmatrix}
    a_{11} & a_{13} & 0 & a_{15} & 0 & 0 \\
    0 & 0 & 1 & 0 & 0 & 0 \\
    a_{31} & 0 & a_{33} & 0 & 0 & 0 \\
    0 & 0 & 0 & 0 & 1 & 0 \\
    a_{51} & 0 & a_{53} & a_{55} & 0 & a_{57} \\
    0 & 0 & 0 & 0 & 0 & 1 \\
    a_{71} & 0 & a_{73} & 0 & a_{75} & 0 & a_{77}
\end{bmatrix}
$$

(39)

where

$$
a_{11} = -\frac{qS}{mV} C_a \cos \alpha \cos \beta - \frac{g}{V} \sin \gamma + \frac{T}{mV} \cos \alpha \cos \beta + \frac{qS}{mV}(C_{y_\delta} \sin \beta - C_{N_\delta} \sin \alpha \cos \beta)\delta p
$$

$$
-\frac{qS}{mV}(C_{y_\delta} \sin \beta - C_{N_\delta} \sin \alpha \cos \beta)\delta q - \frac{qS}{mV}(C_{y_\delta} \sin \beta - C_{N_\delta} \sin \alpha \cos \beta)\delta r
$$

(40)

$$
a_{13} = -\frac{qS}{m} C_N \sin \alpha \cos \beta
$$

(41)

$$
a_{15} = \frac{qS}{m} C_y \sin \beta
$$

(42)

$$
a_{31} = \frac{g}{V^2} \cos \beta \cos \gamma \cos \mu - \frac{qS}{mV^2} \cos \beta \cos \alpha \cos C_N - \frac{qS}{mV^2} \cos \beta \cos \alpha (C_{y_\delta} \delta p - C_{N_\delta} \delta q - C_{N_\delta} \delta r)
$$

(43)

$$
a_{33} = -\frac{T}{mV \cos \beta} \sin \alpha \cos \beta + \frac{qS}{mV \cos \beta} C_a \sin \alpha
$$

(44)

$$
a_{51} = \frac{qS}{mV^2} C_y \cos \beta + (C_{y_\delta} \cos \beta + C_{N_\delta} \sin \alpha \sin \beta) \delta p - (C_{y_\delta} \cos \beta + C_{N_\delta} \sin \alpha \sin \beta) \delta q
$$

$$
-(C_{y_\delta} \cos \beta + C_{N_\delta} \sin \alpha \sin \beta) \delta r
$$

(45)

$$
a_{53} = \frac{1}{2} \frac{qS}{mV} C_{N_\delta} \sin \alpha \sin \beta
$$

(46)

$$
a_{55} = \frac{1}{2} \frac{qS}{mV} C_{N_\delta} \sin \alpha \sin \beta + (\frac{qS}{mV} C_a - \frac{T}{mV}) \cos \alpha \sin \beta
$$

(47)
\[ a_{57} = \frac{g}{V} \cos \gamma \frac{\sin \mu}{\mu} \]  
(48)

\[ a_{71} = \frac{-\bar{q}S}{mV^2} (C_{Y_0} + C_{Y_y} \delta p - C_{Y_x} \delta q - C_{Y_x} \delta r) \cos \mu \cos \beta \tan \gamma \]  
(49)

\[ a_{73} = \frac{1}{2} \frac{T}{mV} \sin \alpha \sin \mu \tan \gamma + \frac{1}{2} \frac{T}{mV} \sin \alpha \tan \beta - \frac{1}{2} \frac{\bar{q}S}{mV} C_A \frac{\sin \alpha}{\alpha} \sin \mu \tan \gamma \]  
(50)

\[ a_{75} = -\frac{T}{mV} \cos \alpha \frac{\sin \beta}{\beta} \cos \mu \tan \gamma + \frac{T}{2} \frac{\cos \beta}{mV} \sin \alpha \frac{\sin \beta}{\beta} - \frac{1}{2} \frac{\bar{q}S}{mV} C_A \frac{\sin \alpha}{\alpha} \frac{\sin \beta}{\beta} \]  

