

Adaptive Output Feedback Control of the NASA GTM Model with Unknown Nonminimum-Phase Zeros

Anthony M. D'Amato*, E. Dogan Sumer†, Kenny S. Mitchell‡,
 Alexey V. Morozov§, Jesse B. Hoagg¶, and Dennis S. Bernstein||

University of Michigan, 1320 Beal Ave., Ann Arbor, MI 48109

In this paper we apply retrospective cost adaptive control (RCAC) to the NASA Generic Transport Model (GTM). The goal is to demonstrate command following and disturbance rejection techniques using a minimum amount of modeling information. In previous work, RCAC was applied to GTM using knowledge of the nonminimum-phase zeros, the relative degree, and the first non-zero Markov parameter associated with the linearized flight dynamics. In this paper, we further reduce the modeling requirements of RCAC by demonstrating that only a limited number of Markov parameters are needed to control the GTM model. We first demonstrate the method on illustrative single-input, single-output (SISO) and multi-input, multi-output (MIMO) problems. Next we demonstrate the method on linearized SISO and MIMO GTM models, and finally, the fully nonlinear GTM model.

I. Introduction

One of the challenges in adaptive control is to develop reliable, easily implementable algorithms that apply to a wide range of plants. Adaptive control algorithms tune themselves to the actual plant, and thus they can overcome the need to sacrifice performance for robustness as in the case of robust control. However, the price paid for this feature is restrictive assumptions, such as full state feedback, positive realness, minimum phase zeros, matched disturbances, as well as information on the sign of the high frequency gain, relative degree, or zero locations.¹⁻⁴ In particular, the retrospective cost adaptive control (RCAC) approach developed in,⁵⁻¹⁵ which is applicable to MIMO (output feedback) plants that are possibly unstable and nonminimum phase (NMP) with uncertain command and disturbance spectra, requires knowledge of the first nonzero Markov parameter as well as locations of the NMP zeros, if any. Alternatively, a collection of Markov parameters can be used as long as a sufficient number is available to capture the NMP zero locations.

The companion paper¹⁶ extends prior RCAC results by describing a modification of RCAC that does not require knowledge of the locations of the NMP zeros. This extension requires knowledge of a limited number of Markov parameters; typically only one Markov parameter is needed. The significant aspect of this extension is the fact that knowledge of the NMP zeros is not needed. This extension thus increases the applicability of the method to systems with unknown NMP zeros, as well as systems with NMP zeros that may be changing slowly due to aging or due to a slowly varying linearization of a nonlinear plant.

In this paper we apply the extended RCAC adaptive control algorithm to a collection of SISO and MIMO systems in order to investigate its ability to adaptively control systems without knowledge of the NMP zeros, if any. To fully understand the features of this algorithm, we consider systems that are square, tall, or wide, which may be stable or unstable, and minimum phase or nonminimum phase, where, for nonsquare systems, we consider the case of NMP zeros of single-channel transfer functions. In addition, we

*NASA GSRP Fellow, Aerospace Engineering Department

†Graduate Student, Aerospace Engineering Department

‡Graduate Student, Aerospace Engineering Department

§Graduate Student, Aerospace Engineering Department

¶Professor, Mechanical Engineering Department, University of Kentucky

||Professor, Aerospace Engineering Department

are especially interested in the wide case in which the Markov parameters are rank deficient. In this case, we show that the rank deficiency, and thus lack of left invertibility, can be overcome by using a sufficient number of Markov parameters to ensure left invertibility of the regressor matrix. In all cases, the number of Markov parameters that are used is not sufficient to capture the NMP zeros of the system. Consequently, these examples demonstrate the ability to control SISO or MIMO NMP systems with unknown NMP zeros.

In addition to these MIMO examples, we apply the extended RCAC algorithm to the NASA GTM model. Unlike the results of,¹⁵ the results given in the present paper require no knowledge of the NMP zero in the pitch transfer function. Consequently, the modeling requirements are significantly alleviated, as is the need for prior or online modeling to determine this modeling information. In effect, the adaptive controller is robust to the presence or location of NMP zeros in the linearized aircraft model.

II. Problem Formulation

Consider the MIMO discrete-time system

$$x(k+1) = Ax(k) + Bu(k) + D_1w(k), \quad (1)$$

$$y(k) = Cx(k) + D_2w(k), \quad (2)$$

$$z(k) = E_1x(k) + E_0w(k), \quad (3)$$

where $x(k) \in \mathbb{R}^n$, $y(k) \in \mathbb{R}^{l_y}$, $z(k) \in \mathbb{R}^{l_z}$, $u(k) \in \mathbb{R}^{l_u}$, $w(k) \in \mathbb{R}^{l_w}$, and $k \geq 0$. The system (1)–(3) can represent a sampled-data application arising from a continuous-time system with sample and hold operations with sample interval T_s , where $x(k)$ represents $x(kT_s)$. The open-loop system (1)–(3) is described by

$$\begin{bmatrix} z \\ y \end{bmatrix} = G(z) \begin{bmatrix} w \\ u \end{bmatrix},$$

where

$$G(z) = \begin{bmatrix} G_{zw}(z) & G_{zu}(z) \\ G_{yw}(z) & G_{yu}(z) \end{bmatrix},$$

and

$$G_{zw}(z) = E_1(zI - A)^{-1}D_1 + E_0,$$

$$G_{zu}(z) = E_1(zI - A)^{-1}B,$$

$$G_{yw}(z) = C(zI - A)^{-1}D_1 + D_2,$$

$$G_{yu}(z) = C(zI - A)^{-1}B.$$

Consider the LTI output feedback controller

$$x_c(k+1) = A_c x_c(k) + B_c y(k), \quad (4)$$

$$u(k) = C_c x_c(k), \quad (5)$$

where $x_c(k) \in \mathbb{R}^{n_c}$. The feedback control (4)–(5) is described by

$$u = G_c(z)y,$$

where

$$G_c(z) = C_c(zI - A_c)^{-1}B_c.$$

The closed-loop system with output feedback (4)–(5) is thus given by

$$\tilde{x}(k+1) = \tilde{A}\tilde{x}(k) + \tilde{D}_1w(k), \quad (6)$$

$$y(k) = \tilde{C}\tilde{x}(k) + D_2w(k), \quad (7)$$

$$z(k) = \tilde{E}_1\tilde{x}(k) + E_0w(k), \quad (8)$$

where

$$\begin{aligned}\tilde{A} &= \begin{bmatrix} A & BC_c \\ B_c C & A_c \end{bmatrix}, \tilde{D}_1 = \begin{bmatrix} D_1 \\ B_c D_2 \end{bmatrix}, \\ \tilde{C} &= \begin{bmatrix} C & 0_{l_y \times n_c} \end{bmatrix}, \tilde{E}_1 = \begin{bmatrix} E_1 & 0_{l_z \times n_c} \end{bmatrix},\end{aligned}\quad (9)$$

and $\tilde{x}(k) = \begin{bmatrix} x(k) \\ x_c(k) \end{bmatrix}$. The closed-loop system (6)–(8) is described by

$$\begin{bmatrix} z \\ y \end{bmatrix} = \tilde{G}(z)w,$$

where

$$\tilde{G}(z) = \begin{bmatrix} \tilde{G}_{zw}(z) \\ \tilde{G}_{yw}(z) \end{bmatrix}, \quad (10)$$

and

$$\tilde{G}_{zw}(z) = \tilde{E}_1(zI - \tilde{A})^{-1}\tilde{D}_1 + E_0, \quad (11)$$

$$\tilde{G}_{yw}(z) = \tilde{C}(zI - \tilde{A})^{-1}\tilde{D}_1 + D_2. \quad (12)$$

The goal is to develop an adaptive output feedback controller that minimizes the performance variable z in the presence of the exogenous signal w with limited modeling information about the dynamics. A block diagram of the feedback system is shown in Figure 1. The components of the signal w can represent either command signals to be followed, external disturbances to be rejected, or both.

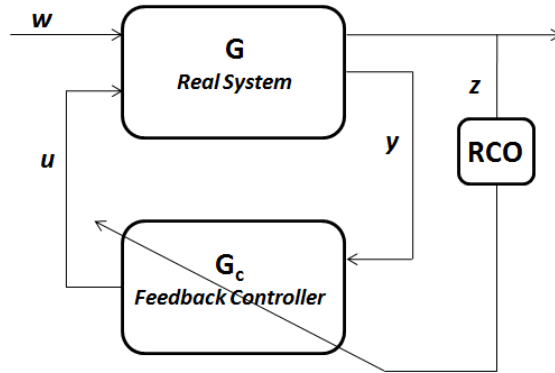


Figure 1. Disturbance-rejection and command-following architecture.

The above problem formulation includes command-following and disturbance-rejection architectures as special cases. If $D_1 = 0$ and $E_0 \neq 0$, then the objective is to have the output $E_1 x$ follow the command signal $-E_0 w$. On the other hand, if $D_1 \neq 0$ and $E_0 = 0$, then the objective is to reject the disturbance w from the performance measurement $E_1 x$. Furthermore, if $D_1 = \begin{bmatrix} \hat{D}_1 & 0 \end{bmatrix}$, $E_0 = \begin{bmatrix} 0 & \hat{E}_0 \end{bmatrix}$, and $w(k) = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$, then the objective is to have $E_1 x$ follow the command $-\hat{E}_0 w_2$ while rejecting the disturbance w_1 . Lastly, if D_1 and E_0 are empty matrices, then the objective is output stabilization, that is, convergence of z to zero.

For the adaptive system, $A_c = A_c(k)$, $B_c = B_c(k)$, and $C_c = C_c(k)$ are time varying, and thus the transfer function model (10)–(12) is not valid during controller adaptation. However, (6)–(8) illustrates the structure of the time-varying closed-loop system in which $\tilde{A} = \tilde{A}(k)$ and $\tilde{D}_1 = \tilde{D}_1(k)$. To monitor the ability of the adaptive controller to stabilize the system, we compute the spectral radius $\text{spr}(\tilde{A}(k))$ of $\tilde{A}(k)$ at each time step. If $\text{spr}(\tilde{A}(k))$ converges to a number less than 1, then the asymptotic closed-loop system is internally stable.

III. Retrospective Surrogate Cost

For $i \geq 1$, define the Markov parameter

$$H_i \triangleq E_1 A^{i-1} B. \quad (13)$$

For example, $H_1 = E_1 B$ and $H_2 = E_1 A B$. Let r be a positive integer. Then, for all $k \geq r$,

$$x(k) = A^r x(k-r) + \sum_{i=1}^r A^{i-1} B u(k-i) + \sum_{i=1}^r A^{i-1} D_1 w(k-i), \quad (14)$$

and thus

$$z(k) = E_1 A^r x(k-r) + \sum_{i=1}^r E_1 A^{i-1} D_1 w(k-i) + E_0 w(k) + \bar{H} \bar{U}(k-1), \quad (15)$$

where

$$\bar{H} \triangleq \begin{bmatrix} H_1 & \dots & H_r \end{bmatrix} \in \mathbb{R}^{l_z \times r l_u}$$

and

$$\bar{U}(k-1) \triangleq \begin{bmatrix} u(k-1) \\ \vdots \\ u(k-r) \end{bmatrix}.$$

Next, we rearrange the columns of \bar{H} and the components of $\bar{U}(k-1)$ and partition the resulting matrix and vector so that

$$\bar{H} \bar{U}(k-1) = \mathcal{H}' U'(k-1) + \mathcal{H} U(k-1), \quad (16)$$

where $\mathcal{H}' \in \mathbb{R}^{l_z \times (r l_u - l_U)}$, $\mathcal{H} \in \mathbb{R}^{l_z \times l_U}$, $U'(k-1) \in \mathbb{R}^{r l_u - l_U}$, and $U(k-1) \in \mathbb{R}^{l_U}$. Then, we can rewrite (15) as

$$z(k) = \mathcal{S}(k) + \mathcal{H} U(k-1), \quad (17)$$

where

$$\mathcal{S}(k) \triangleq E_1 A^r x(k-r) + \sum_{i=1}^r E_1 A^{i-1} D_1 w(k-i) + E_0 w(k) + \mathcal{H}' U'(k-1). \quad (18)$$

For example, $\bar{H} = \begin{bmatrix} H_1 & H_2 & H_3 & H_4 & H_5 \end{bmatrix}$,

$$\mathcal{H}' = \begin{bmatrix} H_1 & H_2 & H_4 \end{bmatrix}, \quad U'(k-1) = \begin{bmatrix} u(k-1) \\ u(k-2) \\ u(k-4) \end{bmatrix},$$

and

$$\mathcal{H} = \begin{bmatrix} H_3 & H_5 \end{bmatrix}, \quad U(k-1) = \begin{bmatrix} u(k-3) \\ u(k-5) \end{bmatrix}.$$

Next, for $j = 1, \dots, s$, we rewrite (17) with a delay of k_j time steps, where $0 \leq k_1 \leq k_2 \leq \dots \leq k_s$, in the form

$$z(k - k_j) = \mathcal{S}_j(k - k_j) + \mathcal{H}_j U_j(k - k_j - 1), \quad (19)$$

where (18) becomes

$$\begin{aligned} \mathcal{S}_j(k - k_j) \triangleq & E_1 A^r x(k - k_j - r) + \sum_{i=1}^r E_1 A^{i-1} D_1 w(k - k_j - i) + E_0 w(k - k_j) \\ & + \mathcal{H}'_j U'_j(k - k_j - 1) \end{aligned} \quad (20)$$

and (16) becomes

$$\bar{H}\bar{U}(k - k_j - 1) = \mathcal{H}'_j U'_j(k - k_j - 1) + \mathcal{H}_j U_j(k - k_j - 1), \quad (21)$$

where $\mathcal{H}'_j \in \mathbb{R}^{l_z \times (r l_u - l_{U_j})}$, $\mathcal{H}_j \in \mathbb{R}^{l_z \times l_{U_j}}$, $U'_j(k - k_j - 1) \in \mathbb{R}^{r l_u - l_{U_j}}$, and $U_j(k - k_j - 1) \in \mathbb{R}^{l_{U_j}}$. Now, by stacking $z(k - k_1), \dots, z(k - k_s)$, we define the *extended performance*

$$Z(k) \triangleq \begin{bmatrix} z(k - k_1) \\ \vdots \\ z(k - k_s) \end{bmatrix} \in \mathbb{R}^{s l_z}. \quad (22)$$

Therefore,

$$Z(k) \triangleq \tilde{\mathcal{S}}(k) + \tilde{\mathcal{H}}\tilde{U}(k - 1), \quad (23)$$

where

$$\tilde{\mathcal{S}}(k) \triangleq \begin{bmatrix} \mathcal{S}_1(k - k_1) \\ \vdots \\ \mathcal{S}_s(k - k_s) \end{bmatrix} \in \mathbb{R}^{s l_z}, \quad (24)$$

$\tilde{U}(k - 1)$ has the form

$$\tilde{U}(k - 1) \triangleq \begin{bmatrix} u(k - q_1) \\ \vdots \\ u(k - q_{l_{\tilde{U}}}) \end{bmatrix} \in \mathbb{R}^{l_{\tilde{U}}}, \quad (25)$$

where, for $i = 1, \dots, l_{\tilde{U}}$, $k_1 \leq q_i \leq k_s + r$, and $\tilde{\mathcal{H}} \in \mathbb{R}^{s l_z \times l_{\tilde{U}}}$ is constructed according to the structure of $\tilde{U}(k - 1)$. The vector $\tilde{U}(k - 1)$ is formed by stacking $U_1(k - k_1 - 1), \dots, U_s(k - k_s - 1)$ and removing copies of repeated components.

For example, with $k_1 = 0$ and $k_2 = 1$, stacking $U_1(k - 1) = \begin{bmatrix} u(k - 1) \\ u(k - 2) \end{bmatrix}$ and $U_2(k - 2) = u(k - 2)$ results in $\tilde{U}(k - 1) = \begin{bmatrix} u(k - 1) \\ u(k - 2) \end{bmatrix}$. The coefficient matrix $\tilde{\mathcal{H}}$ consists of the entries of $\mathcal{H}_1, \dots, \mathcal{H}_s$ arranged according to the structure of $\tilde{U}(k - 1)$.

Next, we define the *surrogate performance*

$$\hat{z}(k - k_j) \triangleq \mathcal{S}_j(k - k_j) + \mathcal{H}_j \hat{U}_j(k - k_j - 1), \quad (26)$$

where the past controls $U_j(k - k_j - 1)$ in (19) are replaced by the surrogate controls $\hat{U}_j(k - k_j - 1)$. In analogy with (22), the *extended surrogate performance* for (26) is defined as

$$\hat{Z}(k) \triangleq \begin{bmatrix} \hat{z}(k - k_1) \\ \vdots \\ \hat{z}(k - k_s) \end{bmatrix} \in \mathbb{R}^{s l_z} \quad (27)$$

and thus is given by

$$\hat{Z}(k) = \tilde{S}(k) + \tilde{\mathcal{H}}\hat{U}(k-1), \quad (28)$$

where the components of $\hat{U}(k-1) \in \mathbb{R}^{l_{\hat{U}}}$ are the components of $\hat{U}_1(k-k_1-1), \dots, \hat{U}_s(k-k_s-1)$ ordered in the same way as the components of $\tilde{U}(k-1)$. Subtracting (23) from (28) yields

$$\hat{Z}(k) = Z(k) - \tilde{\mathcal{H}}\tilde{U}(k-1) + \tilde{\mathcal{H}}\hat{U}(k-1). \quad (29)$$

Finally, we define the *retrospective cost function*

$$J(\hat{U}(k-1), k) \triangleq \hat{Z}^T(k)R(k)\hat{Z}(k), \quad (30)$$

where $R(k) \in \mathbb{R}^{l_z s \times l_z s}$ is a positive-definite performance weighting. The goal is to determine refined controls $\hat{U}(k-1)$ that would have provided better performance than the controls $U(k)$ that were applied to the system. The refined control values $\hat{U}(k-1)$ are subsequently used to update the controller.

IV. Cost Function Optimization with Adaptive Regularization

To ensure that (30) has a global minimizer, we consider the regularized cost

$$\bar{J}(\hat{U}(k-1), k) \triangleq \hat{Z}^T(k)R(k)\hat{Z}(k) + \eta(k)\hat{U}^T(k-1)\hat{U}(k-1), \quad (31)$$

where

$$\eta(k) = \eta_0 Z_{p_c}^T(k)Z_{p_c}(k),$$

and $\eta_0 \geq 0$. Substituting (29) into (31) yields

$$\bar{J}(\hat{U}(k-1), k) = \hat{U}(k-1)^T \mathcal{A}(k) \hat{U}(k-1) + \mathcal{B}(k) \hat{U}(k-1) + \mathcal{C}(k), \quad (32)$$

where

$$\mathcal{A}(k) \triangleq \tilde{\mathcal{H}}^T R(k) \tilde{\mathcal{H}} + \eta(k) I_{l_{\hat{U}}}, \quad (33)$$

$$\mathcal{B}(k) \triangleq 2\tilde{\mathcal{H}}^T R(k) [Z(k) - \tilde{\mathcal{H}}\tilde{U}(k-1)], \quad (34)$$

$$\mathcal{C}(k) \triangleq Z^T(k)R(k)Z(k) - 2Z^T(k)R(k)\tilde{\mathcal{H}}\tilde{U}(k-1) + \tilde{U}^T(k-1)\tilde{\mathcal{H}}^T R(k)\tilde{\mathcal{H}}\tilde{U}(k-1). \quad (35)$$

If either $\tilde{\mathcal{H}}$ has full column rank or $\eta(k) > 0$, then $\mathcal{A}(k)$ is positive definite. In this case, $\bar{J}(\hat{U}(k-1), k)$ has the unique global minimizer

$$\hat{U}(k-1) = -\frac{1}{2}\mathcal{A}^{-1}(k)\mathcal{B}(k). \quad (36)$$

V. Controller Construction

The control $u(k)$ is given by the strictly proper time-series controller of order n_c given by

$$u(k) = \sum_{i=1}^{n_c} M_i(k)u(k-i) + \sum_{i=1}^{n_c} N_i(k)y(k-i), \quad (37)$$

where, for all $i = 1, \dots, n_c$, $M_i(k) \in \mathbb{R}^{l_u \times l_u}$ and $N_i(k) \in \mathbb{R}^{l_u \times l_y}$. The control (37) can be expressed as

$$u(k) = \theta(k)\phi(k-1), \quad (38)$$

where

$$\theta(k) \triangleq \begin{bmatrix} M_1(k) & \cdots & M_{n_c}(k) & N_1(k) & \cdots & N_{n_c}(k) \end{bmatrix} \in \mathbb{R}^{l_u \times n_c(l_u+l_y)} \quad (39)$$

and

$$\phi(k-1) \triangleq \begin{bmatrix} u(k-1) \\ \vdots \\ u(k-n_c) \\ y(k-1) \\ \vdots \\ y(k-n_c) \end{bmatrix} \in \mathbb{R}^{n_c(l_u+l_y)}. \quad (40)$$

Next, substituting (38) into (25) yields

$$\tilde{U}(k) = \Theta(k)\tilde{\phi}(k-1), \quad (41)$$

where

$$\Theta(k) \triangleq \begin{bmatrix} \theta(k-q_1+1) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \theta(k-q_{l_{\tilde{v}}}+1) \end{bmatrix} \in \mathbb{R}^{l_{\tilde{v}}l_u \times l_{\tilde{v}}n_c(l_u+l_y)} \quad (42)$$

and

$$\tilde{\phi}(k-1) \triangleq \begin{bmatrix} \phi(k-q_1) \\ \vdots \\ \phi(k-q_{l_{\tilde{v}}}) \end{bmatrix} \in \mathbb{R}^{l_{\tilde{v}}n_c(l_u+l_y)}. \quad (43)$$

Let d be a positive integer such that $\tilde{U}(k-1)$ contains $u(k-d)$, and let p be the data window size. Then $\hat{\tilde{U}}(k-1), \dots, \hat{\tilde{U}}(k-p+1)$ contain $\hat{u}(k-d), \dots, \hat{u}(k-d+1-p)$, respectively. We can thus construct

$$\Psi_p(k-d) \triangleq \begin{bmatrix} \hat{u}^T(k-d) \\ \vdots \\ \hat{u}^T(k-d+1-p) \end{bmatrix} \in \mathbb{R}^{p \times l_u}. \quad (44)$$

The matrix $\Psi_p(k-d)$ is used below for the controller update.

