Dynamic Conversion of Flight Path Angle Commands to Body Attitude Commands

James R. Cloutier
U.S. Air Force Research Laboratory
Munitions Directorate
101 W. Eglin Blvd., Ste. 341
Eglin AFB, FL 32542-6810
donald.stansbery@eglin.af.mil

Donald T. Stansbery
CACI/TEAS Group
101 W. Eglin Blvd., Ste. 341
Eglin AFB, FL 32542-6810
donald.stansbery@eglin.af.mil

Abstract

A state-dependent Riccati equation (SDRE) based controller with integral servomechanism tracking is designed to convert flight path angle commands to angle-of-attack and bank angle commands for a bank-to-turn air vehicle. This problem is challenging because the controls are highly nonlinear, appearing in the dynamics as products of sines and cosines.

1. Introduction

Today there is an emphasis on real-time trajectory optimization for both civil and military purposes. Increased microprocessor execution speed and storage combined with rapid optimization techniques [1, 2, 3] are making real-time trajectory optimization a reality. If an optimal trajectory is computed as a parameterized, continuously differentiable ($C^1$) function or, if a discrete-time optimal trajectory is curve-fitted as a parameterized, continuously differentiable ($C^1$) function, then the function can be differentiated to provide instantaneous flight path angle reference commands. In order for an air vehicle to follow the optimized trajectory, the flight path angle commands need to be converted to body attitude commands. In this paper, a state-dependent Riccati equation-based controller is designed to convert the flight path angle commands to angle-of-attack and bank angle commands. The system dynamics are presented in the next section. An overview of the SDRE technique with integral servomechanism tracking is given in Section 3. The SDRE design is carried out in Section 4. The design is then evaluated using a detailed six-degrees-of-freedom simulation which employs an SDRE-based autopilot and the results are presented in Section 5. The paper is then closed with a Summary section.

2. System Dynamics

The flight path angle system dynamics are given by:

$$\dot{\gamma} = -\frac{g}{V} \cos \gamma + \frac{T}{mV} (\sin \alpha \cos \mu + \cos \alpha \sin \beta \sin \mu)$$

$$- \frac{dS}{mV} C_A (\sin \alpha \cos \mu + \cos \alpha \sin \beta \sin \mu)$$

$$- \frac{dS}{mV} C_Y \cos \beta \sin \mu$$

$$+ \frac{dS}{mV} C_N (\cos \alpha \cos \mu - \sin \alpha \sin \beta \sin \mu)$$

$$\dot{\chi} = \frac{T}{mV \cos \gamma} (\sin \alpha \sin \mu - \cos \alpha \sin \beta \cos \mu)$$

$$- \frac{dS}{mV \cos \gamma} A (\sin \alpha \sin \mu - \cos \alpha \sin \beta \cos \mu)$$

$$+ \frac{dS}{mV \cos \gamma} C_Y \cos \beta \cos \mu$$

$$+ \frac{dS}{mV \cos \gamma} C_N (\cos \alpha \sin \mu + \sin \alpha \sin \beta \cos \mu)$$

where $T$ is thrust, $m$ is the vehicle's mass, $V$ is the vehicle's speed, $g$ is dynamic pressure, $S$ is the reference area, $\mu$ is the angle of attack, $\gamma$ is the vertical flight angle, $C_A$, $C_Y$, and $C_N$ are the aerodynamic axial force, side force, and normal force coefficients, respectively; $\gamma$ is the vertical flight angle.
path angle, $\chi$ is the horizontal flight path angle, $\alpha$ is the angle-of-attack, $\beta$ is the sideslip angle, and $\mu$ is the bank angle.