\[ + \frac{\bar{q}S}{mV \cos \beta} (C_{N_0} + C_{N_y} \delta p - C_{N_y} \delta q - C_{N_y} \delta r) \cos \alpha \frac{\sin \beta}{\beta} + \frac{\bar{q}S}{mV} C_A \cos \alpha \cos \mu \frac{\sin \beta}{\beta} \tan \gamma \]  

\[ + \frac{1}{2} \frac{\bar{q}S}{mV} (C_{N_0} + C_{N_y} \delta p - C_{N_y} \delta q - C_{N_y} \delta r) \sin \alpha \cos \mu \frac{\sin \beta}{\beta} \tan \gamma - \frac{g}{V \cos \beta} \cos \gamma \cos \mu \frac{\sin \beta}{\beta} \]  
(51)

\[ a_{77} = \frac{1}{2} \frac{T}{mV} \sin \alpha \frac{\sin \mu}{\mu} \tan \gamma - \frac{1}{2} \frac{\bar{q}S}{mV} C_A \frac{\sin \mu}{\mu} \tan \gamma \]  

\[ + \frac{\bar{q}S}{mV} (C_{N_0} + C_{N_x} \delta p - C_{N_x} \delta q - C_{N_x} \delta r) \cos \alpha \frac{\sin \mu}{\mu} \tan \gamma \]  
(52)

**Remark 4.1:**

There are numerous ways of factorizing \( f(x) \) as \( A(x)x \). The principle of choosing \( A(x) \) peculiar to this missile problem is to find the terms with sinusoidal functions such as \( \sin \alpha \), \( \sin \beta \) or \( \sin \mu \) and write these terms as \( \sin(*) \) in which * represents \( \alpha, \beta \text{ or } \mu \). In this way we can pull out the state variables from \( f(x) \) and guarantee \( \sin(*) \) is regular when * goes to zero.

The \( \tilde{B}_{ol}(\tilde{x}) \) becomes:
\[
\tilde{B}_{OL}(\tilde{x}) = \begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
-\tan \beta \cos \alpha & 1 & -\tan \beta \sin \alpha \\
\sin \alpha & 0 & -\cos \alpha \\
0 & 0 & 0 \\
\cos \alpha & 0 & \sin \alpha \\
\cos \beta & 0 & \cos \beta
\end{bmatrix}
\] (53)

Since p, q and r would be the control in the outer-loop, the current values of the fin deflections \(\delta p, \delta q,\) and \(\delta r\) are used in this loop during the simulation which is based on Eqs (22)-(25).

### 4.3 \(\theta-D\) Inner-loop Design

The objective in the inner-loop controller is to follow the commands roll rate \(p_c\), pitch rate \(q_c\), and yaw rate \(r_c\), respectively, which are produced in the outer-loop controller.

In the inner-loop design, we would not adopt the integral servo since our major aim is to follow the angular commands \(\alpha, \beta,\) and \(\mu\), respectively, which is realized in the outer-loop controller.

Tracking of the body rate commands p, q and r in the inner loop is used to improve the transient response of the missile.

Consider Eqs. (26)-(28); We notice that there are state-independent terms 
\[
\frac{\bar{q}Sd(C_u \cos \alpha - C_{n_u} \sin \alpha)}{I_x}, \quad \frac{\bar{q}SdC_{m_u}}{I_y} \quad \text{and} \quad \frac{\bar{q}Sd(C_u \sin \alpha + C_{n_u} \cos \alpha)}{I_z}
\] that won’t go to zero when the inner-loop states approach zero. To see this, note that \(V\) and \(\alpha\), the state variables in the outer-loop, are considered as constants in the inner-loop. As a result, \(\bar{q}\) is not a function of any state variable of the inner-loop state space. In order to impose the \(f(0)=0\) assumption, these terms can be manipulated by multiplying and dividing these terms by a state that will never go to zero.
Since $p$, $q$, $r$ and $\delta p$, $\delta q$, $\delta r$ can all go to zero, an additional state $s$ with stable dynamics is added to the state space in order to absorb the biases.

$$\dot{s} = -\lambda_s s \quad (54)$$

At each pass through the inner loop, $s$ is set to its initial value (assumed small).