A. Batch Least Squares Update

Define

$$\Phi_p(k-d-1) \triangleq \begin{bmatrix} L_{y,p}(k-d-1) & L_{u,p}(k-d-1) \end{bmatrix} \in \mathbb{R}^{p \times [n_c(l_u+l_y)]}, \quad (45)$$

where

$$L_{y,p}(k-d-1) \triangleq \begin{bmatrix} y(k-d-1) & \cdots & y(k-n_c-d) \\ \vdots & \ddots & \vdots \\ y(k-d-p-1) & \cdots & y(k-n_c-d-p) \end{bmatrix} \quad (46)$$

and

$$L_{u,p}(k-d-1) \triangleq \begin{bmatrix} u(k-d-1) & \cdots & u(k-n_c-d) \\ \vdots & \ddots & \vdots \\ u(k-d-p-1) & \cdots & u(k-n_c-d-p) \end{bmatrix}. \quad (47)$$

Next, consider the quadratic cost

$$J_B(\theta(k)) \triangleq \|\Phi_p(k-d-1)\theta^T(k) - \Psi_p(k-d)\|^2 + \alpha(k)\text{tr}[\theta(k)\theta^T(k)], \quad (48)$$

where $\alpha(k) > 0$. Minimizing (48) yields the controller update

$$\theta^T(k) = [\Phi_p^T(k-d-1)\Phi_p(k-d-1) + \alpha(k)I]^{-1}\Phi_p^T(k-d-1)\Psi_p(k-d). \quad (49)$$

B. Recursive Least Squares Update

Next, noting that $\Psi_1(k-d) = \hat{u}^T(k-d)$, we define the cumulative cost function

$$J_R(\theta(k)) \triangleq \sum_{i=d+1}^k \lambda^{k-i} \|\phi^T(i-d-1)\theta^T(i-1) - \hat{u}^T(i-d)\|^2, \quad (50)$$

where $\|\cdot\|$ is the Euclidean norm, and $\lambda(k) \in (0, 1]$ is the forgetting factor. Minimizing (50) yields

$$\begin{aligned} \theta^T(k) = & \theta^T(k-1) + P(k-1)\phi(k-d-1)[\phi^T(k-d)P(k-1)\phi(k-d-1) \\ & + \lambda(k)]^{-1}[\phi^T(k-d-1)\theta^T(k-1) - \hat{u}^T(k-d)]. \end{aligned} \quad (51)$$

The error covariance is updated by

$$\begin{aligned} P(k) = & \lambda^{-1}(k)P(k-1) - \lambda^{-1}(k)P(k-1)\phi(k-d-1) \\ & \cdot [\phi^T(k-d-1)P(k-1)\phi(k-d) + \lambda(k)]^{-1}\phi^T(k-d-1)P(k-1). \end{aligned} \quad (52)$$

We initialize the error covariance matrix as $P(0) = \beta I$, where $\beta > 0$.

VI. SISO Examples

We now illustrate RCAC under nominal conditions, that is, LTI plants with no sensor noise, no unknown delays, and exact knowledge of the required Markov parameters. In this section, we consider stabilization, command following, and disturbance rejection problems for SISO plants. We consider a sequence of examples of increasing complexity, ranging from minimum-phase stable plants to nonminimum-phase or unstable plants. Each example is constructed such that, unless stated otherwise, the first nonzero Markov parameter $H_d = 1$, where d is the relative degree of G_{zu} , $z = y$, and $p_c = 1$. In all cases, the adaptive controller gain matrix is initialized to be zero, that is, $\theta(0) = 0$. Furthermore, in all examples, the state vector x is set randomly when the controller is turned on. Finally, for all examples, it is assumed that the exogenous input $w(k)$ and its characteristics are unknown. The simulation results corresponding to the examples in this section are given in Appendix A.

Example VI.1 (Minimum-phase, unstable plant, stabilization). Consider the plant G_{zu} with $d = 1$, poles 1.3, 1.1, 0.45, and minimum-phase zeros $0.6 \pm 0.4j$. For stabilization, we start with non-zero initial conditions, and let $w(k) \equiv 0$. We take $n_c = 3$, $P_0 = I_{2n_c}$, $\eta_0 = 0$, $p_c = 1$, and $\tilde{\mathcal{H}} = H_1$. The closed-loop response is shown in Figure 18.

Example VI.2 (Minimum-phase, stable plant, command following and disturbance rejection). Consider G_{zu} with $d = 1$, poles 0.8, 0.6, $0.5 \pm 0.5j$, and minimum-phase zeros $0.3 \pm 0.7j$, 0.5. We consider a combined step-command-following and sinusoidal-disturbance-rejection problem with command w_1 and disturbance w_2 given by

$$w(k) = \begin{bmatrix} w_1(k) \\ w_2(k) \end{bmatrix} = \begin{bmatrix} 2 \\ \sin \Omega_1 k \end{bmatrix}, \quad (53)$$

where $\Omega_1 = \pi/3$ rad/sample. With the plant realized in controllable canonical form (that is, $B = [1 \ 0 \ 0 \ 0]^T$), we take $D_1 = [0 \ 0 \ 0 \ 0]^T$ and $E_0 = [-1 \ 0]$, so that the disturbance is not matched with the input $u(k)$. Taking $n_c = 8$, $\alpha = 0.01$, $\eta_0 = 0.1$, and $\tilde{\mathcal{H}} = H_1$, the closed-loop response is shown in Figure 19. The controller converges to an internal-model controller with an integrator to follow the step command and poles at the disturbance frequency $\pi/3$ rad/sample to reject the disturbance.

Example VI.3 (Minimum-phase, unstable plant, disturbance rejection). Consider G_{zu} with $d = 1$, poles $1.1, 0.6 \pm 0.6j$, and minimum-phase zeros $0.3 \pm 0.7j$. We consider a disturbance rejection problem with the two-tone sinusoidal disturbance $w(k) = \begin{bmatrix} \sin \Omega_1 k & \sin \Omega_2 k \end{bmatrix}^T$, where $\Omega_1 = \pi/10$, $\Omega_2 = 2\pi/5$. With the plant realized in controllable canonical form, we take $D_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}^T$, so that none of the disturbances is matched with the input. Taking $n_c = 7$, $P_0 = 1000I_{2n_c}$, $\eta_0 = 0$, and $\tilde{\mathcal{H}} = H_1$, the closed-loop response is shown in Figure 20.

Example VI.4 (Minimum-phase, unstable plant, command following). Consider the double integrator plant $G_{zu} = \frac{1}{(z-1)^2}$. We take $D_1 = 0_{2 \times 1}$, $E_0 = -1$, and consider the sinusoidal command $w(k) = \sin \Omega_1 k$, where $\Omega_1 = 2\pi/15$. Taking $n_c = 5$, $P_0 = 100I_{2n_c}$, $\eta_0 = 0$, and $\tilde{\mathcal{H}} = H_2$, the closed-loop response is shown in Figure 21.

Example VI.5 (Nonminimum-phase, stable plant, disturbance rejection). Consider G_{zu} with $d = 1$, poles $0.5 \pm 0.5j, 0.95, 0.9, 0.2$, minimum-phase zero 0.4 , and nonminimum-phase zeros $1.3, 0.7 \pm 0.9j$. We consider the two-tone sinusoidal disturbance $w(k) = \begin{bmatrix} \sin \Omega_1 k & \sin \Omega_2 k \end{bmatrix}^T$, where $\Omega_1 = \pi/3$ and $\Omega_2 = \pi/3$. With the plant realized in controllable canonical form, we take $D_1 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}^T$. Taking $n_c = 9$, $P_0 = 0.01I_{2n_c}$, $\eta_0 = 0.02$, and $\tilde{\mathcal{H}} = H_1$, the closed-loop response is shown in Figure 22.

Example VI.6 (Nonminimum-phase, stable plant, command following). Consider G_{zu} with $d = 1$, poles $0.83, 0.5, 0.7, 0.42, 0.3 \pm 0.18j$, minimum-phase zero 0.69 , and nonminimum-phase zeros $3.7, 1.2, 1.5 \pm 1.1j$. For command following, we take $D_1 = 0_{6 \times 1}$, $E_0 = -1$, and consider the unknown sinusoidal command $w(k) = \sin \Omega_1 k$, where $\Omega_1 = 2\pi/3$. Taking $n_c = 8$, $\alpha \equiv 0.1$, $\eta_0 = 3$, and $\tilde{\mathcal{H}} = H_1$, the closed-loop response is shown in Figure 23.

Example VI.7 (Nonminimum-phase, unstable plant, disturbance rejection). Consider the plant G_{zu} with $d = 1$, lightly damped stable poles $0.75 \pm 0.65j$, unstable poles $1, 1$, minimum-phase zeros $0.9, 0.5$ and nonminimum phase zero 1.5 . We consider the unknown two-tone sinusoidal disturbance $w(k) = \sin \Omega_1 k + \sin \Omega_2 k$, where $\Omega_1 = \pi/4$, and $\Omega_2 = \pi/2$. With the plant realized in controllable canonical form, we take $D_1 = \begin{bmatrix} 0.9794 \\ -0.2656 \\ -0.5484 \\ -0.0963 \end{bmatrix}$. We take $n_c = 7$, $\alpha = 0.1$, $\eta = 0.25$, $p_c = 2$, and $\tilde{\mathcal{H}} = \begin{bmatrix} H_1 & H_2 \\ 0 & H_1 \end{bmatrix}$. The closed-loop response is shown in Figure 24.

VII. MIMO Examples

In this section we consider command following and disturbance rejection for MIMO plants. We consider square (equal number of inputs and outputs), tall (more outputs than inputs), and wide plants (more inputs than outputs). These plants can be further broken down into four subgroups: minimum-phase stable, minimum-phase unstable, nonminimum-phase stable, and nonminimum-phase unstable (see Table 1). We say that a square plant is minimum phase if all of its transmission zeros are inside the unit circle, and nonminimum phase if it has at least one transmission zero on or outside the unit-circle. In the non-square case, we say that a plant is minimum phase if all of its channel transfer functions are minimum phase, and nonminimum phase if at least one channel transfer function is nonminimum phase. Figure 2 shows transmission poles and zeros of the plants given by Table 1 in the same tabular order. For example, the pole-zero map in the first column of the first row is the pole-zero map of the square, minimum-phase, stable plant. This order will be used in all figures for the remainder of this section.

For all examples in this section, we take $n_c = 10$, $p_c = 2$, $\eta_0 = 1$, $P_0 = I_{(l_u+l_y)n_c}$, and $\tilde{\mathcal{H}} = H_2$.

A. Command Following

For command following the objective is to have the outputs follow the sinusoidal command $w(k) = \sin \Omega_1 k$, where $\Omega_1 = \pi/5$ rad/sample. With the plants described in controllable canonical form, we take $E_0 = -1_{l_z \times l_w}$ and $D_1 = 0_{n \times l_w}$. The closed-loop responses are shown in Figure 3. We see that the controller drives the performance to zero in the square and wide cases. Figure 4 shows that the control $u(k)$ is bounded in all cases. Figure 5 shows that the controller parameters $\theta(k)$ converge in less than 500 samples in all examples.

	MP Stable	MP Unstable	NMP Stable	NMP Unstable
Square	$G_1(z)$ $G_2(z)$ $G_2(z)$ $G_2(z)$	$G_3(z)$ $G_4(z)$ $G_4(z)$ $G_4(z)$	$G_5(z)$ $G_2(z)$ $G_2(z)$ $G_2(z)$	$G_6(z)$ $G_4(z)$ $G_4(z)$ $G_4(z)$
Tall	$G_1(z)$ $G_2(z)$	$G_3(z)$ $G_4(z)$	$G_5(z)$ $G_2(z)$	$G_6(z)$ $G_4(z)$
Wide	$G_1(z)$ $G_2(z)$	$G_3(z)$ $G_4(z)$	$G_5(z)$ $G_2(z)$	$G_6(z)$ $G_4(z)$

Table 1. Transfer matrices of the MIMO plants considered in this section. The tall (wide) transfer matrices are generated by taking the first column (row) of the square transfer matrix. The channel transfer functions are given in Table 2.

$G_1(z)$	$\frac{z+0.2}{(z-0.3)(z-0.4)}$
$G_2(z)$	$\frac{1}{(z-0.3)(z-0.4)}$
$G_3(z)$	$\frac{z+0.2}{(z-1.1)(z-0.4)}$
$G_4(z)$	$\frac{1}{(z-1.1)(z-0.4)}$
$G_5(z)$	$\frac{z-1.2}{(z-0.3)(z-0.4)}$
$G_6(z)$	$\frac{z-1.2}{(z-1.1)(z-0.4)}$

Table 2. Channel transfer functions of the MIMO plants considered in this section.

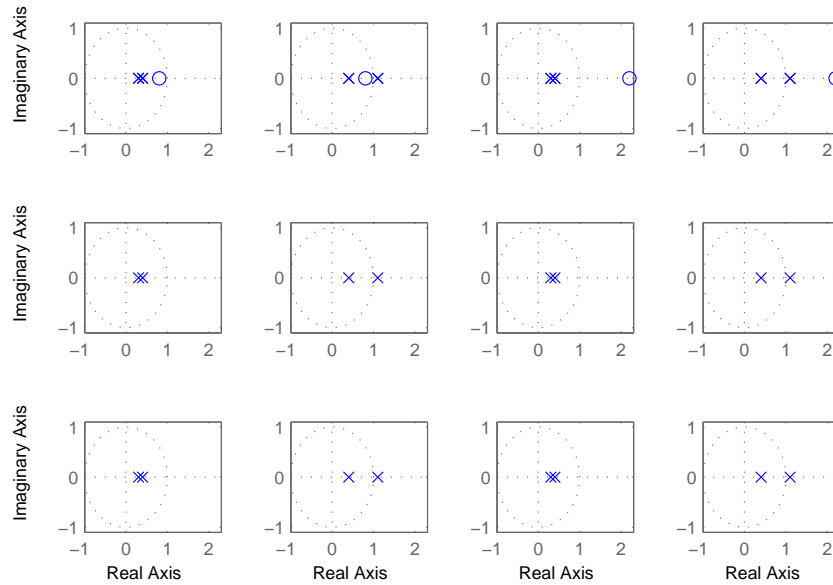


Figure 2. Pole-zero maps for the plants given in Table 1.

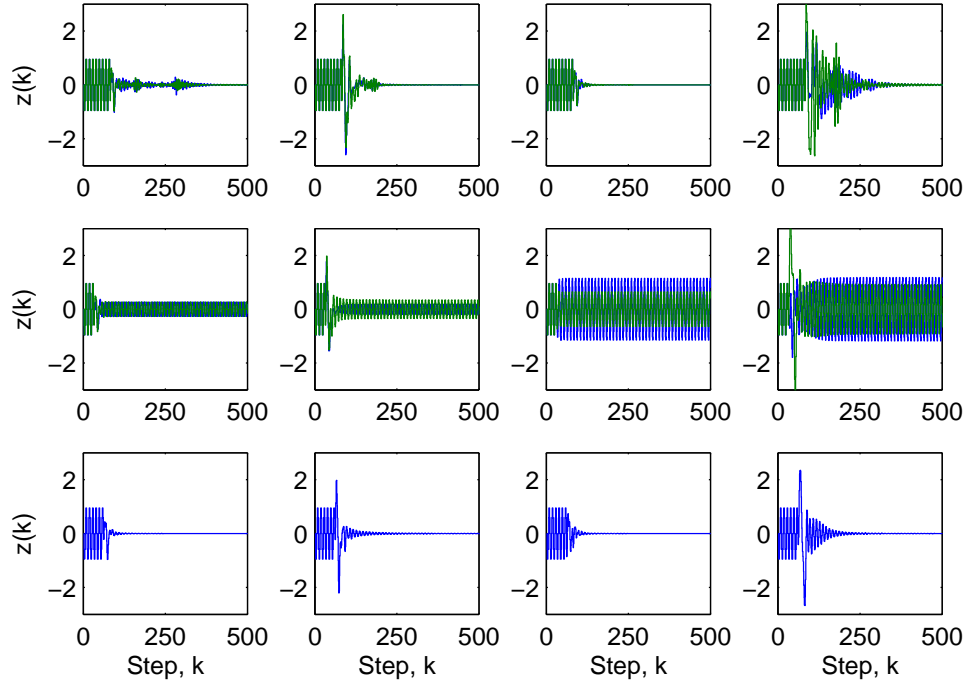


Figure 3. Closed-loop command following responses for the plants of Figure 2. The control is turned on at $k = l_u n_c (l_u + l_y)$. The controller order is $n_c = 10$ with parameters $p_c = 2$, $\eta_0 = 1$, $P_0 = I_{(l_u + l_y)n_c}$, and $\tilde{\mathcal{H}} = H_2$. The performance goes to zero in the square and wide cases.

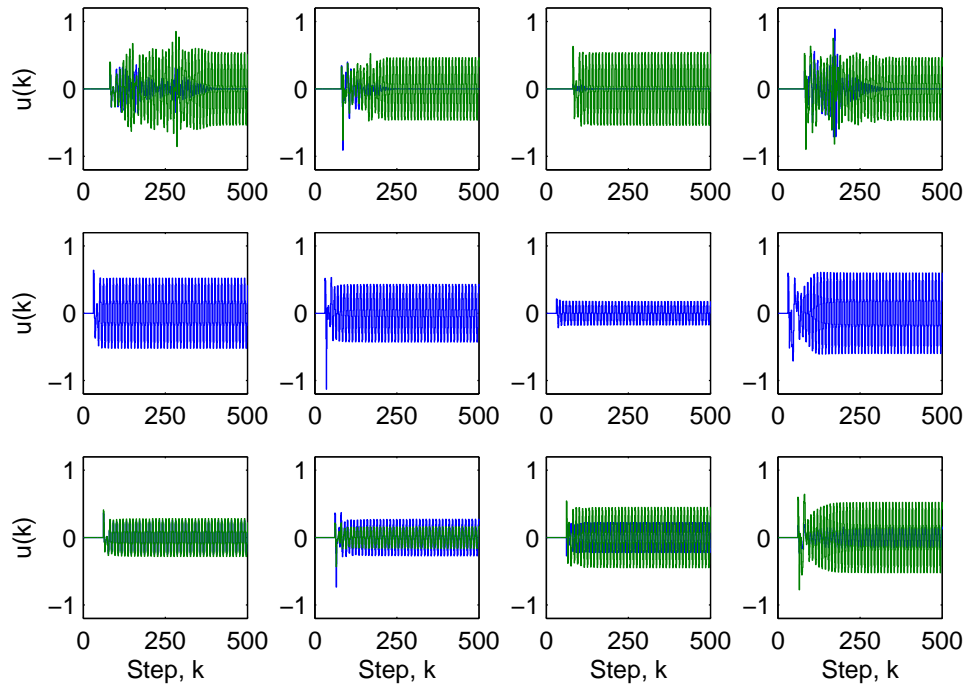


Figure 4. The control signal $u(k)$ exhibits an initial transient and remains bounded in all examples.

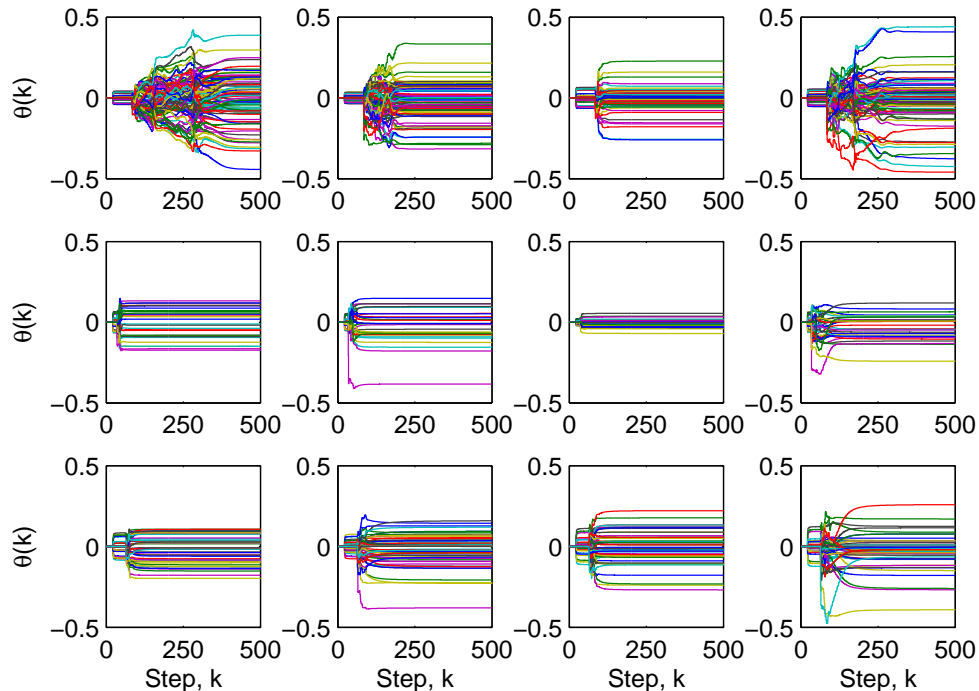


Figure 5. Time histories of the of the components of the adaptive controllers for the examples of Section A.

B. Disturbance Rejection

We now consider disturbance rejection problems, where the objective is to reject the sinusoidal disturbance $w(k) = \sin \Omega_1 k$, where $\Omega_1 = \pi/5$ rad/sample. With the plants described in controllable canonical form, we take $E_0 = 0_{l_z \times 1}$ and $D_1 = \begin{bmatrix} -0.7662 \\ -0.1612 \\ -0.8006 \\ 0.0772 \end{bmatrix}$ for 4th-order plants, and $D_1 = \begin{bmatrix} -0.7662 \\ -0.1612 \end{bmatrix}$ for 2nd-order plants.

Therefore, the disturbance is not matched with the input. In this section we consider stable minimum-phase and stable nonminimum-phase plants. Figure 6 shows that the performance is driven to zero in the square and wide cases, and the disturbance is attenuated in the tall cases by about 80 percent.

VIII. Extended Retrospective Cost Adaptive Control of the NASA GTM Using Minimal Modeling Information

In this section, we demonstrate control of the NASA GTM using extended RCAC. The linearized examples considered in this section are based on the following trim condition:

- (i) Flight path angle and angle of attack of 0 and 3 deg, respectively.
- (ii) Body x -axis, y -axis, and z -axis velocities of 161.66, 0, and 7.12 ft/sec, respectively.
- (iii) Angular velocities in roll, pitch, and yaw of 0, 0, and 0 deg/sec, respectively.
- (iv) Latitude, longitude, and altitude of 37.83 deg, -75.49 deg, and 800 ft, respectively.
- (v) Roll, pitch, and heading angles of 0.07, 3, and 90 deg, respectively.
- (vi) Elevator, aileron, and rudder angles of 2.7, 0, and 0 deg, respectively.