The problem is to convert the flight path angle commands $\{-\gamma_0, \chi_0\}$ to angle-of-attack and bank angle commands $\{\alpha, \mu\}$, while issuing the sideslip command $\beta = 0$ for the vehicle's bank-to-turn autopilot. The controls in the problem are $\alpha$ and $\mu$, which are equivalently $\alpha$ and $\mu$, respectively. In Eqs. (1)-(2), $\beta$ is the actual sideslip angle and not $\beta_0$. It should be noted that the above problem is not an easy one, owing to the fact that it is highly nonlinear in the controls, with $\alpha$ and $\mu$ appearing as the products of sines and cosines.

3. The SDRE Method

Consider the autonomous, infinite-horizon, nonlinear regulator problem for minimizing the performance index

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

with respect to the state $x$ and control $u$ subject to the nonlinear differential constraints:

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

where $Q(x) \geq 0$ and $R(x) > 0$ for all $x$ and $f(0) = 0$.

The SDRE approach for obtaining a suboptimal, locally asymptotically stabilizing solution of problem (3)-(4) is:

i) Use direct parameterization to bring the nonlinear dynamics to the state-dependent coefficient (SDC) form

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

where

$$f(x) = A(x)x$$

In the multivariable case, it is well-known [4, 5] that if $f(x)$ is a continuously differentiable function of $x$, there is an infinite number of ways to factor $f(x)$ into $A(x)x$ and that $A(x)$ can be parameterized as $A(x, \bar{\alpha})$, where $\bar{\alpha}$ is a vector of free design parameters. In order to obtain a valid solution of the SDRE, the pair $\{A(x, \bar{\alpha}), B(x)\}$ has to be pointwise stabilizable in the linear sense for all $x$ in the domain of interest.

ii) Solve the state-dependent Riccati equation

$$A^TP + PA - PB^TBP + Q = 0$$

to obtain $P(x) \geq 0$.

iii) Construct the nonlinear feedback controller equation:

$$u = -R(x)^{-1}B(x)^TP(x)x.$$  (8)

In order to perform command following, the SDRE controller can be implemented as an integral servomechanism as demonstrated in [6]. This is accomplished as follows. First, the state $x$ is decomposed as

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

where it is desired for the vector components of $x_T$ to track a reference command $r_e$. The state vector $x$ is then augmented with $x_I$, the integral states of $x_T$:

$$\bar{x} = \begin{bmatrix} x_I \\ x_T \\ x_N \end{bmatrix}.$$  (10)

The augmented system is given by

$$\dot{\bar{x}} = \bar{A}(\bar{x}, \bar{\alpha})\bar{x} + \bar{B}(\bar{x})u$$

where

$$\bar{A}(\bar{x}, \bar{\alpha}) = \begin{bmatrix} 0 & I & 0 \\ 0 & A(x, \bar{\alpha}) & 0 \end{bmatrix}, \quad \bar{B}(\bar{x}) = \begin{bmatrix} 0 \\ B(x) \end{bmatrix}.$$  (12)

and the SDRE integral servocommander is given by

$$u = -\bar{R}(\bar{x})^{-1}\bar{B}(\bar{x})^TP(\bar{x})\begin{bmatrix} x_I - \int r_e dt \\ x_T - r_e \\ x_N \end{bmatrix}.$$  (13)

In order for the SDRE to have a solution, the pointwise detectability condition must be satisfied. This is accomplished by penalizing the integral states with the corresponding non-zero diagonal elements of $Q(\bar{x})$.

4. SDRE Control Design

In order to handle the nonlinearity of the controls in Eqs. (1)-(2) and to bring the system to a form compatible with the requirements of the SDRE technique, integral control is employed. The design state space is given by

$$z = \begin{bmatrix} \gamma \\ \chi \\ x_T \\ x \\ \alpha \\ \mu \end{bmatrix}, \quad u = \begin{bmatrix} \dot{\alpha} \\ \dot{\mu} \end{bmatrix}.$$  (14)
where \( \gamma_t \) and \( \chi_t \) are the integral states of \( \gamma \) and \( \chi \), respectively. Since both \( \gamma \) and \( \chi \) can be zero, a stable state \( z \) with dynamics