In the inner-loop, a hard bound of 30 degree is imposed on the fin commands. This is achieved by replacing $\delta p$, $\delta q$, and $\delta r$ in Eqs. (26)-(28) with saturation sine functions. That is:

$$\begin{bmatrix}
\delta p \\
\delta q \\
\delta r
\end{bmatrix} = \begin{bmatrix}
\text{sat}\sin(30, \tilde{\delta} p) \\
\text{sat}\sin(30, \tilde{\delta} q) \\
\text{sat}\sin(30, \tilde{\delta} r)
\end{bmatrix} \quad (55)$$

where the actuator dynamics for $\tilde{\delta} p$, $\tilde{\delta} q$, and $\tilde{\delta} r$ are modeled as first-order linear systems:

$$\tilde{\delta} p = -\lambda_p (\tilde{\delta} p - u_p) \quad (56)$$

$$\tilde{\delta} q = -\lambda_q (\tilde{\delta} q - u_q) \quad (57)$$

$$\tilde{\delta} r = -\lambda_r (\tilde{\delta} r - u_r) \quad (58)$$

and the saturation sine function is defined as

$$\text{sat}\sin(m, z) = \begin{cases} 
  m \ast \sin(z) & \text{for } |z| < \frac{\pi}{2} \\
  m & \text{for } z \geq \frac{\pi}{2} \\
  -m & \text{for } z \leq -\frac{\pi}{2}
\end{cases} \quad (59)$$

Thus, for the inner loop, the state space is given by

$$\ddot{x} = [p \ q \ r \ s \ \tilde{\delta} p \ \tilde{\delta} q \ \tilde{\delta} r]^T \quad u = [u_p \ u_q \ u_r]^T \quad (60)$$

The inner-loop controller becomes
\[
\begin{align*}
\mathbf{u} = -\mathbf{R}_m(\mathbf{x})^{-1} \mathbf{B}_m(\mathbf{x})^T \sum_{n=0}^{\infty} T_{\theta}^n(\mathbf{x}, \theta) \theta^n \begin{bmatrix} x_T - r_c \\ x_N \end{bmatrix}
\end{align*}
\]

where
\[
\begin{align*}
x_T &= [p \quad q \quad r]^T, \quad x_N &= [s \quad \delta p \quad \delta q \quad \delta r]^T
\end{align*}
\]

We follow the same factorization form as Eq. (35). The corresponding inner-loop state-dependent coefficient factorization is given by [22]:

\[
\tilde{\mathbf{A}}_m(\mathbf{x}) = \begin{bmatrix}
 a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} & a_{17} \\
 a_{21} & a_{22} & a_{23} & a_{24} & a_{25} & a_{26} & a_{27} \\
 a_{31} & a_{32} & a_{33} & a_{34} & a_{35} & a_{36} & a_{37} \\
 0 & 0 & 0 & -\lambda_x & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & -\lambda_p & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & -\lambda_q & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & -\lambda_r
\end{bmatrix}
\]