We consider adaptive control of GTM linearized about this trim condition, as well as control of the fully nonlinear system. In the linearized SISO examples, z consists of the altitude deviation from the commanded

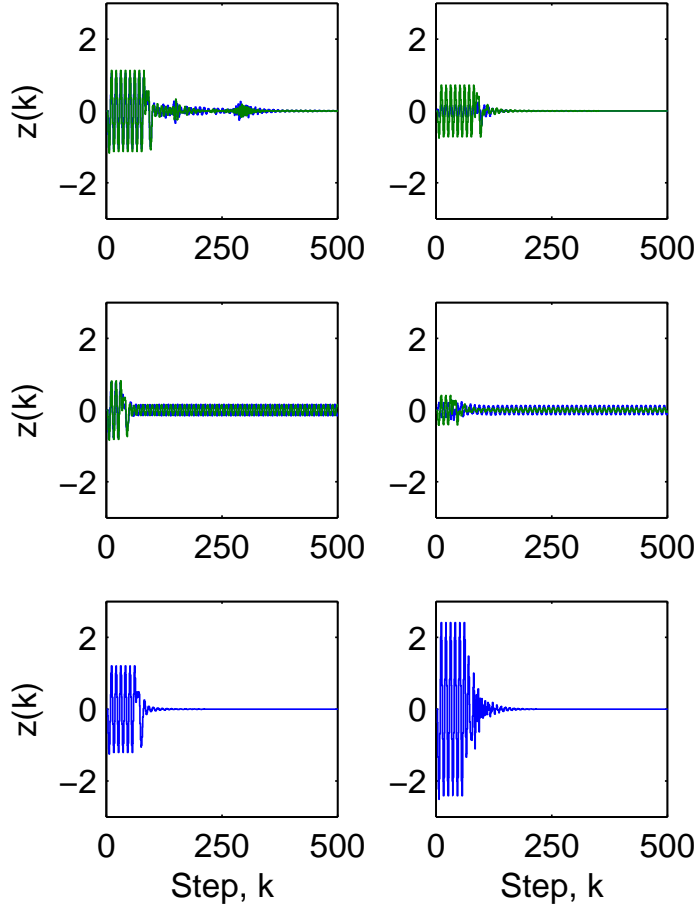


Figure 6. Closed-loop disturbance rejection responses for the plants given in Figure 2. The controller is turned on at $k = l_u n_c (l_u + l_y)$. The controller order is $n_c = 10$ with parameters $p_c = 2$, $\eta_0 = 1$, $P_0 = I_{(l_u + l_y)n_c}$, and $\tilde{\mathcal{H}} = H_2$. The performance converges to zero in the square and wide cases, and the disturbance is attenuated by 80 percent in the tall cases.

value, y consists of z , as well as the altitude command for feedforward control, while u consists of the commanded deviation in the elevator angle. In the MIMO examples, z consists of the deviations in the roll angle θ , yaw angle ψ , and altitude h from the commanded values, y consists of the deviations in θ , ψ , h , as well as the commands to altitude, roll, and heading for command feedforward, while u consists of the commanded deviations in elevator angle, differential aileron angle, and rudder angle from the trim flight condition.

For the linear and nonlinear GTM examples in this section, the adaptive controller is implemented with a command feedforward structure, a block diagram of which is shown in Figure 7. Furthermore, in all examples, GTM is sampled at 10 Hz, and we initialize the controller gains to be zero, that is, $\theta(0) = 0$, and the controller states are initialized to be zero. Finally, we stress that extended RCAC is the only controller implemented in the feedback loop, that is, no baseline or nominal controller is used, including no insertion of integrators.

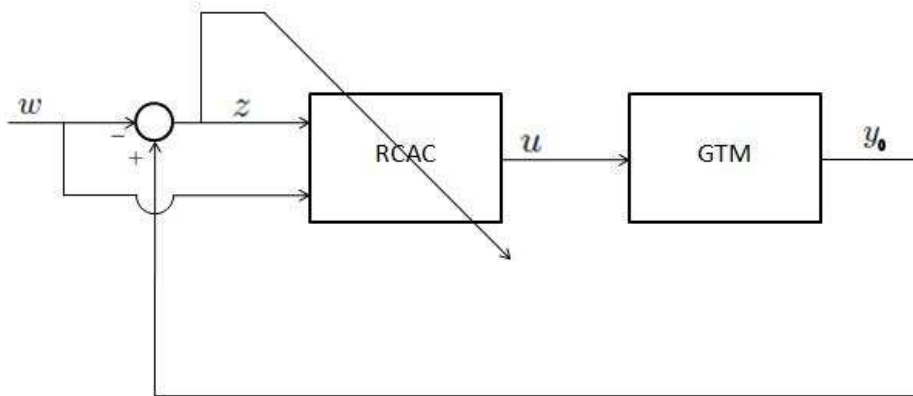


Figure 7. Command-following using a command-feedforward architecture. The performance variable is the error $z = y_0 - w$, and the controller input is $y = [z \quad w]^T$.

A. Linearized SISO GTM

Consider the SISO longitudinal dynamics of the NASA GTM model linearized about the trim condition given above and discretized with the sample time $T_s = 0.1$ second using zero-order hold. The input to the dynamical system is the commanded deviation in elevator angle, while the performance z is the altitude deviation from the commanded value. The resulting discrete-time plant G_{zu} is shown in Figure 8. Note that the linearized plant has a nonminimum-phase zero 25.1824, several weakly controllable/observable modes, and several poles near 1, making the plant dynamics extremely slow.

For command following, we take $D_1 = 0_{9 \times 1}$, $E_0 = -1$, and consider the sinusoidal altitude command $w(k) = \sin \Omega_1 k$, where $\Omega_1 = 2\pi/1000$ rad/sec/sample. We take $n_c = 7$, $\eta_0 = 0.001$, $P_0 = 0.1I_{3n_c}$, $p_c = 1$, and $\tilde{\mathcal{H}} = H_5$. The closed-loop response of the performance is shown in Figure 9. The altitude tracks the sinusoidal altitude command as shown in Figure 10.

Next, keeping D_1 and E_0 unchanged, we consider a command following problem whose objective is to have the altitude track a sequence of doublets with frequency $\Omega = 2\pi/1000$ rad/sec/sample. We take $n_c = 7$, $\eta_0 = 0.01$, $P_0 = 0.1I_{3n_c}$, $p_c = 1$, and $\tilde{\mathcal{H}} = H_5$. The closed-loop response of the performance is shown in Figure 11. The altitude tracks the altitude command as shown in Figure 12. Note that the transients improve at each doublet as the controller adapts.

B. Linearized MIMO GTM

We now consider the MIMO dynamics of the NASA GTM model linearized about the stated trim conditions and discretized using zero-order hold with the sampling period $T_s = 0.1$ second. The input u consists

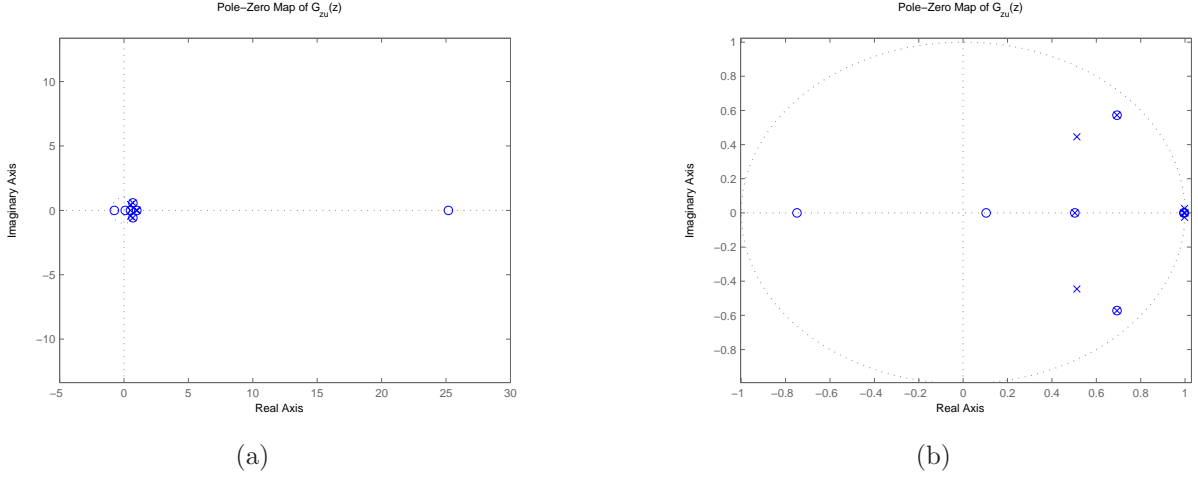


Figure 8. Pole-zero map of the linearized SISO GTM sampled at $\nu_s = 10$ Hz. (a) shows the NMP zero, while (b) shows the poles and zeros in the unit circle.

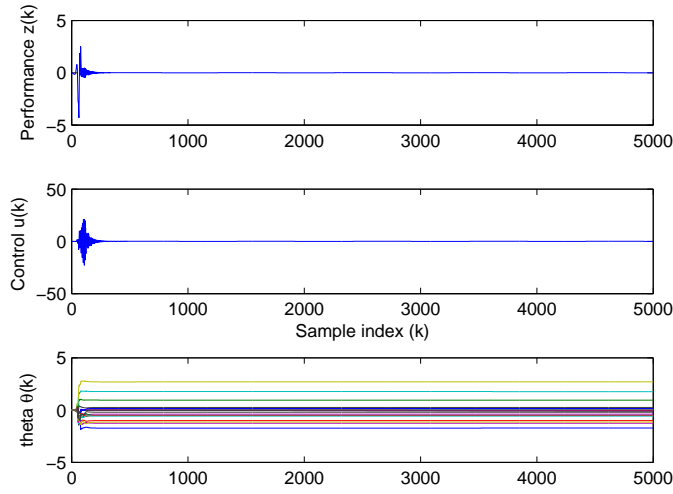


Figure 9. Closed-loop command-following response of the linearized SISO GTM with elevator input and altitude measurement. The controller order is $n_c = 7$ with parameters $P_0 = 0.1I_{3n_c}$ and $\eta = 0.001$. We take $\tilde{\mathcal{H}} = H_5$, so that only a single Markov parameter is used by the adaptive controller. Performance $z(k)$, which is the altitude deviation from the commanded value measured in feet, converges to zero in about 250 steps.

of deviations in the elevator angle, differential aileron angle, and rudder angle from the trim condition, while the performance z consists of the deviations in the roll angle, yaw angle, and altitude from the commanded values. We consider a MIMO command following problem whose objective is to have the output vector follow the command vector $w(k)$.

First, we consider the sinusoidal command $w(k) = \begin{bmatrix} A_1 \sin \Omega_1 k \\ A_2 \sin \Omega_2 k \\ A_3 \sin \Omega_3 k \end{bmatrix}$, where $\Omega_1 = 0.002$ rad/sec/sample, $\Omega_2 = 0.001$ rad/sec/sample, $\Omega_3 = 0.01$ rad/sec/sample, $A_1 = 0.08$ deg, $A_2 = 0.02$ deg, $A_3 = 1$ ft. Therefore, the goal is to have the roll angle, yaw angle, and altitude follow sinusoids with frequencies Ω_1 , Ω_2 , and Ω_3 , respectively. We take $n_c = 5$, $\eta_0 = 0.001$, $P_0 = 0.01I_{9n_c}$, $p_c = 1$, and $\tilde{\mathcal{H}} = H_4 \in \mathbb{R}^{3 \times 3}$. The closed-loop response of the performance $z(k)$, the control input $u(k)$, and adaptive controller gains are shown in Figure 13. The output vector tracks the command as shown in Figure 14.

We now consider a command signal consisting of a sequence of doublets having frequencies $\Omega_1 = 2\pi/500$ rad/sec/sample, $\Omega_2 = 2\pi/1000$ rad/sec/sample, $\Omega_3 = 2\pi/100$ rad/sec/sample, and $A_1 = A_2 = 0.08$ deg,

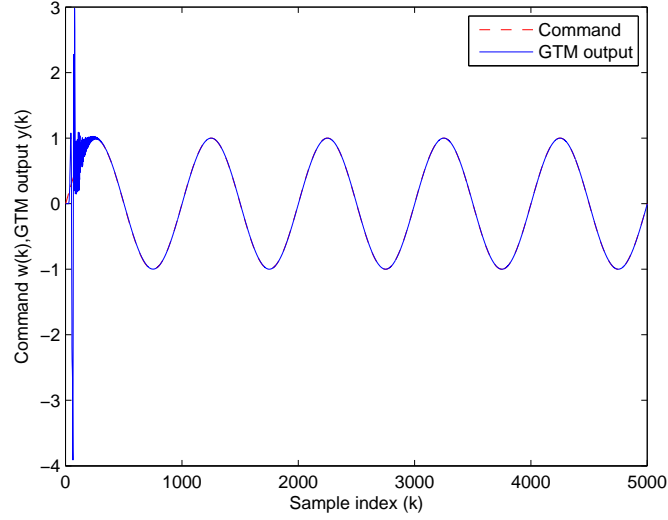


Figure 10. Closed-loop altitude response with sinusoidal altitude command, measured in feet. The controller order is $n_c = 7$ with parameters $P_0 = 0.1I_{3n_c}$ and $\eta_0 = 0.001$.

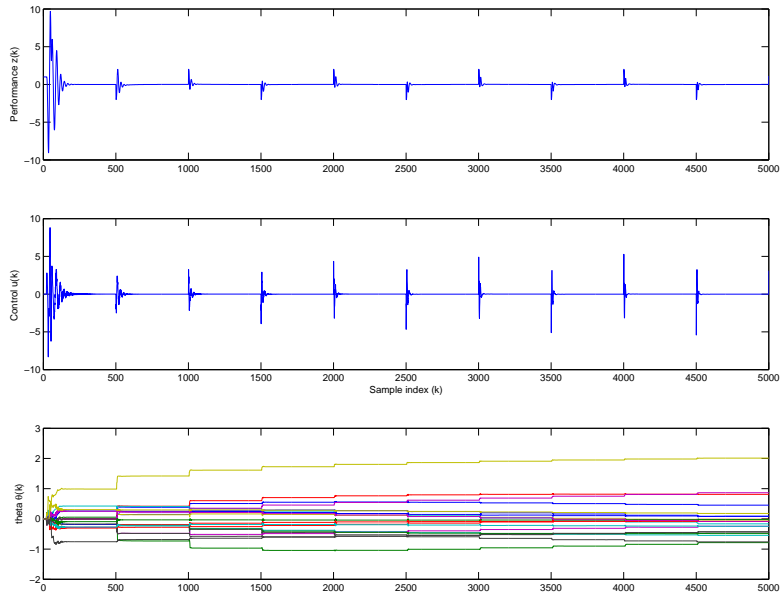


Figure 11. Closed-loop performance response of the linearized SISO GTM with elevator input and a sequence of doublet commands. The controller order is $n_c = 7$ with parameters $P_0 = 0.1I_{3n_c}$ and $\eta_0 = 0.01$. We take $\tilde{\mathcal{H}} = H_5$, so that only a single Markov parameter is used by the adaptive controller. Performance $z(k)$, which is the altitude deviation from the commanded value measured in feet, converges to zero in less than 100 steps at each doublet.

$A_3 = 1$ ft. We take $n_c = 5$, $\eta_0 = 0.001$, $P_0 = 0.01I_{9n_c}$, $p_c = 1$, and $\tilde{\mathcal{H}} = H_4 \in \mathbb{R}^{3 \times 3}$. The closed-loop response of the performance z , the control input u , and the adaptive controller gains are shown in Figure 15. The output vector tracks the command as seen in Figure 16.

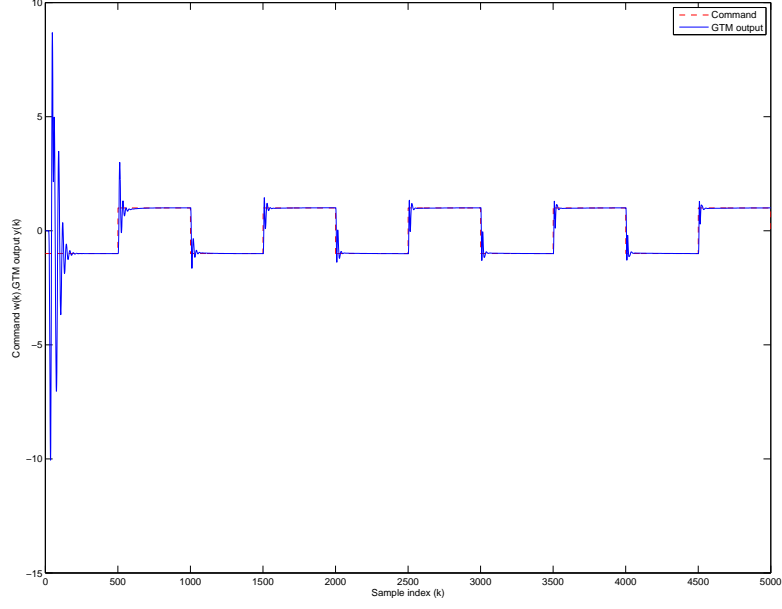


Figure 12. Closed-loop altitude reponse with doublet altitude command, measured in feet. The controller order is $n_c = 7$ with parameters $P_0 = 0.1I_{3n_c}$ and $\eta_0 = 0.01$.

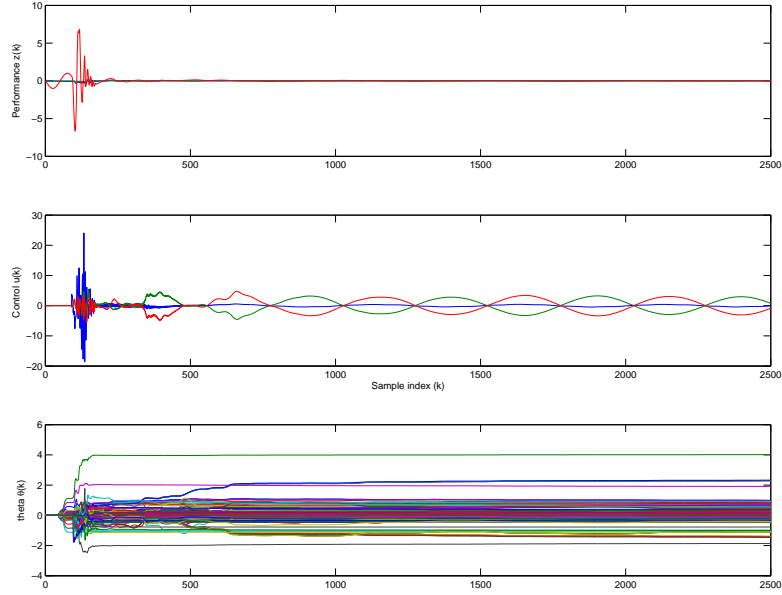


Figure 13. Closed-loop response of the linearized MIMO GTM with a sinusoidal command for each output. The closed-loop performance converges to zero, and the controller gains converge. The controller order is $n_c = 5$ with parameters $P_0 = 0.01I_{9n_c}$, $\eta_0 = 0.001$, and $\mathcal{H} = H_4 \in \mathbb{R}^{3 \times 3}$.

C. Nonlinear MIMO GTM

We now demonstrate adaptive control for the fully nonlinear MIMO GTM using the retrospective cost adaptive controller (38) based on the RLS update law (51)–(52) with the forgetting factor $\lambda = 1$. The

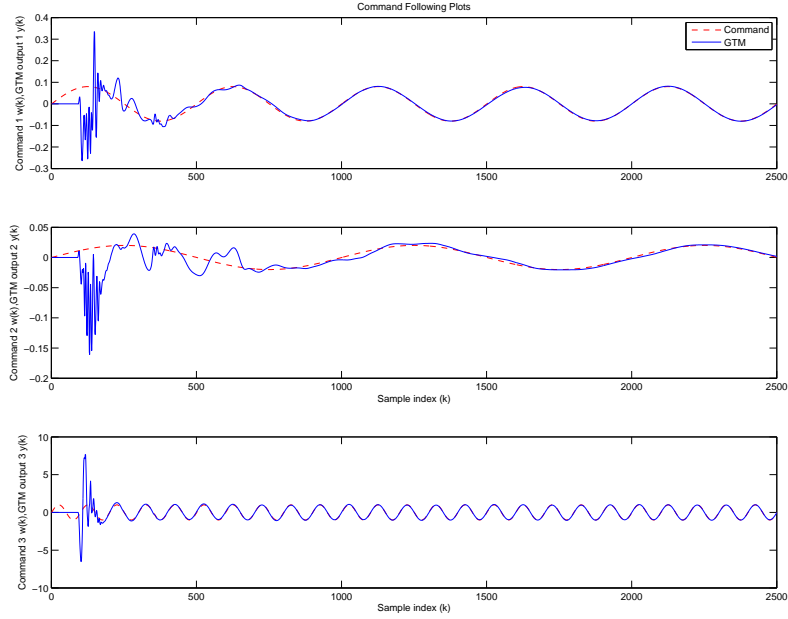


Figure 14. Closed-loop response of the linearized MIMO GTM with sinusoidal commands for each output. The controller order is $n_c = 5$ with parameters $P_0 = 0.01I_{9n_c}$, $\eta_0 = 0.001$, and $\hat{\mathcal{J}}\mathcal{C} = H_4 \in \mathbb{R}^{3 \times 3}$.

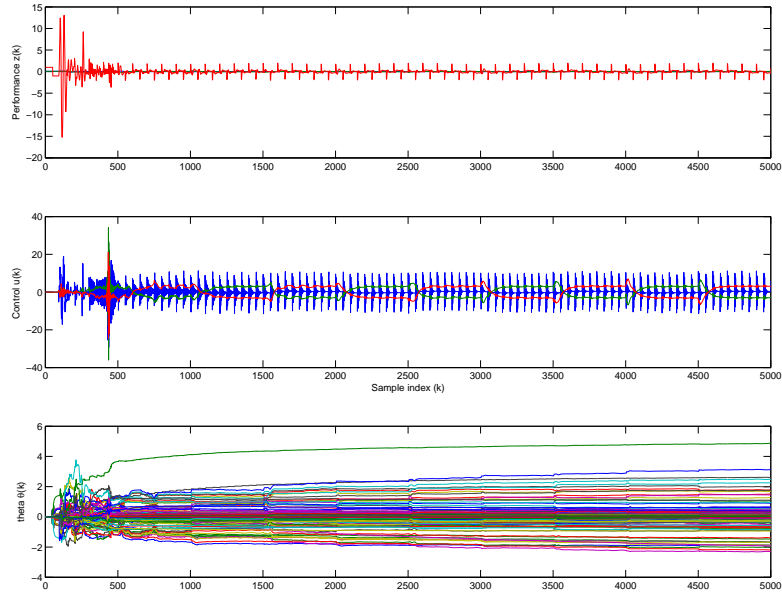


Figure 15. Closed-loop response of the linearized MIMO GTM with a sequence of doublet commands for each output. The closed-loop performances converge to zero after each doublet command, and the controller gains converge. The controller order is $n_c = 5$ with parameters $P_0 = 0.01I_{9n_c}$, $\eta_0 = 0.001$, and $\hat{\mathcal{J}}\mathcal{C} = H_4$.

simulation results corresponding to the examples considered in this section are given in Appendix B.