\[
\dot{z} = -\lambda z
\]

where \( \lambda > 0 \) has been added to the state vector. As will be seen, this is done for the purpose of handling state-independent terms which will arise when the cosine terms in Eqs. (1)-(2) are shifted to the origin to allow for their state-dependent coefficient factorization [7]. For example, consider the first term in Eq. (1), \(-g \cos \gamma/V\). As \( z \) cannot be factored out of this term because \( \frac{\cos \gamma}{\gamma} \) goes to infinity as \( \gamma \to 0 \). However, we can write

\[
-\frac{g}{V} \cos \gamma = -\frac{g}{V} (\cos \gamma - 1) = -\frac{g}{V \cdot \frac{\cos \gamma - 1}{\gamma}} \cdot \frac{\frac{\gamma}{V^2}}{z}.
\]

The term \( (\cos \gamma - 1)/\gamma \) is well-behaved and goes to zero as \( \gamma \to 0 \). The resulting bias term \(-g/V\) is then factored by multiplying and dividing by \( z \). The value of \( z \) is reset to its initial value each time through the controller so the factor \(-g/(Vz)\) is well-behaved. In generating the state-dependent coefficient factor \( A(x, \theta) \), all of the cosine terms in Eqs. (1)-(2) were shifted to the origin in this way. Additionally, the \( \cos \gamma \) factors appearing in the denominators of the terms in the \( \chi \)-dynamics (2) were rewritten as \( \sec \gamma \) factors in the numerators of these terms. The \( \sec \gamma \) factors were then shifted to the origin. Finally, any term containing more than one state variable was parameterized [5] and apportioned among the corresponding elements of the \( A(x, \theta) \) matrix. The structure of the \( A(x, \theta) \) matrix is given in the Appendix.

6. Simulation Results

An investigation of the performance of the controller has been conducted using a detailed six-degrees-of-freedom simulation. The simulation uses the SDRE-based autopilot developed in [6]. The autopilot has a two-loop structure, with the new flight path angle servo forming a third loop as shown in Figure 1. The flight path angle servo converts the command flight angles \( \gamma_e, \chi_e \) to angle-of-attack and bank angle commands, \( \alpha_e, \mu_e \), with the sideslip command set to \( \beta_e = 0 \). The autopilot’s outer loop converts \( \alpha_e, \beta_e, \mu_e \) to roll-rate, pitch-rate, and yaw-rate commands, \( \dot{p}_e, \dot{q}_e, \dot{r}_e \), for the inner loop. The autopilot’s inner loop then converts \( p_e, q_e, r_e \) to fin commands for the actuators.

The simulation imposes rate limits of 500°/sec on \( p \) and 200°/sec on \( q \) and \( r \). The fin deflection limit is 30° and the actuator dynamics have a time constant of 0.1. The vehicle presently being flown in the simulation is a generic air-to-air missile. Our intent is to eventually replace the missile with a winged vehicle to support uninhabited combat aerial vehicle (UCAV) research.

The state and control weighting matrices chosen for the flight path angle servo were:

\[
Q(x) = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 100 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 100 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 100 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix},
\]

and

\[
R(x) = \begin{bmatrix}
.1 & 0 \\
0 & .1
\end{bmatrix},
\]

where

\[
q_1(x) = \min \left\{ 10, \frac{180}{\pi} (\gamma - \gamma_0)^2 + .001 \right\}
\]

\[
q_2(x) = \min \left\{ 10, \frac{180}{\pi} (\chi - \chi_0)^2 + .001 \right\}
\]

\[
q_6(x) = \frac{1}{\text{abs} \left( \frac{180}{\pi} \alpha - 25 \right) + \epsilon_1},
\]

\[
q_7(x) = \frac{1}{\text{abs} \left( \frac{180}{\pi} \mu - 90 \right) + \epsilon_2},
\]