where
\[
\begin{align*}
a_{11} &= \frac{Q_s d^2}{2 I_x V} C_{ip} \cos \alpha \\
a_{12} &= \frac{1}{2} \frac{I_z - I_x}{I_x} r \\
a_{13} &= \frac{1}{2} \frac{I_z - I_x}{I_x} q - \frac{Q_s d^2}{2 I_x V} C_{i}\sin \alpha \\
a_{14} &= \frac{Q_s d}{s I_x} (C_{i}\cos \alpha - C_{n}\sin \alpha) \\
a_{15} &= \frac{Q_s d}{I_x} (C_{i}\cos \alpha - C_{n}\sin \alpha) \frac{\sin \alpha \delta p}{\delta p} \\
a_{16} &= -\frac{Q_s d}{I_x} (C_{i}\cos \alpha - C_{n}\sin \alpha) \frac{\sin \alpha \delta q}{\delta q} \\
a_{17} &= -\frac{Q_s d}{I_x} (C_{i}\cos \alpha - C_{n}\sin \alpha) \frac{\sin \alpha \delta r}{\delta r} \\
a_{21} &= \frac{1}{2} \frac{I_z - I_x}{I_y} r \\
a_{22} &= \frac{Q_s d^2}{2 I_y V} C_{m} \\
a_{23} &= \frac{1}{2} \frac{I_z - I_x}{I_y} \\
a_{24} &= \frac{Q_s d}{s I_y} C_{m} \\
a_{25} &= \frac{Q_s d}{I_y} C_{mp} \frac{\sin \alpha \delta p}{\delta p} \\
a_{26} &= -\frac{Q_s d}{I_y} C_{mq} \frac{\sin \alpha \delta q}{\delta q} \\
a_{27} &= \frac{Q_s d}{I_y} C_{mr} \frac{\sin \alpha \delta r}{\delta r}
\end{align*}
\]
The linear factorization is taken except that \( \frac{\text{sat} \sin(30, \delta p)}{\Delta} \) replaces \( \Delta \) wherever it appears. \( \Delta \) represents \( \delta p, \delta q, \) or \( \delta r. \)

Also

\[
\tilde{B}_n(\tilde{x}) = \begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix}
\]  \( (64) \)

5. Bank-to-Turn/Skid-to-Turn Command Logic

A hybrid BTT/STT autopilot command logic is used to convert the commanded accelerations from the guidance laws to reference angle commands for the autopilot. In the midcourse and terminal phases of a missile flight, the BTT control is employed to prevent any engine flameout. As the missile approaches the endgame phase and passes a preset time-to-go threshold, BTT commands are executed simultaneously with STT commands to improve transient responses. During this interval, the missile is flying as a hybrid BTT/STT.
The BTT mode is broken into three commanded-acceleration magnitude regions in the inertial frame (I) as defined below [22]:

If $a'_c < 0.1g$ then STT control:

$$\mu_c = \mu$$

(65)

If $0.1g \leq a'_c < 1g$ then reduced BTT control:

$$\mu_c = \mu + \left[ \tan^{-1}\left( \frac{-a'_{yc}}{a'_{zc}} \right) - \mu \right] a'_c \frac{1}{1g}$$

(66)

If $a'_c \geq 1g$ then full BTT control:

$$\mu_c = \tan^{-1}\left( \frac{-a'_{yc}}{a'_{zc}} \right)$$

(67)

where $a'_c = [a'_y, a'_z]'$ is the commanded acceleration in the inertial frame.

The idea of choosing command logic this way is to attempt to reduce the effect of noise affecting the controller. When the commanded acceleration is small, it is desired to make the error between commanded angles and actual angles small such that the controller would not respond to the noise if the noise magnitude is comparable to the commanded accelerations. As the commanded accelerations go up, the weight on errors would be increased accordingly. If above 1g, the true desired angle commands are used.

In the full BTT mode:

$$\alpha_c = \frac{\|a_c\| m/\bar{q}S - |C_z - C_{za}\alpha|}{C_{za}}, \quad \beta_c = 0$$

(68)

where $a_c = [a_y, a_z]'$ is the commanded acceleration in the body frame.

In the reduced BTT mode,

$$\alpha_c = \frac{\|a_c\| m/\bar{q}S - |C_z - C_{za}\alpha|}{C_{za}} \cos(\mu_{full} - \mu_c)$$

(69)

where absolute value is used to ensure that angle of attack command for BTT mode is positive.

$$\beta_c = \frac{\|a_c\| m/\bar{q}S - |C_y - C_{ya}\beta|}{C_{ya}} \sin(\mu_{full} - \mu_c)$$

(70)
where  
\[ \mu_{\text{full}} = \tan^{-1}\left( \frac{-a_{\text{y},c}}{a_{\text{z},c}} \right) \]  

(71)

Note that STT is used for small acceleration commands to prevent the missile from performing 180° rolls to achieve insignificant accelerations. STT control is also used in the endgame for quicker response.