In each example, we control the nonlinear GTM using Markov parameters of GTM linearized about the trim condition given above. In all examples, we assume that the elevator, rudder, and differential ailerons

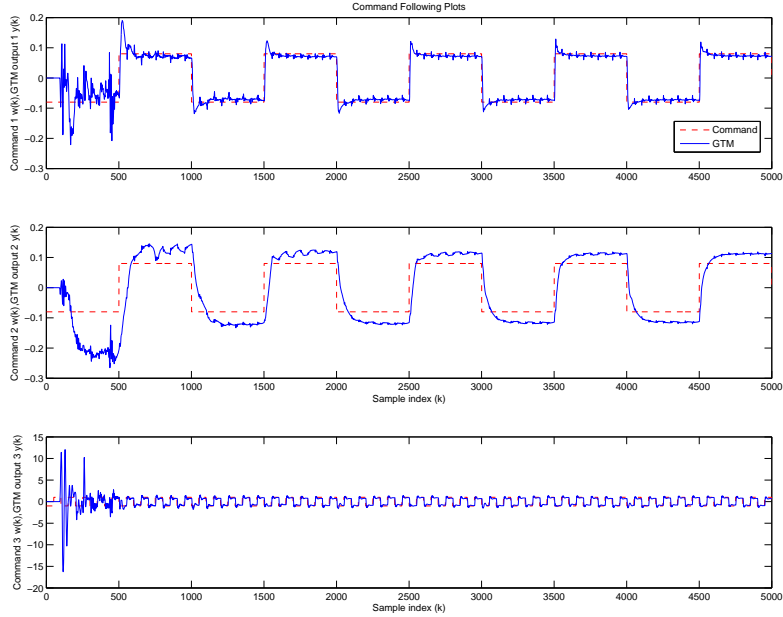


Figure 16. Closed-loop response of the linearized MIMO GTM with a sequence of doublet commands for each output. The controller order is $n_c = 5$ with parameters $P_0 = 0.01I_{9n_c}$ and $\eta_0 = 0.001$.

are available for control. We further assume that left and right throttles are available for control in Example VIII.10, where RCAC follows an airspeed command. GTM is sampled at the rate of 10 Hz, and the adaptive controller is implemented as shown in Figure 7. In Example VIII.10, two separate RCAC blocks are used in a decentralized architecture to achieve the control objective, a block diagram of which is shown in Figure 17. In that case, the first RCAC block is used to control differential ailerons and rudder to follow roll and heading commands, while the second RCAC block is used to control elevator, left throttle, and right throttle in order to follow airspeed, roll, and heading commands. In all examples, the controller gain matrix $\theta(k)$ and controller states are initialized to be zero.

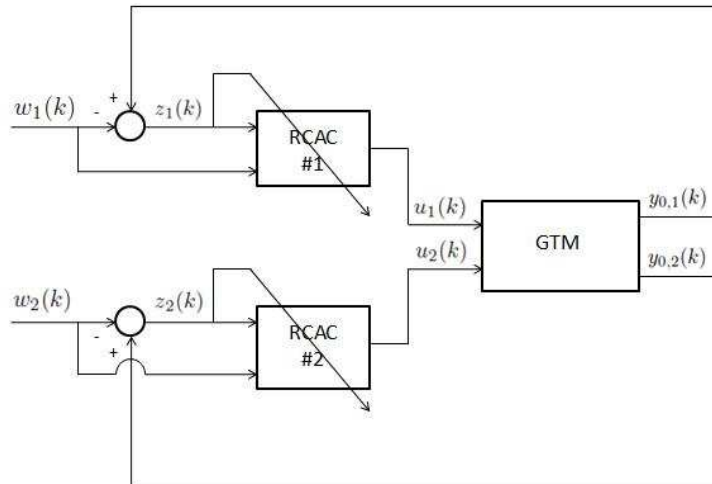


Figure 17. Decentralized command following architecture with 2 RCAC blocks.

The examples considered in this section include:

- Example VIII.1: Altitude command following.
- Example VIII.2: Altitude and heading command following.

- Example VIII.3: Altitude and heading command following.
- Example VIII.4: Altitude and roll command following.
- Example VIII.5: Altitude and roll command following.
- Example VIII.6: Altitude and heading command following with rudder failure.
- Example VIII.7: Altitude and roll command following with aileron failure.
- Example VIII.8: Altitude and roll command following with erroneous measurements.
- Example VIII.9: Roll and heading command following with erroneous measurements.
- Example VIII.10: Roll, heading, and airspeed command following with multiple RCAC blocks.

Example VIII.1 (Altitude command following). Consider the altitude command following problem, where the control objective is to have a 50-ft increase in altitude in 20 sec (200 time steps). To achieve the control objective, we let

$$y(k) = \begin{bmatrix} h(k) - w(k) & w(k) \end{bmatrix}^T,$$

and

$$z(k) = h(k) - w(k).$$

We let $n_c = 16$, $P_0 = 0.1I_{5n_c}$, $p_c = 10$, $\eta_0 = 0.1$, and $\tilde{\mathcal{H}} = \begin{bmatrix} H_4^T & H_3^T & H_2^T \end{bmatrix}^T \in \mathbb{R}^{3 \times 3}$. Figure 25 shows the time history of the closed-loop performance and control, whereas Figure 26 shows the time history of the trajectory of the aircraft.

Example VIII.2 (Altitude and heading command following). Consider the altitude and heading command following problem, where the control objective is to have the aircraft climb 100 ft while yawing 10 deg to change the heading course to 100 deg in 20 sec. To achieve the control objective, we let

$$y(k) = \begin{bmatrix} h(k) - w_1(k) & \psi(k) - w_2(k) & w_1(k) & w_2(k) \end{bmatrix}^T,$$

and

$$z(k) = \begin{bmatrix} h(k) - w_1(k) & \psi(k) - w_2(k) \end{bmatrix}^T,$$

where w_1 is the altitude command, and w_2 is the heading command. We let $n_c = 16$, $P_0 = 0.1I_{7n_c}$, $p_c = 10$, $\eta_0 = 0.1$, and $\tilde{\mathcal{H}} = \begin{bmatrix} H_4^T & H_3^T \end{bmatrix}^T \in \mathbb{R}^{4 \times 3}$. Figure 27 shows the time history of the closed-loop performance and control, and Figure 28 shows the time history of the trajectory of the aircraft.

Example VIII.3 (Altitude and heading command following). Consider the altitude and heading command following problem, where the control objective is to first have GTM climb to 900 ft while yawing to change the heading course from 90 deg to 100 deg. Next, after maintaining the altitude for 30 sec, the objective is to have the aircraft dive to 830 ft in 10 sec, and finally, after maintaining both altitude and heading in their commanded values for 10 sec, to have GTM yaw 5 deg to change the heading course from 100 deg to 95 deg in 10 sec. To achieve the control objective, we define $y(k)$ and $z(k)$ as in Example VIII.2. We let $n_c = 16$, $P_0 = 0.1I_{7n_c}$, $p_c = 10$, $\eta_0 = 0.1$, and $\tilde{\mathcal{H}} = \begin{bmatrix} H_4^T & H_3^T \end{bmatrix}^T \in \mathbb{R}^{4 \times 3}$. Figure 29 shows the time history of the closed-loop performance and control, and Figure 30 shows the time history of the trajectory of the aircraft.

Example VIII.4 (Altitude and roll command following). Consider the altitude and roll command following problem, where the control objective is to have the aircraft climb 100 ft while rolling 10 deg in 20 sec. To achieve the control objective, we let

$$y(k) = \begin{bmatrix} h(k) - w_1(k) & \theta(k) - w_2(k) & w_1(k) & w_2(k) \end{bmatrix}^T,$$

and

$$z(k) = \begin{bmatrix} h(k) - w_1(k) & \theta(k) - w_2(k) \end{bmatrix},$$

where w_1 is the altitude command, and w_2 is the heading command. We let $n_c = 16$, $P_0 = 0.1I_{7n_c}$, $p_c = 10$, $\eta_0 = 0.019$, and $\tilde{\mathcal{H}} = \begin{bmatrix} H_4^T & H_3^T \end{bmatrix}^T \in \mathbb{R}^{4 \times 3}$. Figure 31 shows the time history of the closed-loop performance and control, and Figure 32 shows the time history of the trajectory of the aircraft.

Example VIII.5 (Altitude and roll command following). Consider the altitude and roll command following problem, where the control objective is to first have the aircraft climb to 900 ft while rolling 10 deg. Next, after maintaining the altitude and roll trim for 30 sec, the objective is to have the aircraft dive to 830 ft in 10 sec, and finally, after maintaining the altitude and roll trim for 10 sec, to have the aircraft change the roll trim from 10 deg to 5 deg in 10 sec. To achieve the control objective, we define $y(k)$ and $z(k)$ as in Example VIII.4. We let $n_c = 16$, $P_0 = 0.1I_{7n_c}$, $p_c = 10$, $\eta_0 = 0.019$, and $\tilde{\mathcal{H}} = \begin{bmatrix} H_4^T & H_3^T \end{bmatrix}^T \in \mathbb{R}^{4 \times 3}$. Figure 33 shows the time history of the closed-loop performance and control, and Figure 34 shows the time history of the trajectory of the aircraft.

Example VIII.6 (Altitude and heading command following with rudder failure). Consider the altitude and heading command following problem, where the control objective is to have the aircraft climb to 900 ft while yawing to change the heading course from 90 deg to 100 deg in 20 sec. Furthermore, assume that, after 20 sec, the rudder experiences damage and is locked at the off-nominal angle -0.08 deg. To achieve the control objective, we define $y(k)$ and $z(k)$ as in Example VIII.2. We let $n_c = 16$, $P_0 = 0.1I_{7n_c}$, $p_c = 10$, $\eta_0 = 0.05$, and $\tilde{\mathcal{H}} = \begin{bmatrix} H_4^T & H_3^T \end{bmatrix}^T \in \mathbb{R}^{4 \times 3}$. Figure 35 shows the time history of the closed-loop performance and control, and Figure 36 shows the time history of the trajectory of the aircraft.

Example VIII.7 (Altitude and roll command following with aileron failure). Consider the altitude and roll command following problem, where the control objective is to have the aircraft climb to 900 ft while rolling 10 deg in 20 sec. Furthermore, assume that, after 40 sec, the ailerons experience damage, and are locked at the off-nominal differential angle -0.2 deg. To achieve the control objective, we define $y(k)$ and $z(k)$ as in Example VIII.4. We let $n_c = 16$, $P_0 = 0.01I_{7n_c}$, $p_c = 10$, $\eta_0 = 0.019$, and $\tilde{\mathcal{H}} = \begin{bmatrix} H_4^T & H_3^T \end{bmatrix}^T \in \mathbb{R}^{4 \times 3}$. Figure 37 shows the time history of the closed-loop performance and control, and Figure 38 shows the time history of the trajectory of the aircraft.

Example VIII.8 (Altitude and roll command following with sensor bias). Consider the altitude and roll command following problem, where the control objective is to have the aircraft climb from 20000 ft to 20050 ft in 20 sec while maintaining the roll angle constant at 0 deg. Furthermore, assume that the sensors are imperfect, and the roll measurement $\tilde{\theta}(k)$ has a 10 deg bias from the true roll angle $\theta(k)$. To achieve the control objective, we define $y(k)$ and $z(k)$ as in Example VIII.4. We let $n_c = 16$, $P_0 = 0.01I_{7n_c}$, $p_c = 10$, $\eta_0 = 0.019$, and $\tilde{\mathcal{H}} = \begin{bmatrix} H_4^T & H_3^T \end{bmatrix}^T \in \mathbb{R}^{4 \times 3}$. Figure 39 shows the time history of the closed-loop performance and control, and Figure 40 shows the time history of the trajectory of the aircraft.

Example VIII.9 (Roll and heading command following with sensor bias). Consider the roll and heading command following problem, where the control objective is to have the aircraft yaw to change the heading course from 90 deg to 100 deg in 20 sec, while keeping the roll angle constant at 0 deg. Furthermore, assume that the sensors are imperfect, and the roll measurement $\tilde{\theta}(k)$ has a -5 deg bias from the true roll angle $\theta(k)$. To achieve the control objective, we let

$$y(k) = \begin{bmatrix} \tilde{\theta}(k) - w_1(k) & \psi(k) - w_2(k) & w_1(k) & w_2(k) \end{bmatrix}^T,$$

and

$$z(k) = \begin{bmatrix} \tilde{\theta}(k) - w_1(k) & \psi(k) - w_2(k) \end{bmatrix},$$

where w_1 is the roll command, and w_2 is the heading command. We let $n_c = 16$, $P_0 = 0.01I_{7n_c}$, $p_c = 10$, $\eta_0 = 0.01$, and $\tilde{\mathcal{H}} = \begin{bmatrix} H_4^T & H_3^T \end{bmatrix}^T \in \mathbb{R}^{4 \times 3}$. Figure 41 shows the time history of the closed-loop performance and control, and Figure 42 shows the time history of the trajectory of the aircraft.

Example VIII.10 (Roll, heading, and airspeed command following with multiple RCAC blocks). Consider the roll, heading, and airspeed command following problem, where the control objective is to have the aircraft roll 10 deg, yaw to attain a heading of 100 deg, and increase the airspeed V to 150 knot in 20 seconds. To achieve the control objective, we use decentralized RCAC architecture (see Figure 17) with

$$\begin{aligned} y_1(k) &= \begin{bmatrix} \theta(k) - w_{1,1}(k) & \psi(k) - w_{1,2}(k) & w_{1,1}(k) & w_{1,2}(k) \end{bmatrix}^T, \\ z_1(k) &= \begin{bmatrix} \theta(k) - w_{1,1}(k) & \psi(k) - w_{1,2}(k) \end{bmatrix}^T, \\ y_2(k) &= \begin{bmatrix} \theta(k) - w_{2,1}(k) & \psi(k) - w_{2,2}(k) & V(k) - w_{2,3}(k) & w_{2,1}(k) & w_{2,2}(k) & w_{2,3}(k) \end{bmatrix}^T, \\ z_2(k) &= \begin{bmatrix} \theta(k) - w_{2,1}(k) & \psi(k) - w_{2,2}(k) & V(k) - w_{2,3}(k) \end{bmatrix}^T, \\ w_1(k) &= \begin{bmatrix} w_{1,1}(k) & w_{1,2}(k) \end{bmatrix}^T, \end{aligned}$$

and

$$w_2(k) = \begin{bmatrix} w_{2,1}(k) & w_{2,2}(k) & w_{2,3}(k) \end{bmatrix}^T,$$

where $w_{1,1} = w_{2,1}$ is the roll command, $w_{1,2} = w_{2,2}$ is the heading commands, $w_{2,3}$ is the airspeed command, $u_1(k)$ consists of aileron and rudder, and $u_2(k)$ consists of elevator, left throttle, and right throttle. For the first RCAC block, we let $n_{c,1} = 16$, $P_{0,1} = 0.01I_{6n_c}$, $p_{c,1} = 10$, $\eta_{0,1} = 0.01$, and $\tilde{\mathcal{H}} = H_{4,1}$, where $H_{4,1} \in \mathbb{R}^{2 \times 2}$ is the 4th Markov parameter of $G_{z_1 u_1}$; for the second RCAC block, we let $n_{c,2} = 16$, $P_{0,2} = 0.01I_{9n_c}$, $p_{c,2} = 10$, $\eta_{0,2} = 0.01$, and $\tilde{\mathcal{H}} = H_{4,2}$, where $H_{4,2} \in \mathbb{R}^{2 \times 2}$ is the 4th Markov parameter of $G_{z_2 u_2}$. Figure 43 shows the time history of the closed-loop performance and control, and Figure 44 shows the time history of the trajectory of the aircraft.

IX. Conclusions

In this paper we demonstrated an extended version of retrospective cost adaptive control, in which less modeling information is required than in prior versions of the technique. Specifically, this paper provides a method for adaptive control of MIMO possibly nonminimum-phase (NMP) plants without the need to know the locations of the NMP zeros. The only required modeling information is a limited number of Markov parameters. For SISO plants, one Markov parameter is needed, while, for MIMO plants, a sufficient number of Markov parameters is needed to provide left invertibility of the coefficient matrix in the retrospective cost function. With this limited modeling information, the NMP zeros, if any, of the plant are not captured. Consequently, the method that we have presented is robust to the presence and location of NMP zeros, assuming that the identified values of the chosen Markov parameters are correct. Ongoing research is focusing on the stability and performance robustness of the adaptive closed-loop system to uncertainty in the chosen Markov parameters.

A. Plots for Section VI

In this appendix, we present simulation results corresponding to the numerical SISO examples considered in Section VI. Each figure contains 8 subplots arranged in four rows and two columns. In each figure, first row shows the time history of the performance variable $z(k)$ and the control signal $u(k)$; second row shows the time traces of the controller gain vector $\theta(k)$ and the spectral radius $\text{spr}(\tilde{A})$ of the closed-loop state matrix; third row shows the pole-zero maps of the open-loop plant $G_{zu}(z)$ and the output feedback controller $G_c(z)$ after convergence; fourth row shows Bode magnitude plots of the relevant transfer functions after convergence, plotted versus $\theta \in [0, \pi]$, where 0 corresponds to the DC-frequency and π corresponds to the Nyquist frequency.

In the controller pole-zero maps, the controller poles that are located close to disturbance frequencies are marked in red, while the poles that are located close to command frequencies are marked in green. Likewise, the closed-loop bode magnitude plots from a disturbance signal to performance are drawn in red, while the closed-loop bode magnitude plots from a command signal to performance are drawn in green.

B. Plots for Section VIII.C

In this appendix, we present simulation results corresponding to the numerical examples concerning control of the fully nonlinear GTM. For each example, we present a figure to illustrate the time traces of important closed-loop states and control signals, as well as a figure to illustrate the trajectory of the GTM in 3D coordinate frame. Since the nonlinear GTM model includes actuator dynamics and nonlinearities, the requested control signals are in general not equal to the actual control surface responses. Therefore, for each control surface, the requested control signal from RCAC is drawn in dashed red, whereas the actual control surface response is drawn in solid blue.

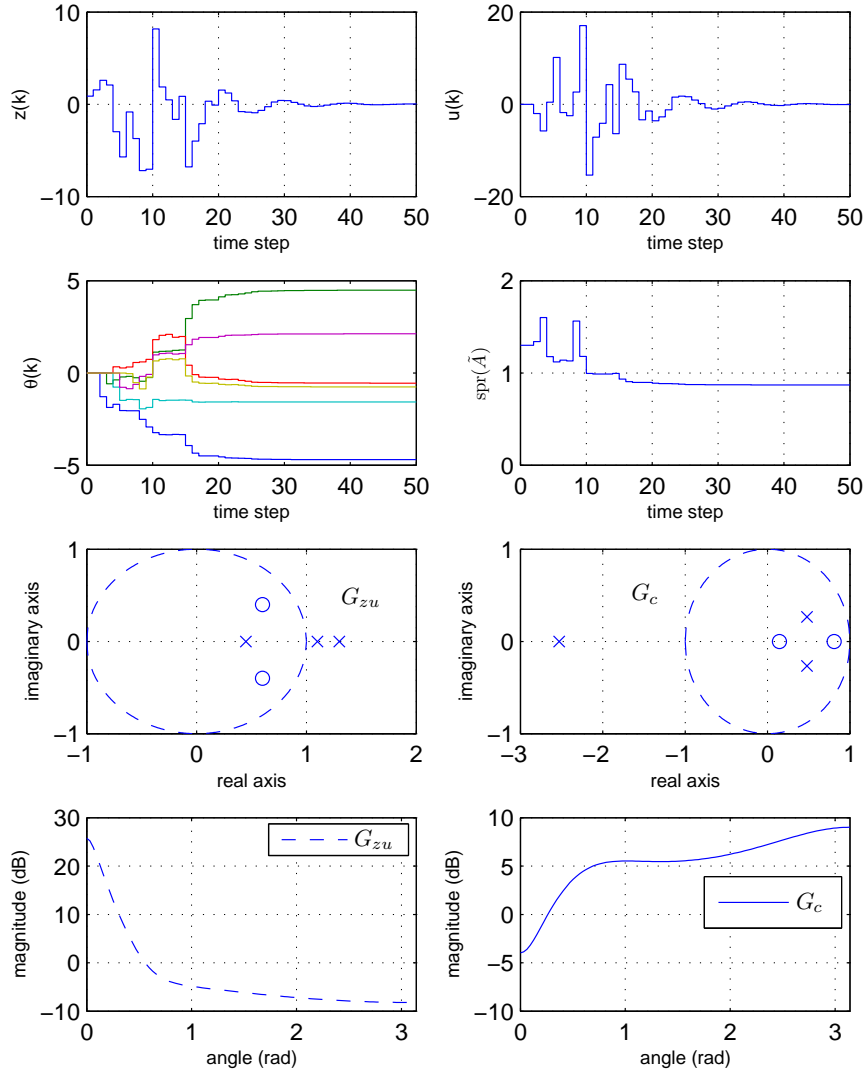


Figure 18. Example VI.1: Minimum-phase, unstable plant, $n = 3$, $d = 1$, stabilization. RCAC uses $\tilde{\mathcal{H}} = H_1$, so that only one Markov parameter is used by the adaptive controller. The controller is turned on at $k = 1$. The controller order is $n_c = 3$ with parameters $P_0 = I_{2n_c}$, $p_c = 1$, and $\eta_0 = 0$. The performance $z(k)$ converges to zero in about 40 steps. After convergence, the closed-loop system is asymptotically stable with spectral radius $\text{spr}(\hat{A}) = 0.87$.