and where \( \epsilon_1, \epsilon_2 \) are small numbers and \( N_1, N_2 \geq 1 \). These parameters were selected as \( \epsilon_1 = \epsilon_2 = .01 \) and \( N_1 = N_2 = 6 \). The state-dependent weightings \( q_1(x) \) and \( q_2(x) \), imposed on the integral states \( \gamma_t \) and \( \chi_t \), enhance the rise time and reduce the overshoot of \( \gamma \) and \( \chi \), respectively. The state-dependent weightings \( q_6(x) \) and \( q_7(x) \) place soft bounds on \( \alpha \) and \( \mu \) as they near \( \pm 25^\circ \) and \( \pm 90^\circ \), respectively [8, 9]. To desensitize the controller to large differences between the flight path angle commands and their actual values, especially between \( \chi_e \) and \( \chi_t \), the differences \( \gamma_e - \gamma_t \) and \( \chi_e - \chi_t \) were passed through a saturation function with an upper and lower limit of 4 degrees before being input to the controller.

The three-loop control system was evaluated using the initial flight condition \( \text{Mach (M)} = \)
1.8, Altitude \( (h) = 20000 \text{ ft} \). The simulation was initiated with \( \gamma = \chi = \alpha = \beta = \mu = 0 \) and with \( p = q = r = 0 \). Initial flight path angle commands were zero; at three seconds into the simulation, the flight path angle commands were changed to \( \gamma_e = 2^\circ \) and \( \chi_e = 15^\circ \). Figures 2 and 3 show the tracking of the flight path angle commands. Figures 4-6 show the autopilot's outer-loop tracking of \( \alpha_e, \beta_e, \mu_e \) and Figure 7 shows the autopilot's inner-loop tracking of \( \rho_e \). As can be seen, the controllers in all three loops are well-behaved and provide excellent tracking for the given flight condition.

Figure 4: Commanded and Achieved Angles of Attack \( (M = 1.8, h = 20K\text{ ft}) \)

Figure 2: Commanded and Achieved Vertical Flight Path Angles \( (M = 1.8, h = 20K\text{ ft}) \)

Figure 5: Commanded and Achieved Angles of Sideslip \( (M = 1.8, h = 20K\text{ ft}) \)

Figure 3: Commanded and Achieved Horizontal Flight Path Angles \( (M = 1.8, h = 20K\text{ ft}) \)

Figure 6: Commanded and Achieved Bank Angles \( (M = 1.8, h = 20K\text{ ft}) \)
6. Summary

The state-dependent Riccati equation control method has been used to design a controller to dynamically convert flight path angle commands to body attitude commands. This conversion problem is a challenging one since the controls, angle-of-attack and bank angle, are highly nonlinear, appearing in the dynamics as the products of sines and cosines. This nonlinearity was transferred from the controls to the augmented state vector by introducing integral control. Integral servomechanisms were then employed to achieve unbiased tracking of the flight path angle commands.

References


Appendix

State-Dependent Coefficient Factorization

The coefficient matrix $A(z, \alpha)$ has the structure

$$
A(z, \alpha) = 
\begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & a_{23} & 0 & 0 & a_{26} & a_{28} & a_{27} \\
0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & -\lambda & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

The expressions for $a_{23}, a_{26}, a_{28}, a_{42}, a_{45}, a_{46}, a_{47}$ are functions of the state $z$ and the free design parameter vector $\alpha$ contains twenty-one elements. For the design, all twenty-one elements were set to the value .5.

If $a_{47} = 0$, then the pair $\{A(z, \alpha), B(z)\}$ is not pointwise controllable so a legitimate Riccati equation solution cannot be obtained. This condition occurs at $t = 0$, since the simulation is initialized with all of the states in each of the three loops set to zero and with the fins in their neutral position. This singularity can be avoided by setting $a_{47} = .00001$ whenever $a_{47} = 0$. 

![Figure 7: Commanded and Achieved Roll Rates ($M = 1.8, h = 20K ft$)](image-url)