In the STT mode:

\[ \alpha_c = \frac{(a_{z,m}/qS) - (C_z - C_{z_{\alpha}}) \alpha}{C_{z_{\alpha}}}, \quad \beta_c = \frac{(a_{y,m}/qS) - (C_y - C_{y_{\beta}}) \beta}{C_{y_{\beta}}}, \quad \mu_c = \mu \]  

(72)

As the missile approaches the endgame phase and passes a preset time-to-go threshold, the STT commands are switched into the BTT commands over a preselected time interval to attenuate transient responses,

\[
\begin{bmatrix}
\alpha_c \\
\beta_c \\
\mu_c
\end{bmatrix}
= \rho
\begin{bmatrix}
\alpha_c \\
\beta_c \\
\mu_c_{\text{STT}}
\end{bmatrix}
+ (1 - \rho)
\begin{bmatrix}
\alpha_c \\
\beta_c \\
\mu_c_{\text{BTT}}
\end{bmatrix}
\]

(73)

where \( \rho \) is a parameter that varies linearly from zero to one over the specified time interval.

6. Simulation and Analysis of the Results

6.1 Simulation setup

Numerical results are obtained using a six degree-of-freedom BTT/STT missile model with a Simulink setup developed in [22]. To ensure that the autopilot only command deflections that are achievable, a 200°/sec hard limit is imposed upon commanded roll rate \( p \), pitch rate \( q \) and yaw rate \( r \) respectively. The fin deflection limit is set to 30°. The actuator dynamics is modeled as a first order system with a time constant of 0.01.
The goal of this study is to design an autopilot that maintains a good response throughout a wide flight envelope. The state weighting on $\alpha, \beta, \alpha, \beta$ and $\beta$ are varied at low, medium and high altitudes until comparable performance is obtained over a range of altitudes. The state weights are chosen according to [22]. They are curve fit to a quadratic function of dynamic pressure, $\bar{q}$, which in turn is a function of missile velocity, one of the states in the outer-loop.

$$\hat{Q}(\tilde{x})_{ol} = diag\{0, q_{22}(\tilde{x}), q_{33}(\tilde{x}), q_{44}(\tilde{x}), q_{55}(\tilde{x}), q_{66}(\tilde{x}), 120\}$$

$$\tilde{R}(\tilde{x})_{ol} = diag\{0.5, 1, 0.5\}$$

(74)

where $diag\{*\}$ represents the diagonal matrix;

$$q_{22}(\tilde{x}) = 11.6742 \times 10^{-11} \bar{q}(\tilde{x})^2 + 2.0182 \times 10^{-5} \bar{q}(\tilde{x}) + 0.6137$$

$$q_{33}(\tilde{x}) = 11.4251 \times 10^{-10} \bar{q}(\tilde{x})^2 - 2.1170 \times 10^{-3} \bar{q}(\tilde{x}) + 105.7985$$

$$q_{44}(\tilde{x}) = q_{22}(\tilde{x}), q_{55}(\tilde{x}) = q_{33}(\tilde{x}), q_{66}(\tilde{x}) = 1.2 * \min\{100,[(\mu - \mu_c)180/\pi]^2 + 0.001\}$$

The inner-loop state and control weightings are chosen in a similar manner to those in the outer-loop design:

$$\tilde{Q}(\tilde{x})_{il} = diag\{16, q_{22}, q_{33}, 0, 0, 0, 0\}$$

$$\tilde{R}(\tilde{x})_{il} = diag\{0.04, 0.3, 0.06\}$$

(75)

where

$$q_{22} = 1.4609 \times 10^{-10} \bar{q}^2 - 1.6816 \times 10^{-3} \bar{q} + 48.7078$$

$$q_{33} = 1.3880 \times 10^{-10} \bar{q}^2 - 1.6573 \times 10^{-7} \bar{q} + 99.5426$$