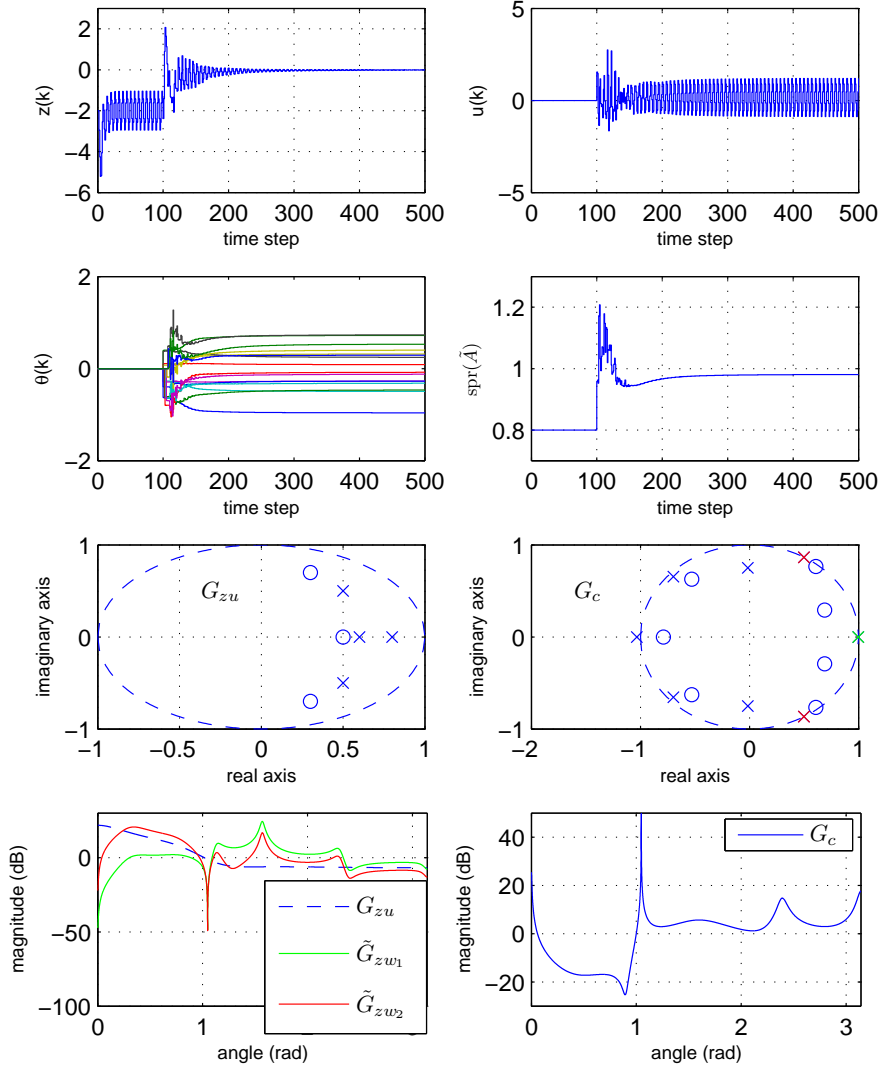


Figure 19. Example VI.2: Minimum-phase, stable plant, $n = 4$, $d = 1$, command following and disturbance rejection. RCAC uses $\mathcal{H} = H_1$, so that only one Markov parameter is used by the adaptive controller. The controller is turned on at $k = 100$. The controller order is $n_c = 8$ with parameters $\alpha \equiv 0.01$ and $\eta_0 = 0.1$. The performance $z(k)$ converges to zero in about 160 steps. RCAC converges to an internal model controller with an integrator and poles at disturbance frequency $\pi/3$ rad/sample. After convergence, the closed-loop system is asymptotically stable with spectral radius $\text{spr}(\tilde{A}) = 0.98$.

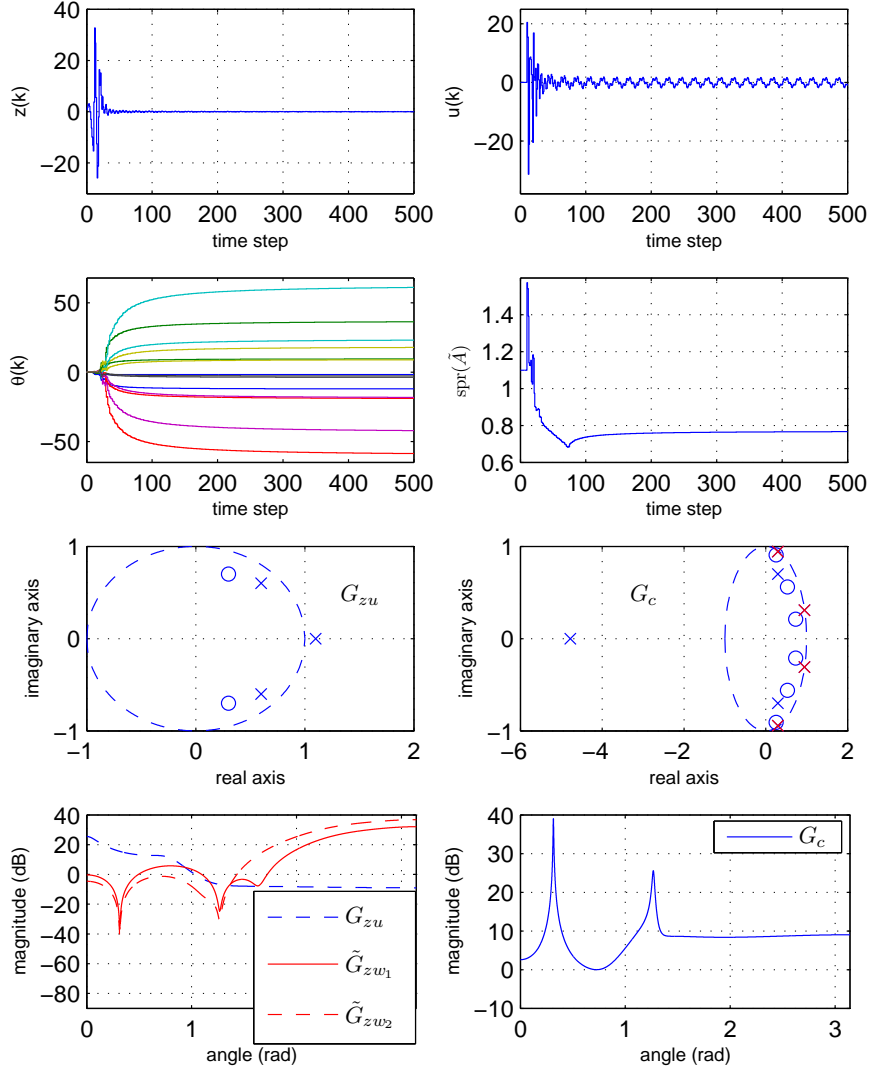


Figure 20. Example VI.3: Minimum-phase, unstable plant, $n = 3$, $d = 1$, disturbance rejection. RCAC uses $\mathcal{H} = H_1$, so that only one Markov parameter is used by the adaptive controller. The controller is turned on at $k = 10$. The controller order is $n_c = 7$ with parameters $P_0 = 1000I_{2n_c}$, and $\eta_0 = 0$. The performance $z(k)$ converges to zero in about 110 steps. RCAC converges to an internal model controller with poles at the disturbance frequencies $\pi/10$ rad/sec and $2\pi/5$ rad/sec. After convergence, the closed-loop system is asymptotically stable with spectral radius $\text{spr}(\hat{A}) = 0.77$.

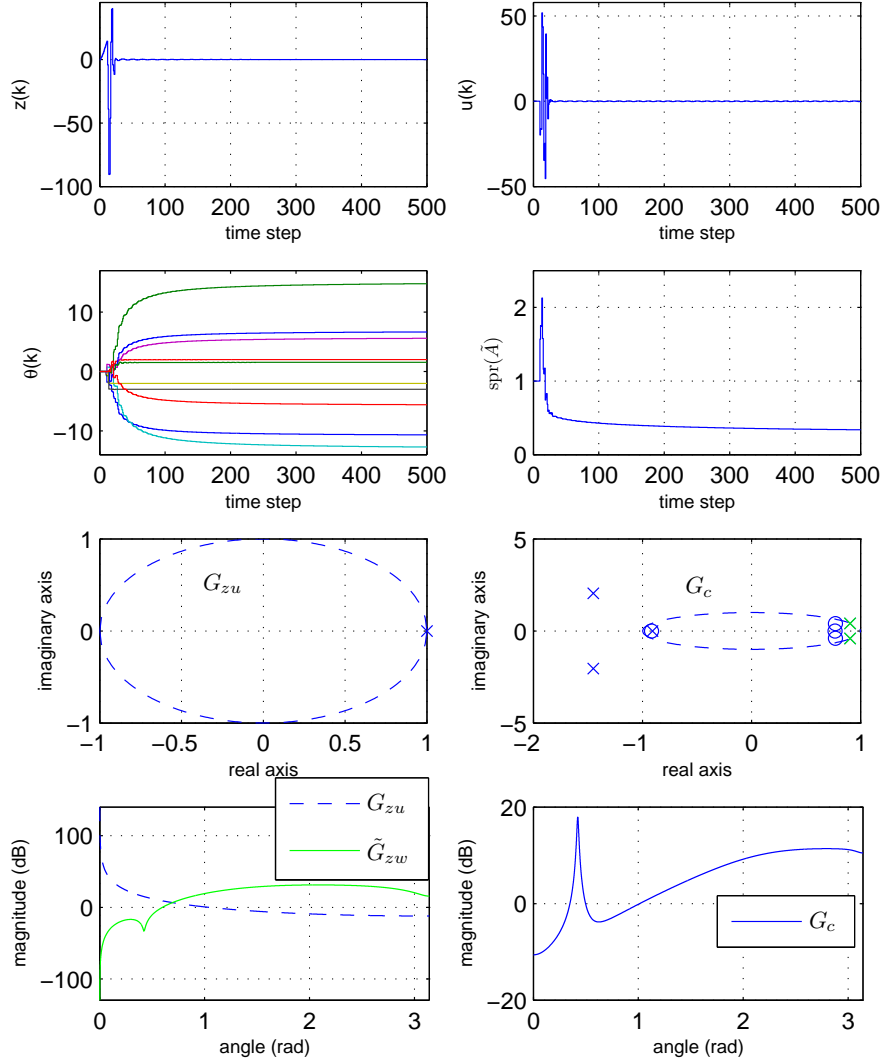


Figure 21. Example VI.4: Minimum-phase, unstable plant, $n = 2$, $d = 2$, command following. RCAC uses $\mathcal{H} = H_2$, so that only one Markov parameter is used by the adaptive controller. The controller is turned on at $k = 10$. The controller order is $n_c = 5$ with parameters $P_0 = 100I_{2n_c}$, and $\eta_0 = 0$. The performance $z(k)$ converges to zero in about 60 steps. RCAC converges to an internal model controller with poles at the command frequency $2\pi/15$. After convergence, the closed-loop system is asymptotically stable with spectral radius $\text{spr}(\hat{A}) = 0.34$.

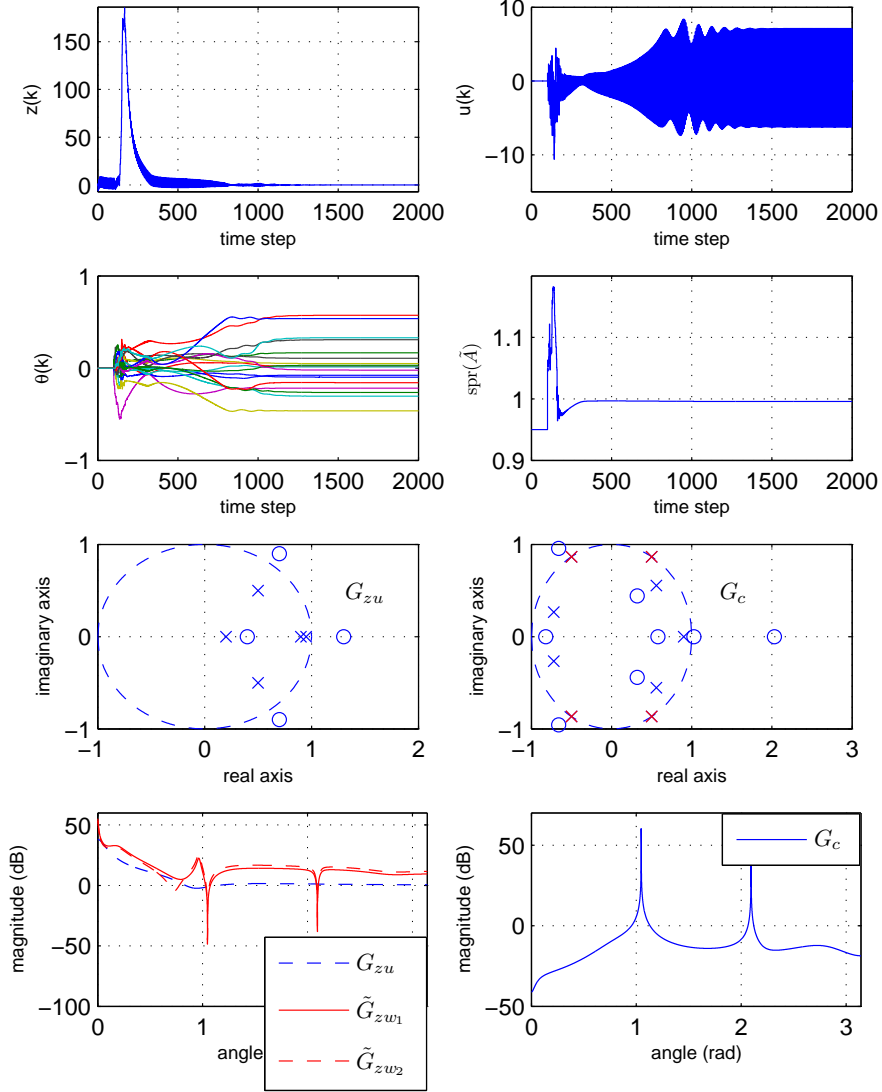


Figure 22. Example VI.5: Nonminimum-phase, stable plant, $n = 5$, $d = 1$, disturbance rejection. RCAC uses $\mathcal{H} = H_1$, so that only one Markov parameter is used by the adaptive controller. The controller is turned on at $k = 100$. The controller order is $n_c = 9$ with parameters $P_0 = 0.01I_{2n_c}$, and $\eta_0 = 0.02$. The performance $z(k)$ converges to zero in about 1200 steps. RCAC converges to an internal model controller with poles at the disturbance frequencies $\pi/3$ rad/sample and $2\pi/3$ rad/sample. After convergence, the closed-loop system is asymptotically stable with spectral radius $\text{spr}(\tilde{A}) = 0.996$.

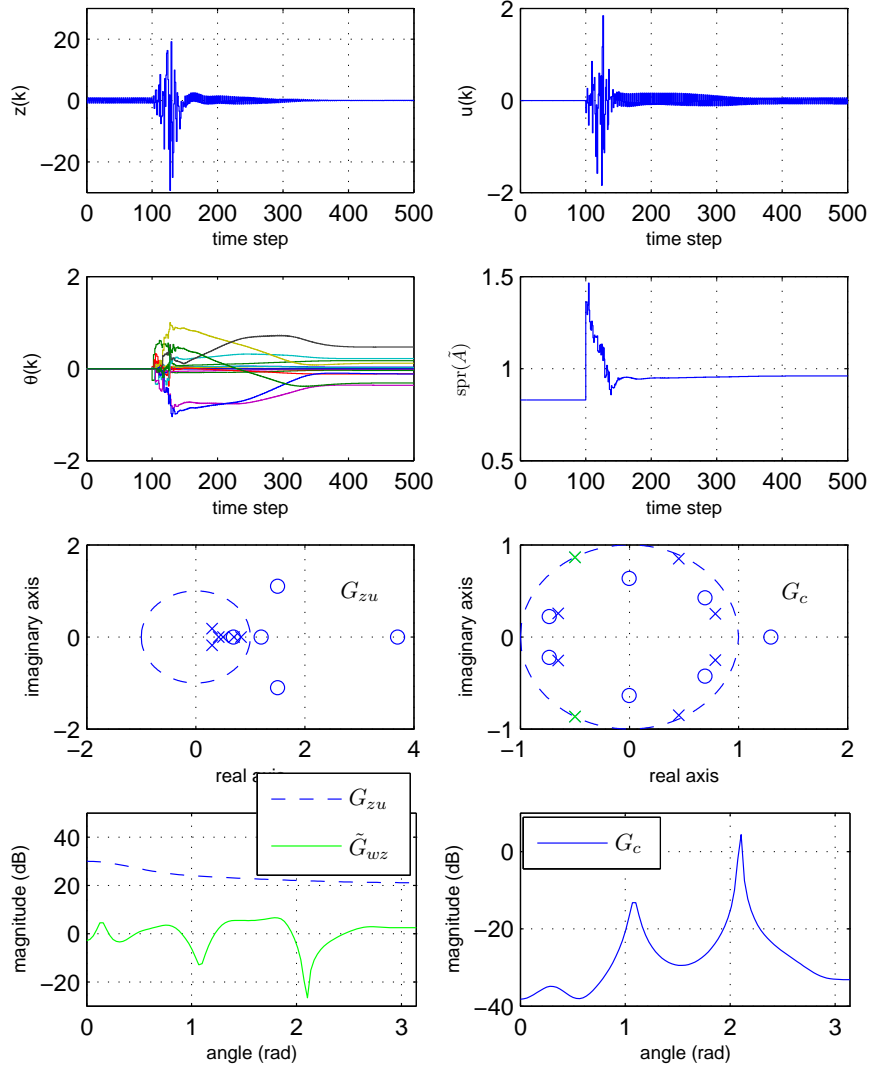


Figure 23. Example VI.6: Nonminimum-phase, stable plant, $n = 6$, $d = 1$, command following. RCAC uses $\mathcal{H} = H_1$, so that only one Markov parameter is used by the adaptive controller. The controller is turned on at $k = 100$. The controller order is $n_c = 8$ with parameters $\alpha \equiv 0.1$, and $\eta_0 = 3$. The performance $z(k)$ converges to zero in about 250 steps. RCAC converges to an internal model controller with poles at the command frequency $2\pi/3$ rad/sample. After convergence, the closed-loop system is asymptotically stable with spectral radius $\text{spr}(\tilde{A}) = 0.96$.

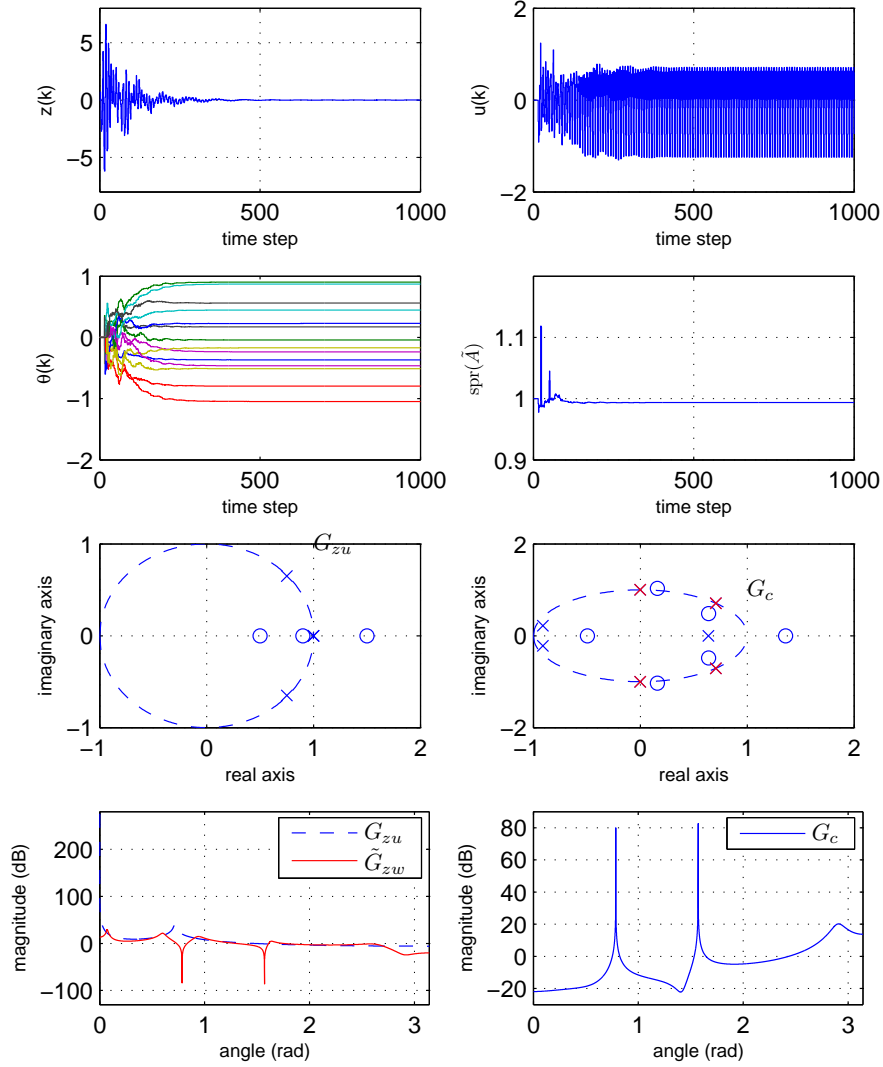


Figure 24. Example VI.7: Nonminimum-phase, unstable plant, $n = 4$, $d = 1$, disturbance rejection. Two Markov parameters H_1 and H_2 are used in the construction of $\mathcal{H} \in \mathbb{R}^{2 \times 2}$. The controller is turned on at $k = 15$. The controller order is $n_c = 7$ with parameters $\alpha \equiv 0.1$, and $\eta_0 = 0.25$. The performance $z(k)$ converges to zero in about 250 steps. RCAC converges to a stabilizing internal model controller with poles at the disturbance frequencies $\pi/2$ rad/sample and $\pi/4$ rad/sample. After convergence, the closed-loop system is asymptotically stable with spectral radius $\text{spr}(\tilde{A}) = 0.99$.

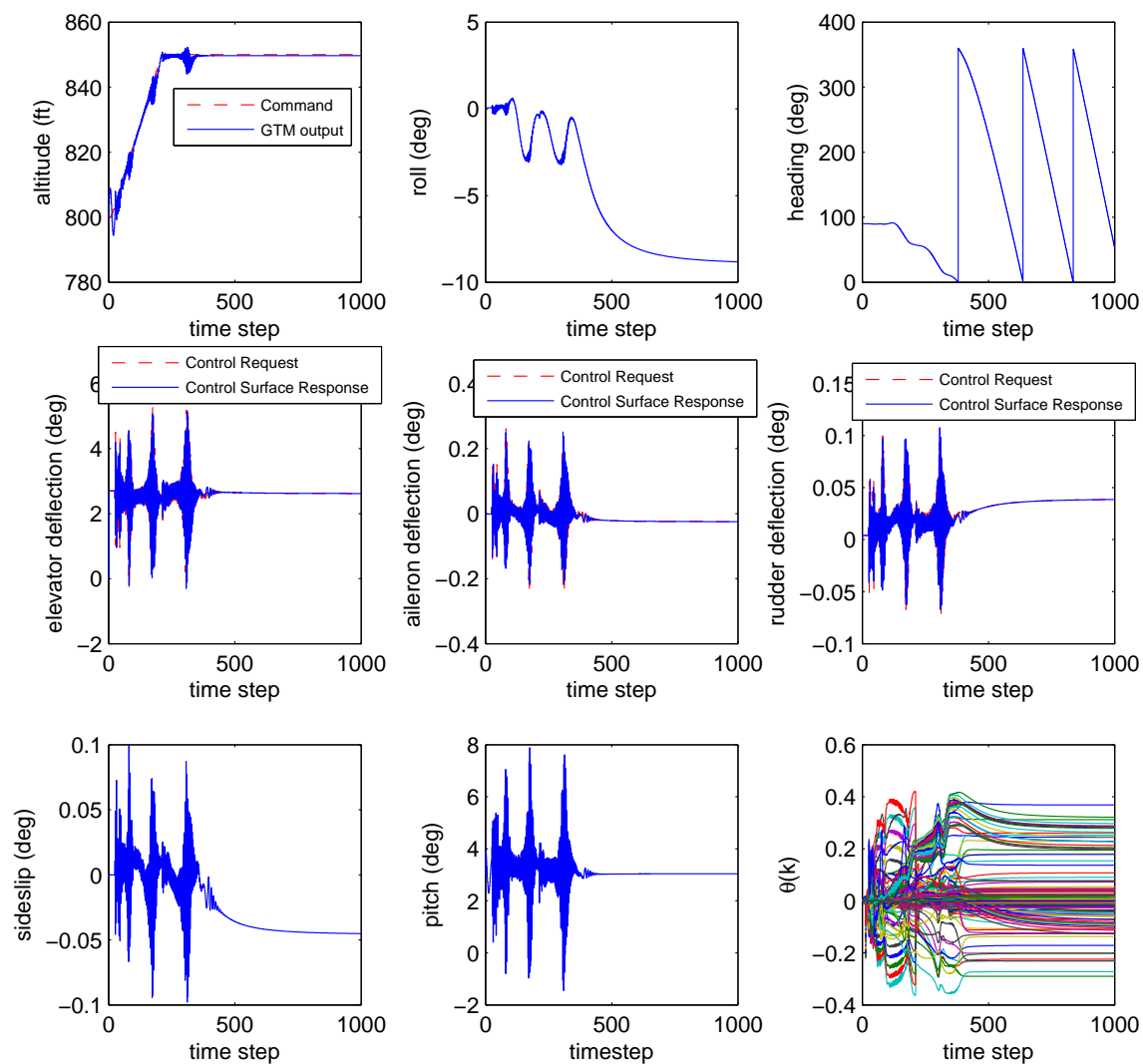


Figure 25. Example VIII.1: Altitude command following with $n_c = 16$, $P_0 = 0.1I_{5n_c}$, $p_c = 10$, $\eta_0 = 0.1$, and $\tilde{\mathcal{H}} = [H_4^T \ H_3^T \ H_2^T]^T \in \mathbb{R}^{3 \times 3}$. The Markov parameters are taken from the GTM model linearized at the trim conditions. The controller gains converge, and the altitude follows the commanded trajectory. Roll and heading are not commanded in this example.