$\tilde{Q}_0$ and $\tilde{Q}_s(\tilde{x})$ required in the $\theta - D$ method (Eq. (10)) are chosen in the same manner as in the Eq. (35), i.e. $\tilde{Q}_0 = \tilde{Q}(\tilde{x}_0)$ and $\tilde{Q}_s(\tilde{x}) = \tilde{Q}(\tilde{x}) - \tilde{Q}(\tilde{x}_0)$.

The time constants used the Eqs. (54), (56)-(58) are given by

$$\lambda_p = \lambda_q = \lambda_r = \lambda_s = 1$$

(76)

In order to show the effectiveness of the $\theta - D$ autopilot design for a wide flight envelope, the simulation is evaluated at three different flight conditions: The first condition is the initial design.
condition: flight Mach number \( M=2.7 \) and flight altitude \( h=20,000 \text{ft} \). The second and third conditions are at \( M=2.7, h=100 \text{ft} \) (low altitude flight) and \( M=2.7, h=40,000 \text{ft} \) (high altitude flight), respectively. At the start of simulation, a commanded inertial acceleration of \( a_y = 10g, a_z = -10g \) is fed into the system. These values are rotated into the body frame to be used as inputs to the BTT/STT command logic. Since these commanded accelerations are larger than 1g, the missile responded in the BTT mode.

6.2 \( D_i \) design with the \( \theta-D \) controller:

In this simulation, first three terms, i.e. \( T_0, T_1 \) and \( T_2 \), in the control equation (16) are used to compute the necessary control. The simulation results will show that they are sufficient to achieve satisfactory performance. The \( \theta-D \) outer/inner controller was designed based on the first flight condition. It was found that the controller worked very well for the second and third conditions using the same design parameters. The \( D_i \) matrices for the outer-loop controller were chosen as:

\[
D_1 = 0.9 e^{-10t} \left[ -T_0 \bar{A}_{\theta \theta}(\bar{x}) \theta - \bar{A}_{\theta \theta}(\bar{x}) T_0 + T_0 g_0 R^{-1} g'_{\theta} \theta T_0 + T_0 g_{\theta} R^{-1} g'_{\theta} \theta T_0 - \bar{Q}(\bar{x}) \right] 
\]

\[
D_2 = 0.9 e^{-10t} \left[ -T_0 \bar{A}_{\theta \theta}(\bar{x}) \theta - \bar{A}_{\theta \theta}(\bar{x}) T_0 + T_0 g_0 R^{-1} g'_{\theta} \theta T_1 + T_0 g_{\theta} R^{-1} g'_{\theta} \theta T_1 + T_0 g_{\theta} R^{-1} g_{\theta} R^{-1} g'_{\theta} \theta T_0 + T_0 g_{\theta} R^{-1} g_{\theta} R^{-1} g_{\theta} R^{-1} g'_{\theta} \theta T_0 \right] 
\]

\[
+ T_1 g(\bar{x}) R^{-1} g_0 \theta T_0 + T_1 g_{\theta} R^{-1} g_0 \theta T_0 \]

(77)

(78)

Note that in the outer-loop design, \( \bar{B}(\bar{x}) \) is state dependent. Therefore there would be a \( g(\bar{x}) \) term in \( D_1 \) and \( D_2 \). Also recall in Remark 2.3 that \( \theta \) is an intermediate variable that does not affect the results.

The inner-loop controller is designed with
Note that in the inner-loop design, $\tilde{B}(\tilde{x})$ is a constant matrix. So there would be no $g(\tilde{x})$ term in $D_1$ and $D_2$.

As seen in Section 2, $D_i$ matrices play an important role in the $\theta - D$ method design. $k_i$ and $l_i$ in the $D_i$ equations (17)-(19) are the design parameters. As pointed out in Remark 2.2, they can be used to adjust the system performance as well as ensure the stability of the closed-loop system. The selection of $(k_i, l_i)$ is based on the following method.

This approach is based on the observation that the $\theta - D$ approach gives an approximate closed-form solution to the State Dependent Riccati Equation (SDRE)

$$F^T(x)P(x) + P(x)F(x) - P(x)B(x)R^{-1}B^T(x)P(x) + Q = 0$$

where $F(x) = A_0 + A(x)$.

To see this, assume that

$$\frac{\partial V}{\partial x} = P(x)x$$

and $P(x)$ is symmetric. Substituting Eq. (82) into the HJB equation (3) and writing nonlinear $f(x)$ in a linear like structure $f(x) = F(x)x = [A_0 + A(x)]x$ leads to the state dependent Riccati equation (81). This is in fact the idea of the recently emerging State Dependent Riccati Equation (SDRE) technique [16]. Compared with the SDRE approach, the $\theta - D$ method solves a perturbed HJB equation (10) and assumes that

$$\frac{\partial V}{\partial x} = \sum_{i=0}^{N} T_i(x, \theta)\theta'x$$

As can be seen, the $\theta - D$ method is similar to the SDRE approach in the sense that both bring the nonlinear equation into a linear-like structure $f(x) = F(x)x$ and solve the optimal control problem. The former gives an approximate closed-form solution while the latter solves the
algebraic Riccati equation at each state. It can be predicted that the $\theta - D$ solution is close to the SDRE solution, i.e.

$$P(\theta - D) \approx P(SDRE)$$

where $P(\theta - D) = \sum_{i=0}^{\infty} T_i(x, \theta) \theta^i$ is obtained from solving Eqs. (12)-(15) offline and $P(SDRE)$ is obtained by solving Eq. (81) online.

An efficient procedure for finding the $(k_i, l_i)$ was determined as follows: The SDRE controller is used to generate a state trajectory and then the maximum singular value of $P(SDRE)$, i.e. $\sigma_{\text{max}}(P(SDRE))$, is computed at each state point. Similarly, the $(k_i, l_i)$ parameters determine $P(\theta - D)$ and its associated $\sigma_{\text{max}}(P(\theta - D))$. Curve fits are then applied to $\sigma_{\text{max}}(P(SDRE))$ and $\sigma_{\text{max}}(P(\theta - D))$, and the $(k_i, l_i)$ are selected to minimize the difference between these singular value histories in a least squares sense. The $(k_i, l_i)$ can all be determined in one least squares run off-line.

**6.3. Numerical results and analysis**

Figure 2 shows the performance of the outer/inner loop autopilot at the first flight condition. The three figures on the left column of Figure 2 demonstrate the outer-loop tracking, i.e. $\alpha_c, \beta_c$ and $\mu_c$. The right column shows the inner-loop tracking, i.e. $p_c, q_c$ and $r_c$. Likewise, the results for the second and third flight conditions are presented in Figure 3 and Figure 4, respectively. As can be seen, both the outer-loop and inner-loop controllers provide excellent tracking response over a large region of the operating envelope.

Figures 5-7 show the required fin deflections for these three cases. It can be seen that the fin deflections are all well-behaved. Since the missile is flying in the BTT mode, the sideslip should be as small as possible. From the results of these three altitude simulations it can be observed that the sideslip angle is kept at less than 1 degree throughout. It can also be seen that the settling
time of the tracking responses at 40,000ft is longer than the other two. This is due to the variation of the air density at high altitudes. Lower density at higher altitude leads to reduce aerodynamic forces and moments. The same commanded accelerations need bigger aerodynamic angles to achieve the necessary forces and moments. Therefore, the control response at 40,000ft shows larger fin deflections compared to lower altitude cases. At 100ft altitude, the missile goes through lowest angle of attack, sideslip angle and control effort. Figure 8 presents the autopilot tracking of the acceleration commands coming from the guidance law. As can be seen, they are tracked very well in all three altitudes.