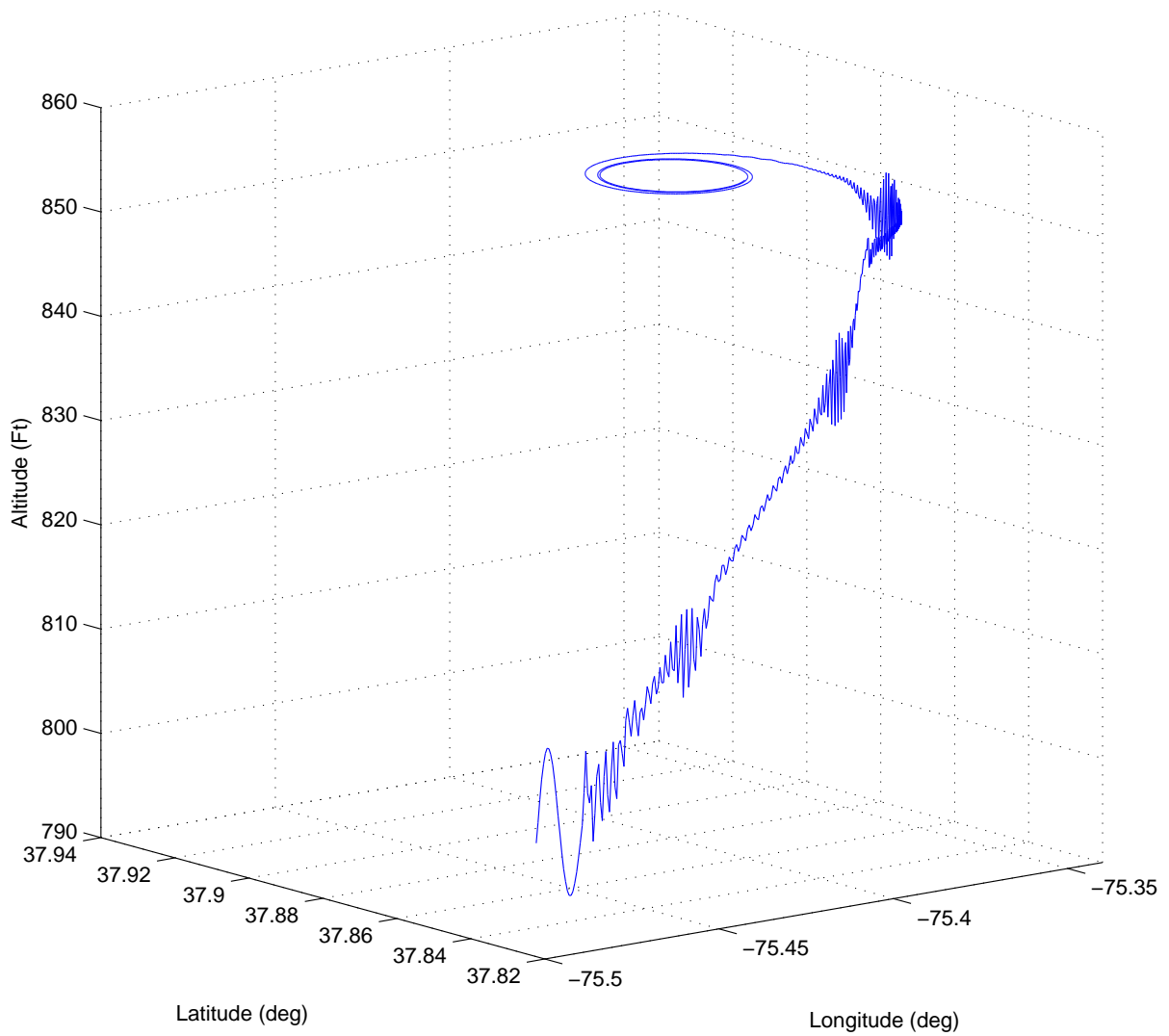


Figure 26. Example VIII.1: Altitude command following trajectory. The aircraft reaches the commanded altitude of 850 ft. Roll and heading are not commanded in this example.

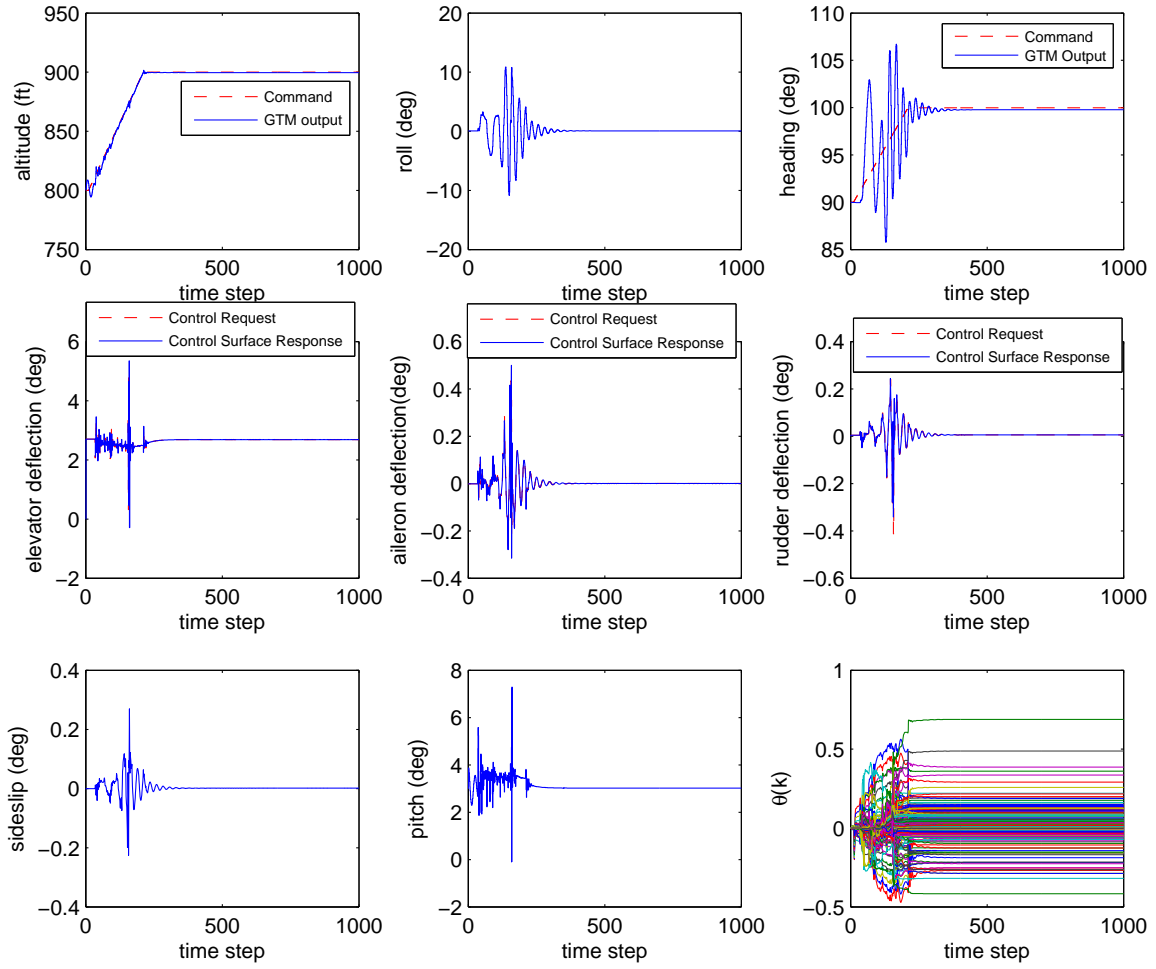


Figure 27. Example VIII.2: Altitude and heading command following with $n_c = 16$, $P_0 = 0.1I_{7n_c}$, $p_c = 10$, $\eta_0 = 0.1$, and $\tilde{\mathcal{H}} = [H_4^T \ H_3^T]^T \in \mathbb{R}^{4 \times 3}$. The controller gains converge, and the altitude and heading reach the commanded values 900 ft and 100 deg, respectively, in 20 sec. Roll is not commanded in this example.

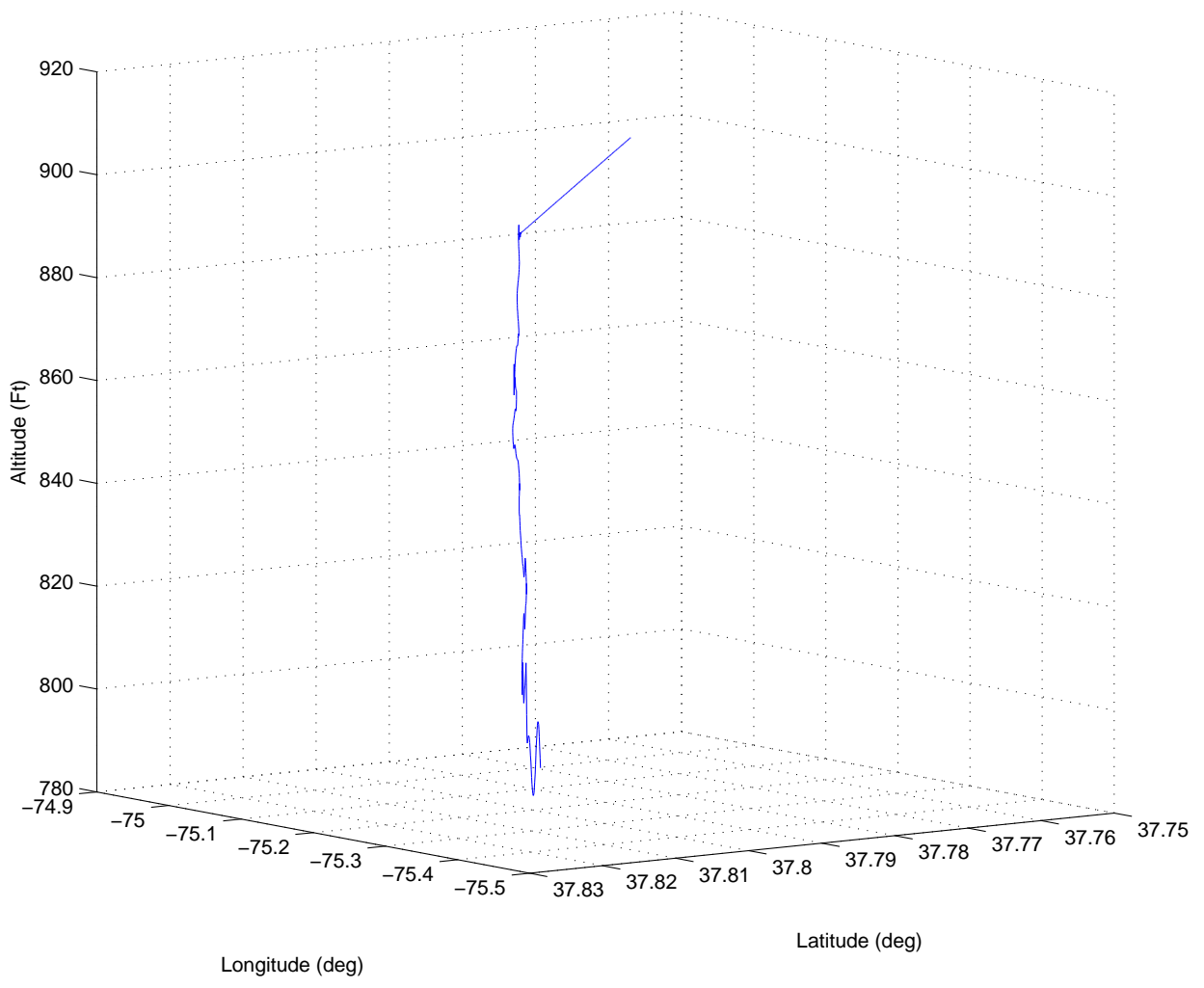


Figure 28. Example VIII.2: Altitude and heading command following trajectory. The aircraft climbs to 900 ft and follows the heading of 100 deg.

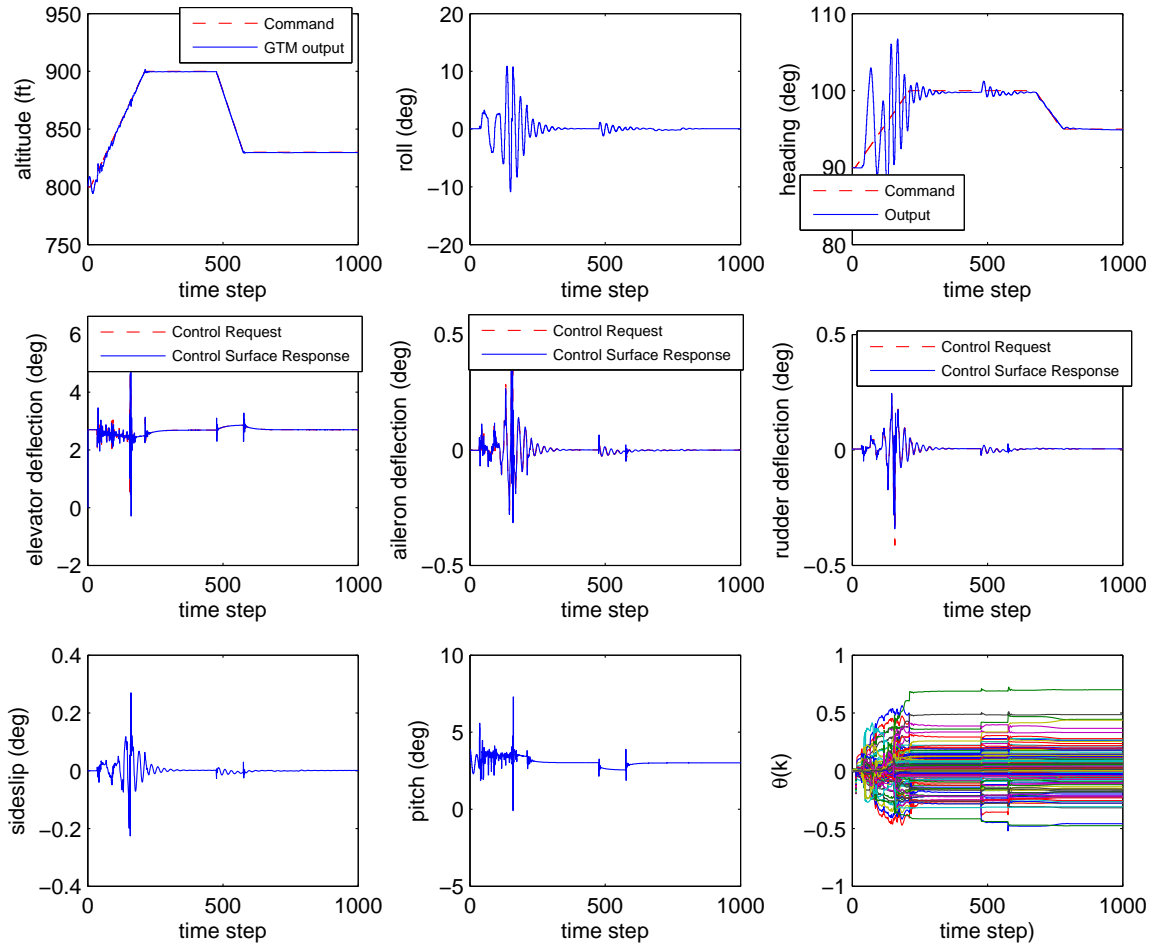


Figure 29. Example VIII.3: Altitude and heading command following with $n_c = 16$, $P_0 = 0.1I_{7n_c}$, $p_c = 10$, $\eta_0 = 0.1$, and $\tilde{\mathcal{H}} = [H_4^T \ H_3^T]^T \in \mathbb{R}^{4 \times 3}$. The controller gains adapt to the changes in command, and the aircraft follows the altitude and heading commands. Roll is not commanded in this example.

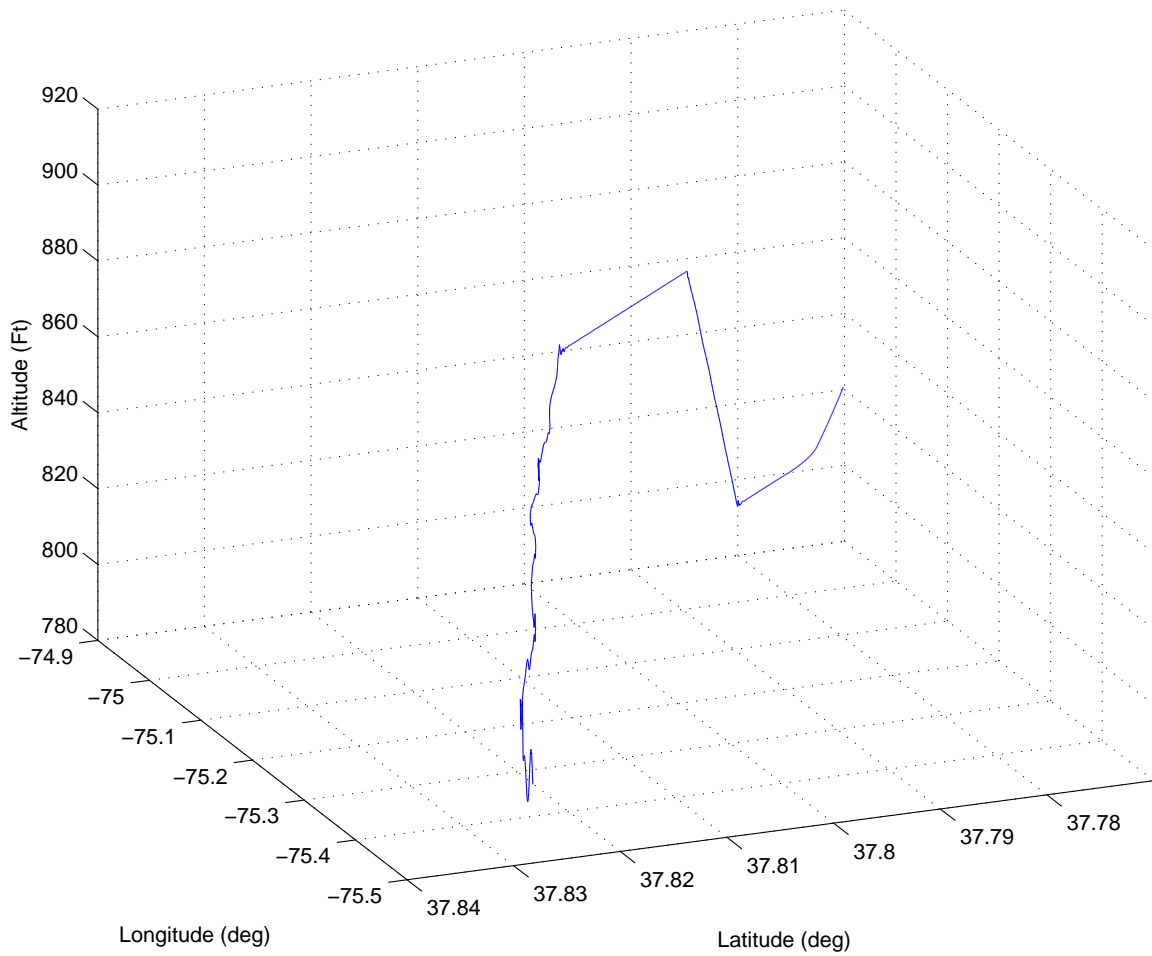


Figure 30. Example VIII.3: Altitude and heading command following trajectory. The aircraft climbs to 900 ft, changes the heading course 100 deg, dives back to 930 ft, and finally changes heading to 95 deg. Roll is not commanded in this example.