To evaluate the sensitivity of the autopilot design to speed, the autopilot designed at M=2.7, h=20,000ft is also evaluated at two other different Mach numbers, M=2.0 and M=3.2. Figures 9-11 show the tracking responses and achieved fin deflections at these two Mach numbers, respectively.

It can be seen that the autopilot still performs very well. Note that at the higher Mach number the required angle of attack and control level are smaller than those at the lower Mach number. The reason is that aerodynamic coefficients associated with the aerodynamic angles and fin deflections are larger when the Mach number is higher. Also, at a given altitude, the dynamic pressure is higher if the Mach number is higher. Thus for the same force commands, they need less aerodynamic angles $\alpha$ and $\beta$ and fin deflections. Since the missile is flying at the BTT mode, the bank angle should be kept constant. Figure 12 demonstrates the comparison of bank angle response at all the five different flight conditions. It can be seen that the autopilot performs very well in all these cases.

From the simulations, several advantages of the $\theta-D$ method can be observed. First, the $\theta-D$ outer/inner loop controllers show excellent tracking characteristics and are insensitive in
operating flight conditions. Second, the $\theta - D$ based autopilot design is based on optimal control of a cost function. It allows the designer to consider the tracking performance and control energy in one integrated design. The third important advantage of the $\theta - D$ method is that it is available in closed-form expression if we take finite terms in Eq. (16). Compared with the SDRE technique [22], the $\theta - D$ method does not need excessive on-line computations. This implementation advantage is more significant in the missile autopilot design problem since it involves $7 \times 7$ matrices in both outer and inner loop designs. Solving $7 \times 7$ Riccati equations on-line would be very time-consuming. The final remark is that the tuning of design parameters in $D_i$ matrices is not an involved process. Usually a diagonal matrix is picked and multiplied by the state dependent terms which are given in Eqs. (17)-(19). The numerical experiment also shows that system performance is not sensitive to the variations around the values of $k_i$ and $l_i$ in $D_i$.

7. Conclusions

In this paper, a new nonlinear suboptimal control synthesis technique, $\theta - D$ method, has been used to design a hybrid Bank-to-Turn/Skid-to-Turn autopilot for a generic air-to-air missile. We formulate this autopilot design as an optimal control problem. An outer loop and an inner loop control structures are used. The outer-loop converts commanded angle-of-attack, sideslip, and bank angle to body rate commands for the inner loop. The inner loop converts the body rate commands to fin deflection commands. The simulation results demonstrate excellent tracking performance and insensitivity to flight conditions over a wide flight envelope. The $\theta - D$ design gives a closed-form solution to this nonlinear optimal control problem and is easy to implement.

Acknowledgement

Grant from Anteon Corporation in support of this study for Naval Surface Warfare Center is gratefully acknowledged.
Reference:


Figure 2: (\(M=2.7, h=20,000\text{ft}\)) Tracking responses of \(\alpha, \beta, \mu, p, q, r\)
Figure 3: (M=2.7, h=100ft) Tracking responses of $\alpha$, $\beta$, $\mu$, $p$, $q$, $r$
Figure 4: (M=2.7, h=40,000ft) Tracking responses of $\alpha, \beta, \mu, p, q, r$
Figure 5: (M=2.7, h=20,000ft) Achieved fin deflections

Figure 6: (M=2.7, h=100ft) Achieved fin deflections

Figure 7: (M=2.7, h=40,000ft) Achieved fin deflections

Figure 8: Total body YZ acceleration tracking at different altitudes
Figure 9: (M=2.0, h=20,000ft) tracking response of $\alpha, \beta, \mu, p, q, r$
Figure 10: (M=3.2, h=20,000ft) Tracking response of $\alpha, \beta, \mu, p, q, r$
Figure 11: (h=20,000ft) Fin deflection comparison at different Mach numbers
Figure 12: Bank angle comparison at different flight conditions