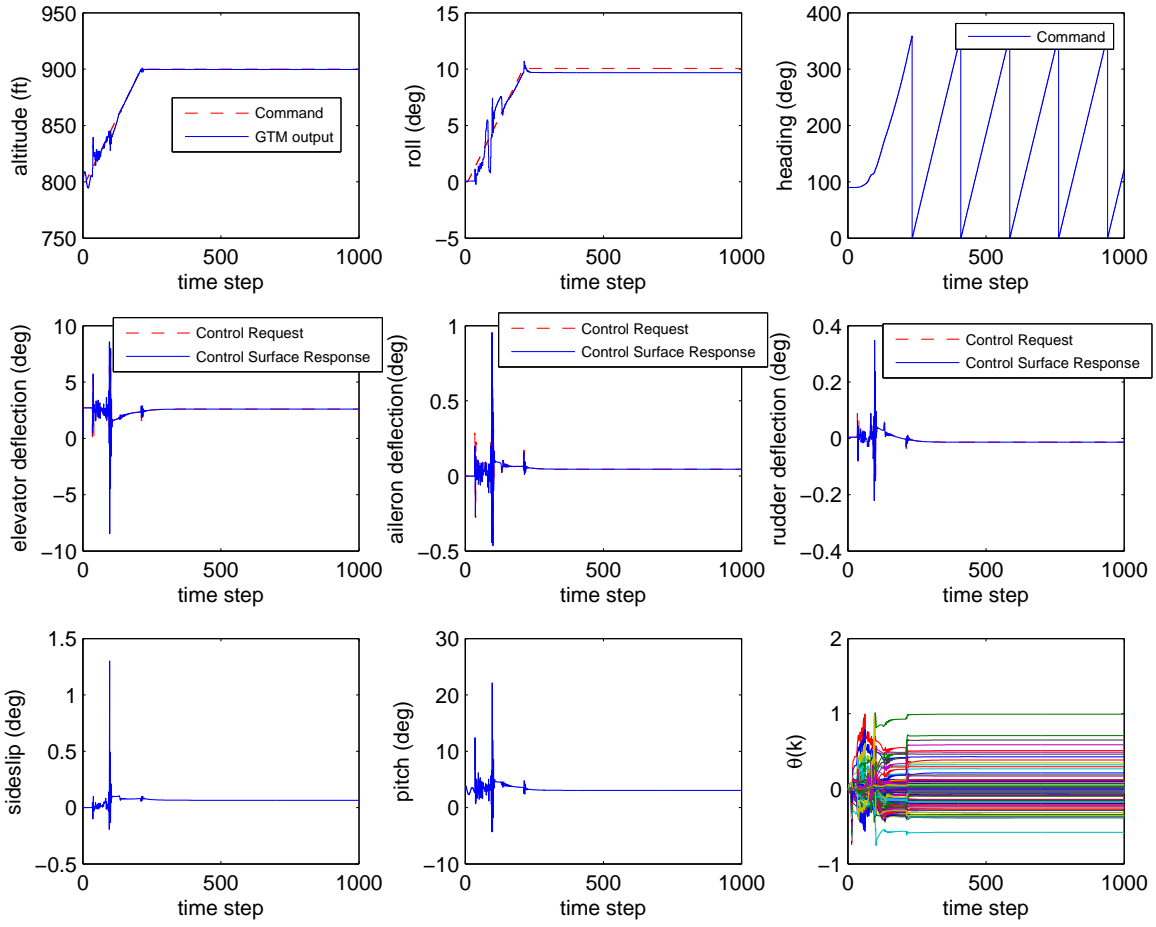


Figure 31. Example VIII.4: Altitude and roll command following with $n_c = 16$, $P_0 = 0.1I_{7n_c}$, $p_c = 10$, $\eta_0 = 0.019$, and $\mathcal{H} = [H_4^T \ H_3^T]^T \in \mathbb{R}^{4 \times 3}$. The controller gains converge, and the altitude and roll reach the commanded values 900 ft and 10 deg, respectively, in 20 sec. Heading is not commanded in this example.

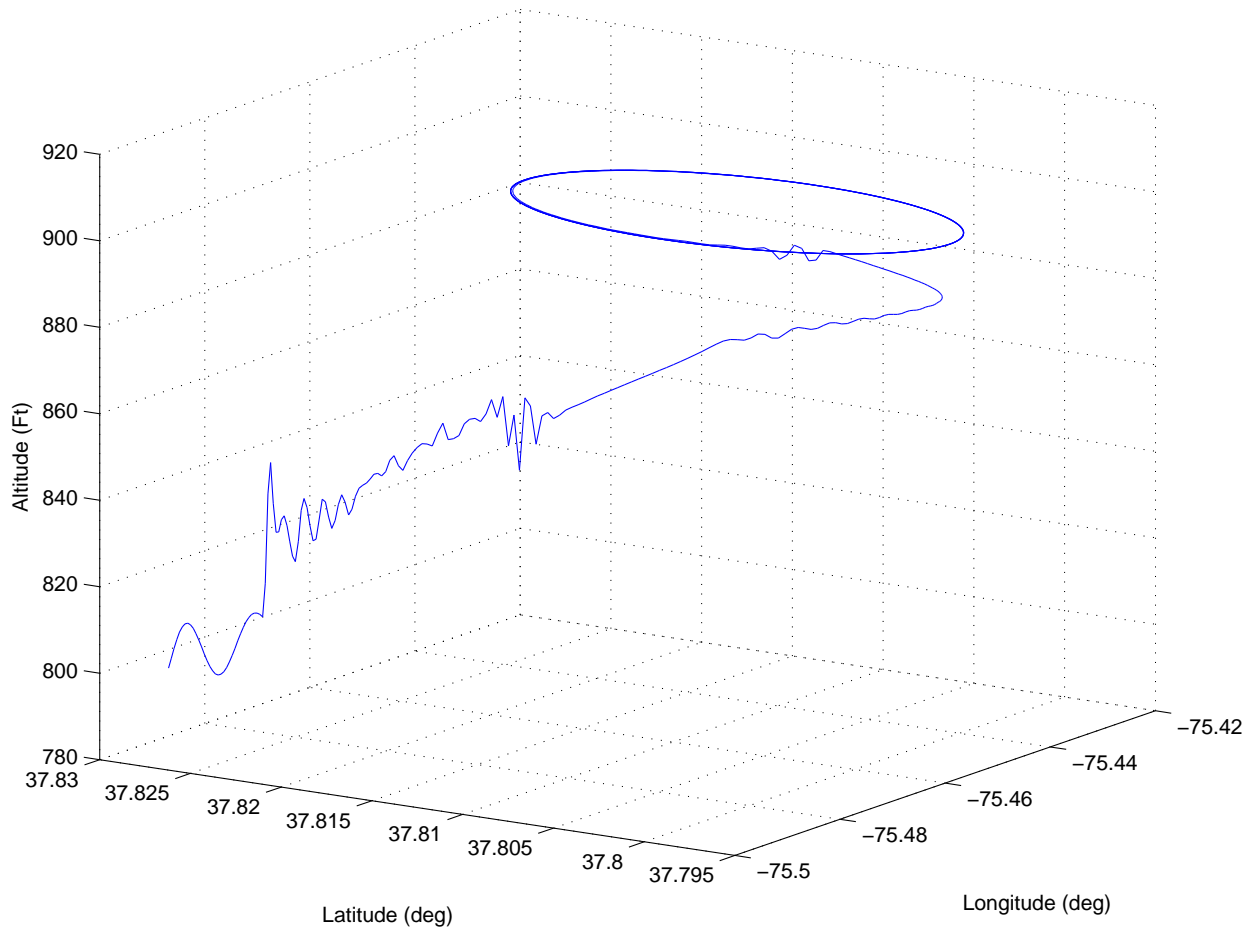


Figure 32. Example VIII.4: Altitude and roll command following trajectory. The aircraft climbs to 900 ft, and follows a closed trajectory due to the constant roll command. Heading is not commanded in this example.

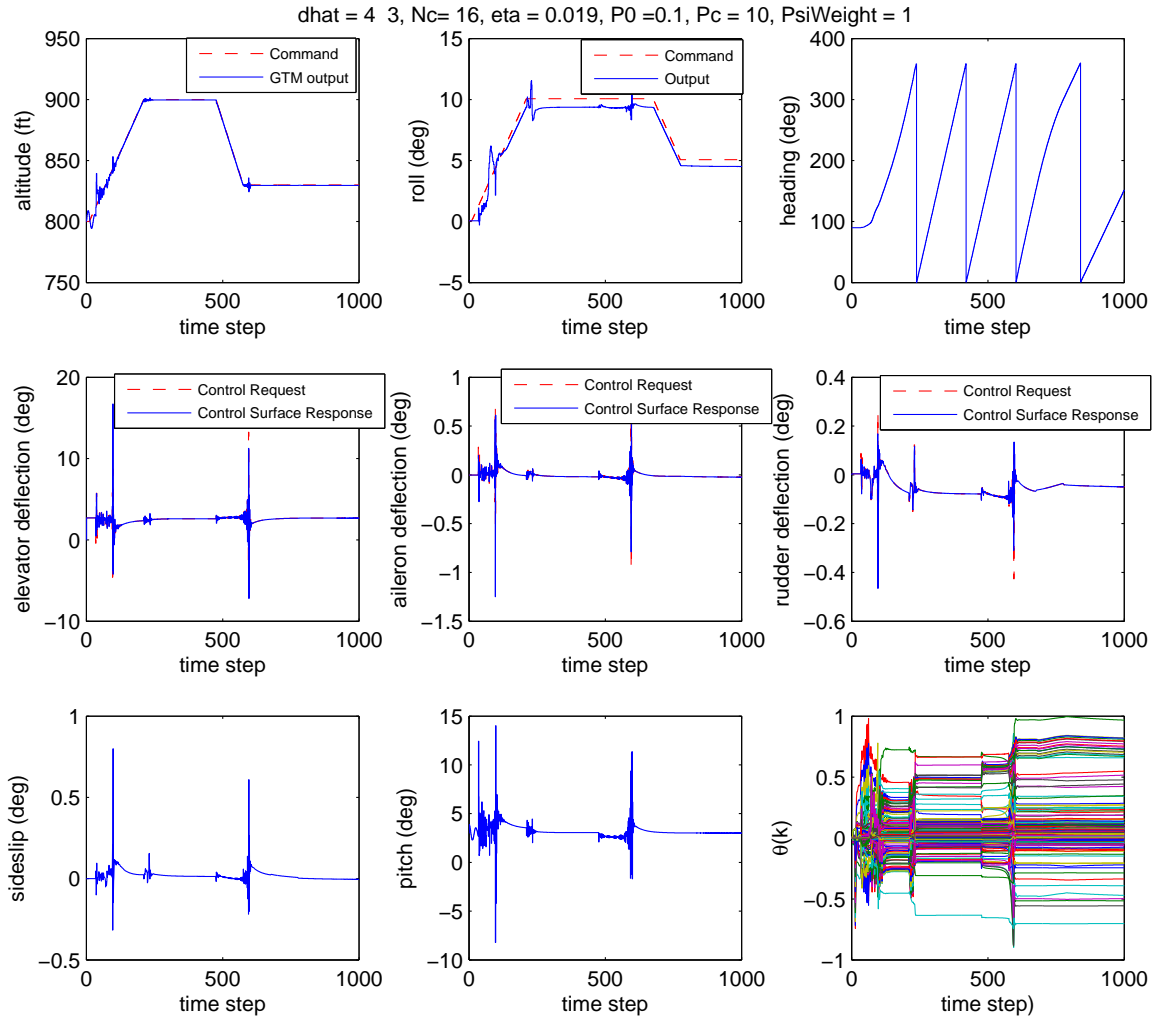


Figure 33. Example VIII.5: Altitude and roll command following with $n_c = 16$, $P_0 = 0.1I_{7n_c}$, $p_c = 10$, $\eta_0 = 0.019$, and $\mathcal{H} = [H_4^T \ H_3^T]^T \in \mathbb{R}^{4 \times 3}$. The controller gains adapt to the time-dependent command, and then the aircraft follows the constant altitude and roll commands with about 5% steady state error in roll. Heading is not commanded in this example.

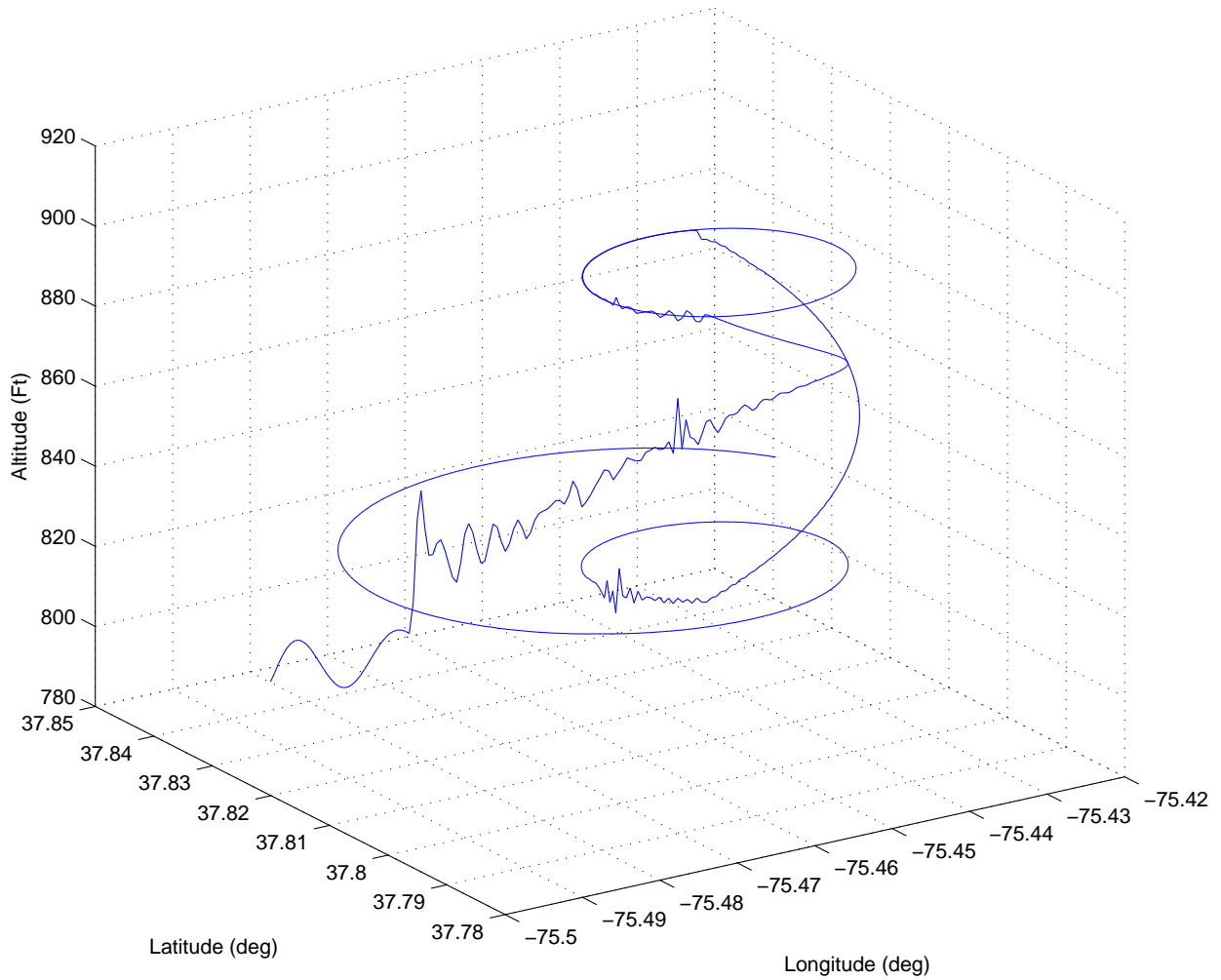


Figure 34. Example VIII.5: Altitude and roll command following trajectory. The aircraft climbs to 900 ft and rolls 10 deg. Next, maintaining the roll trim, the aircraft dives back to 930 ft. Finally, maintaining the altitude, the aircraft changes the roll trim from 10 deg to 5 deg. Heading is not commanded in this example.

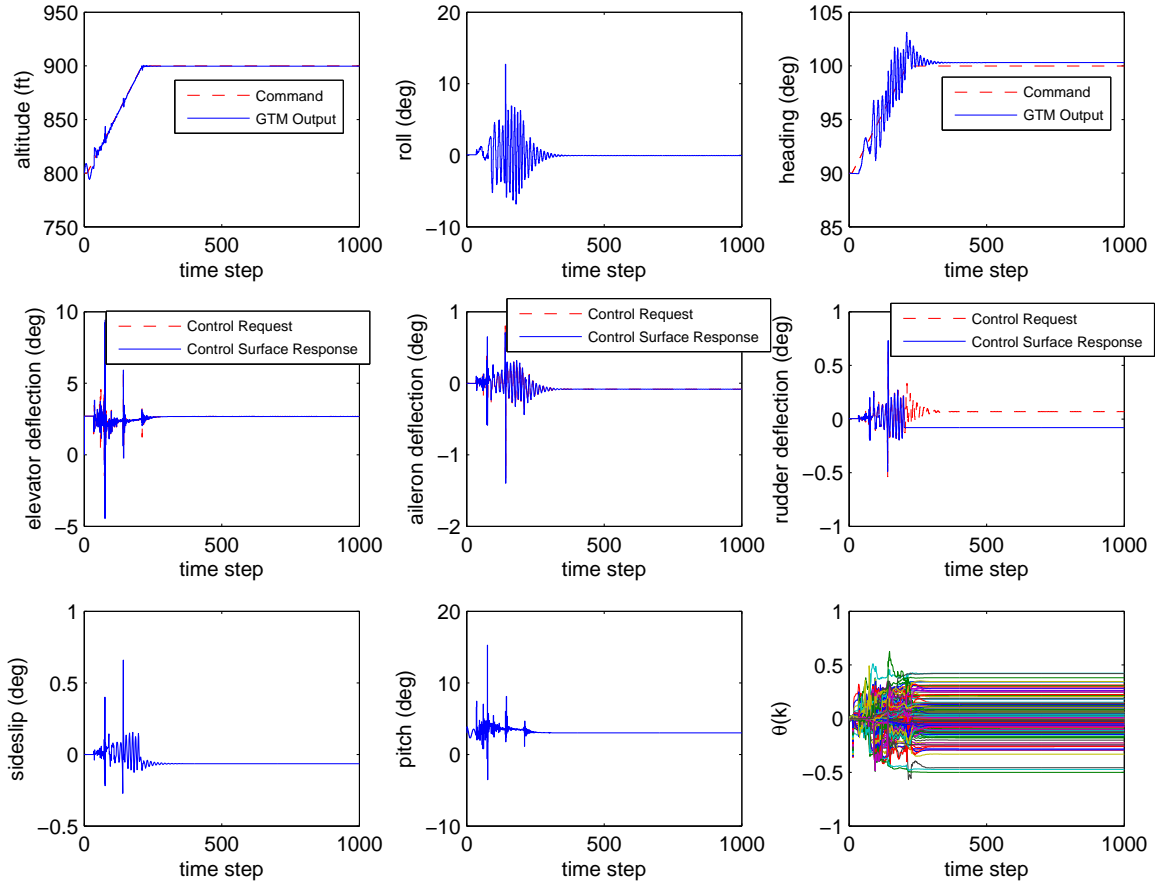


Figure 35. Example VIII.6: Altitude and heading command following with rudder damage, $n_c = 16$, $P_0 = 0.1I_{7n_c}$, $p_c = 10$, $\eta_0 = 0.05$, and $\mathcal{H} = [H_4^T \ H_3^T]^T \in \mathbb{R}^{4 \times 3}$. The controller gains converge, and the aircraft follows the altitude and heading commands with about 3% steady state error in heading. RCAC attempts to compensate for the off-nominal rudder angle by trimming the ailerons to -0.08 deg (compare with Figure 27). Roll is not commanded in this example.

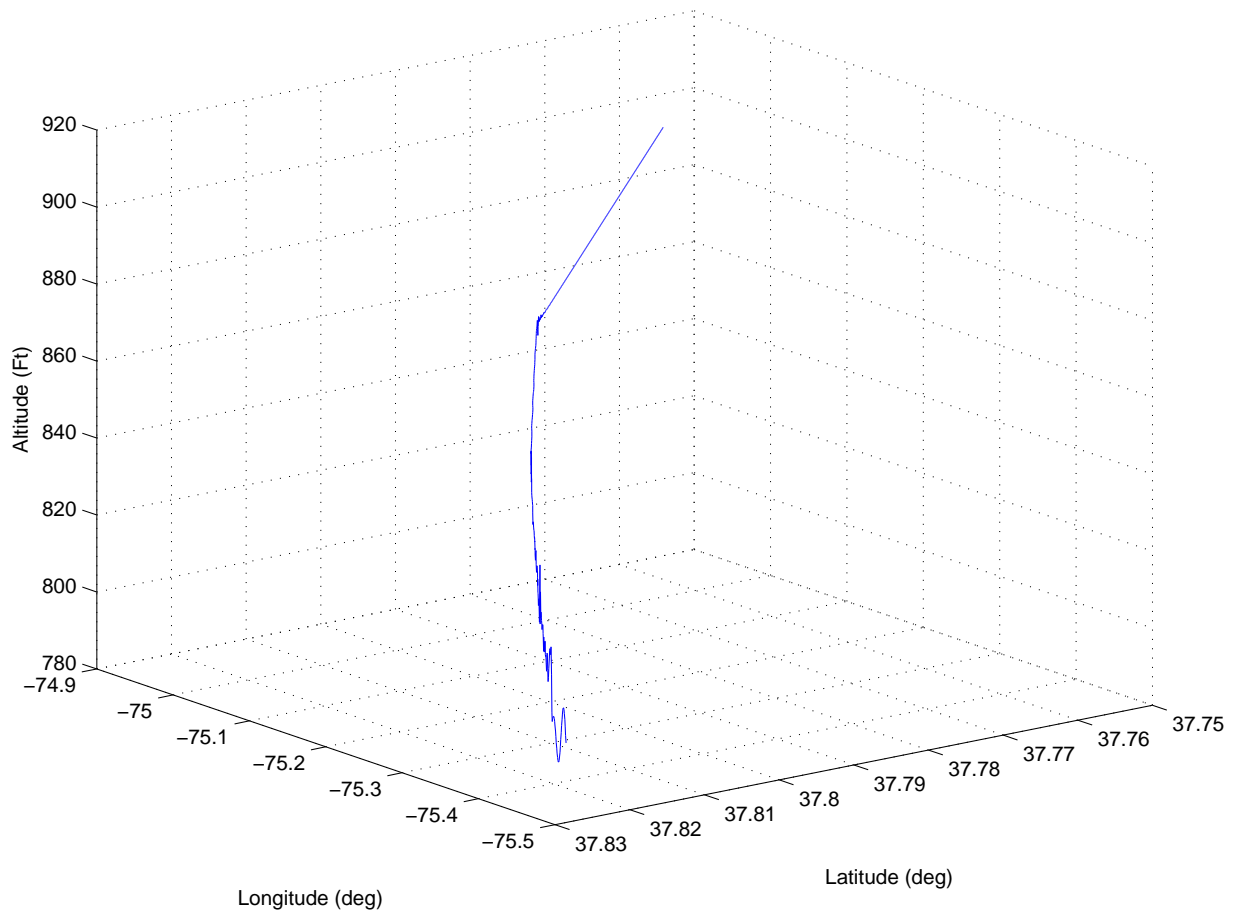


Figure 36. Example VIII.6: Altitude and heading command following trajectory with rudder damage. The aircraft climbs to 900 ft and yaws to attain a heading of 100 deg. The heading course is not affected by the rudder damage. Roll is not commanded in this example.

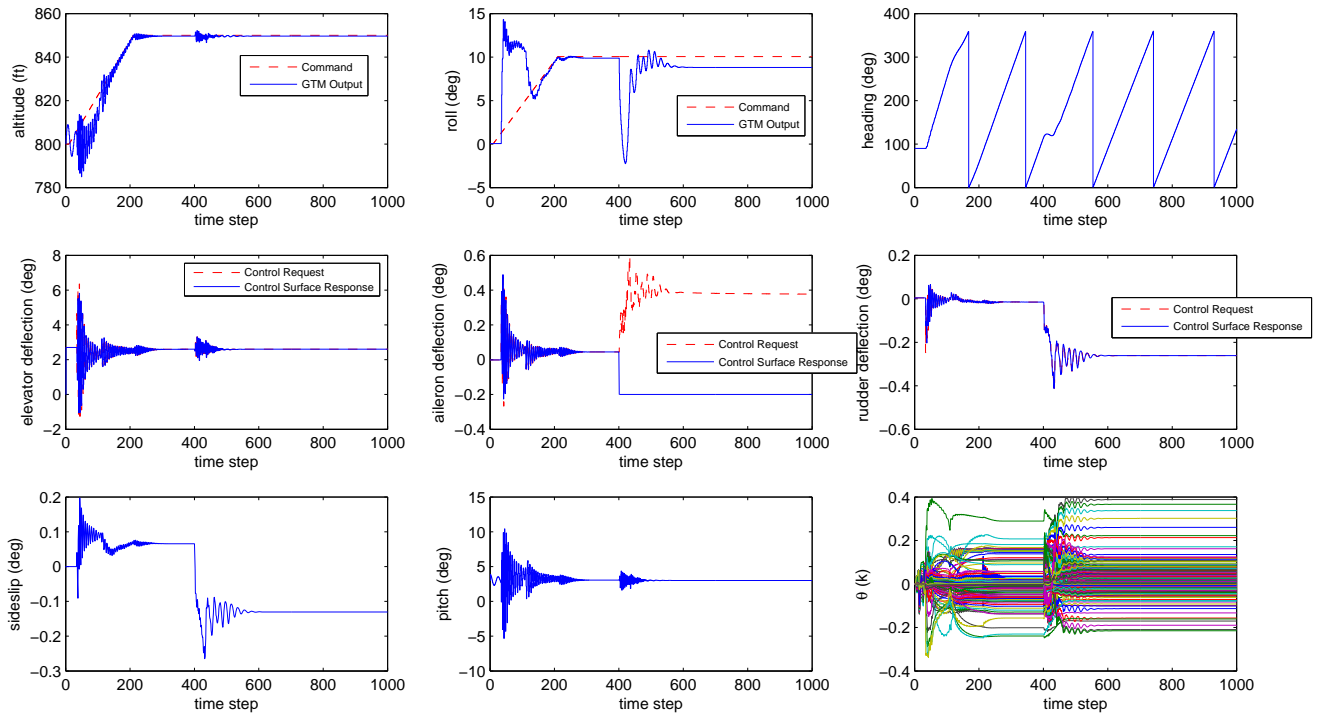


Figure 37. Example VIII.7: Altitude and roll command following with aileron damage, $n_c = 16$, $P_0 = 0.01I_{7n_c}$, $p_c = 10$, $\eta_0 = 0.019$, and $\tilde{\mathcal{H}} = [H_4^T \ H_3^T]^T \in \mathbb{R}^{4 \times 3}$. The controller gains converge, and the aircraft follows the altitude and roll commands with about 1.5 deg steady state error in roll, despite the aileron damage after 40 sec. Heading is not commanded in this example.

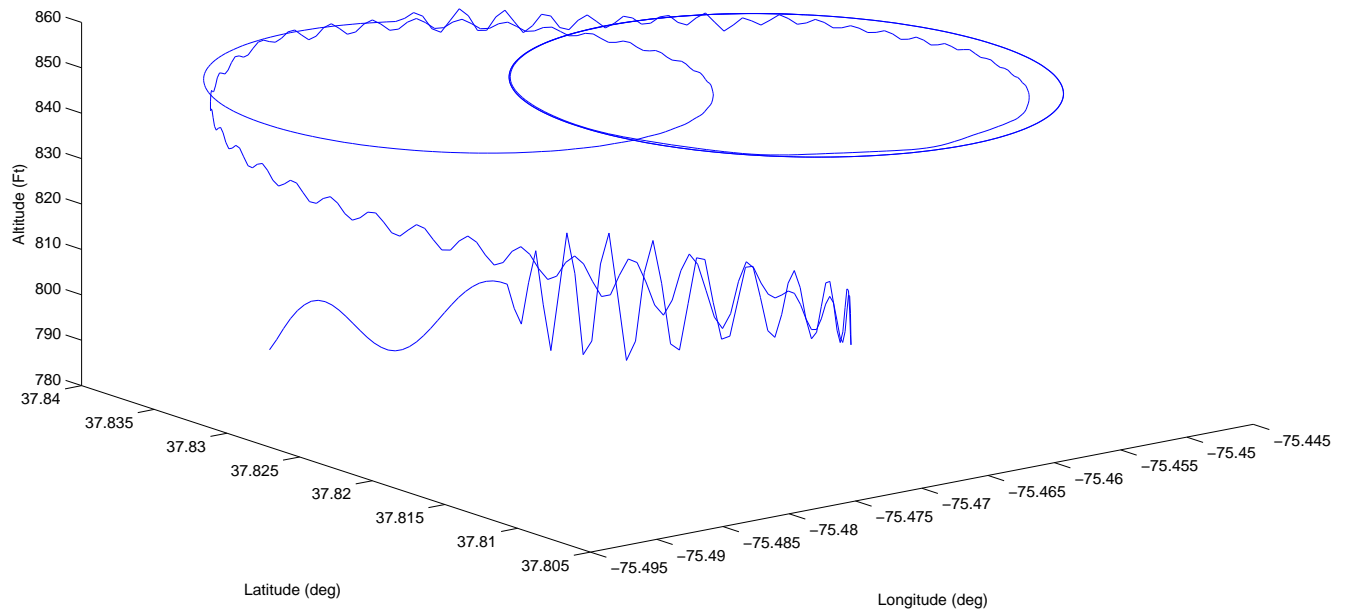


Figure 38. Example VIII.7: Altitude and heading command following trajectory with aileron damage. The aircraft climbs to 900 ft, rolls about 10 deg, experiences aileron damage, and finally reaches a roll angle of about 8.5 deg. Heading is not commanded in this example.

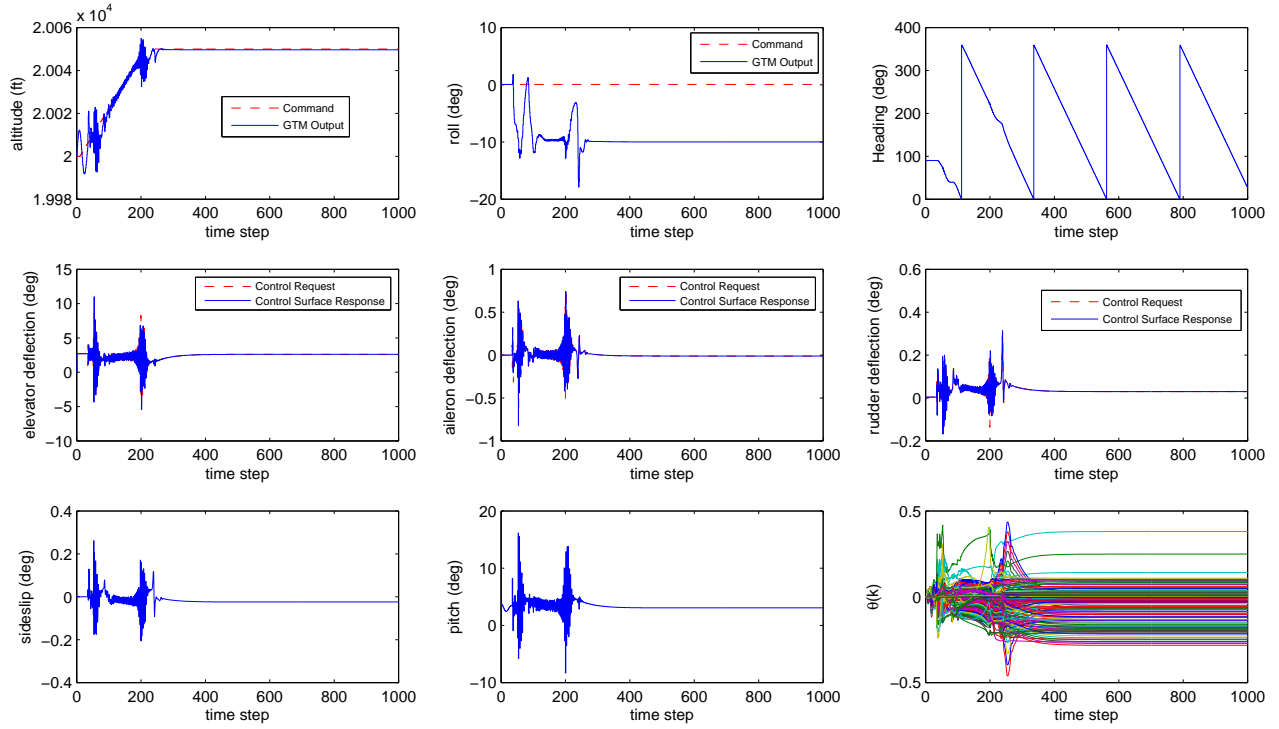


Figure 39. Example VIII.8: Altitude and roll command following with biased roll measurements, $n_c = 16$, $P_0 = 0.01I_{7n_c}$, $p_c = 10$, $\eta_0 = 0.019$, and $\tilde{\mathcal{H}} = [H_4^T \ H_3^T]^T \in \mathbb{R}^{4 \times 3}$. The controller gains converge, and the aircraft follows altitude and roll commands with about 10 deg steady state error in roll, caused by the biased roll measurement. Heading is not commanded in this example.

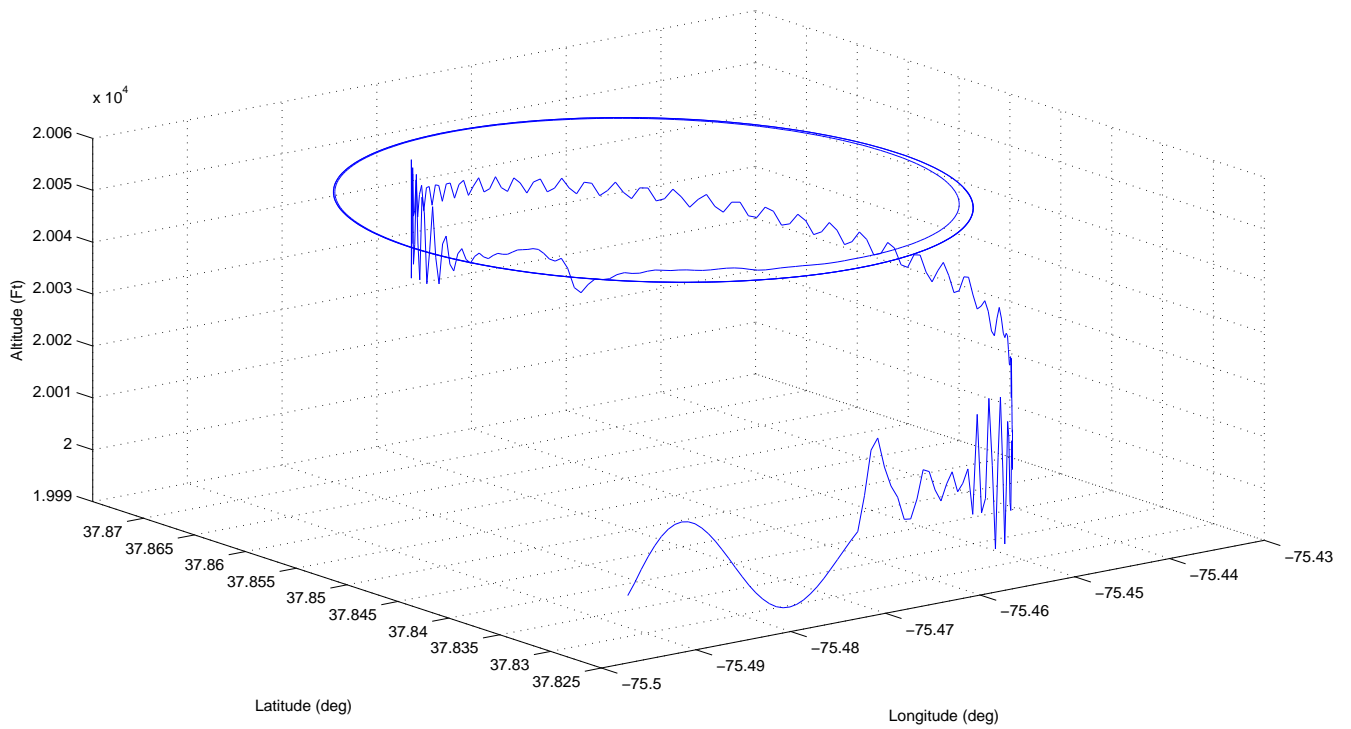


Figure 40. Example VIII.8: Altitude and roll command following trajectory with biased roll measurements. The aircraft climbs to 20050 ft and maintains the new altitude despite the nonzero roll angle caused by the biased roll measurement. Heading is not commanded in this example.

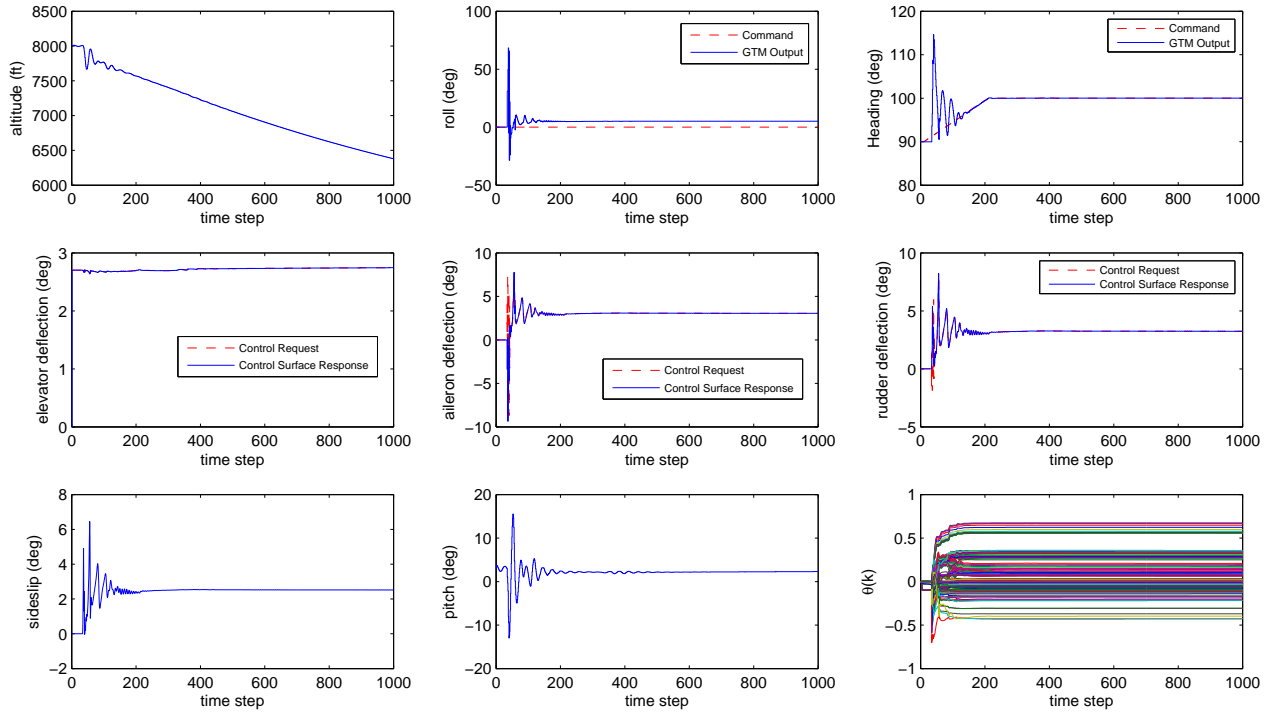


Figure 41. Example VIII.9: Roll and heading command following with biased roll measurements, $n_c = 16$, $P_0 = 0.01I_{7n_c}$, $p_c = 10$, $\eta_0 = 0.01$, and $\mathcal{H} = [H_4^T \ H_3^T]^T \in \mathbb{R}^{4 \times 3}$. The controller gains converge, and the aircraft follows the roll and heading commands with about 5 deg steady state error in roll, caused by the biased roll measurement. Altitude is not commanded in this example.

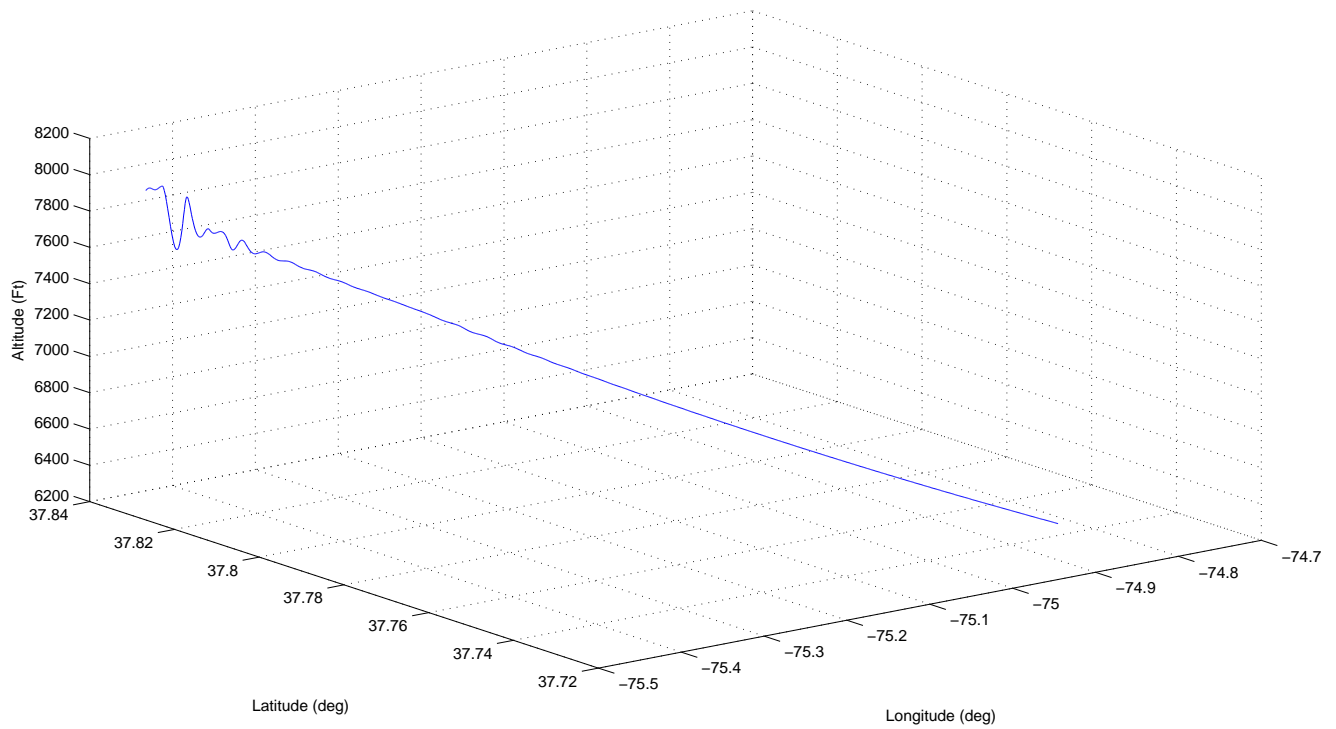


Figure 42. Example VIII.9: Roll and heading command following trajectory with biased roll measurements. The aircraft yaws to attain a heading of 100 deg and maintains the new heading course despite the nonzero roll angle caused by the biased roll measurement. Altitude is not commanded in this example.

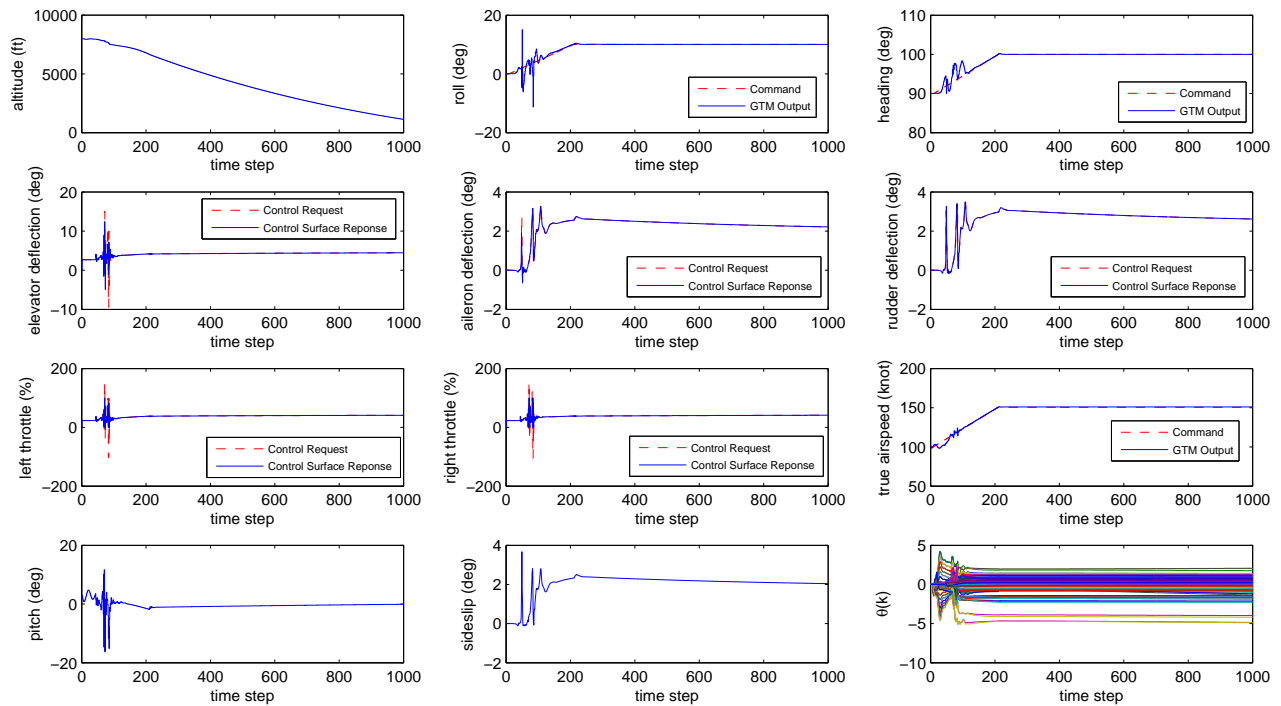


Figure 43. Example VIII.10: Roll, heading and airspeed command following with decentralized RCAC architecture using two separate RCAC blocks. We take $n_c = 16$, $P_0 = 0.01I$, $p_c = 10$, $\eta_0 = 0.01$, and $\tilde{C} = H_4$ for both RCAC blocks. The controller gains converge, and the aircraft follows the roll, heading and airspeed commands with zero steady state error in each channel. Altitude is not commanded in this example.

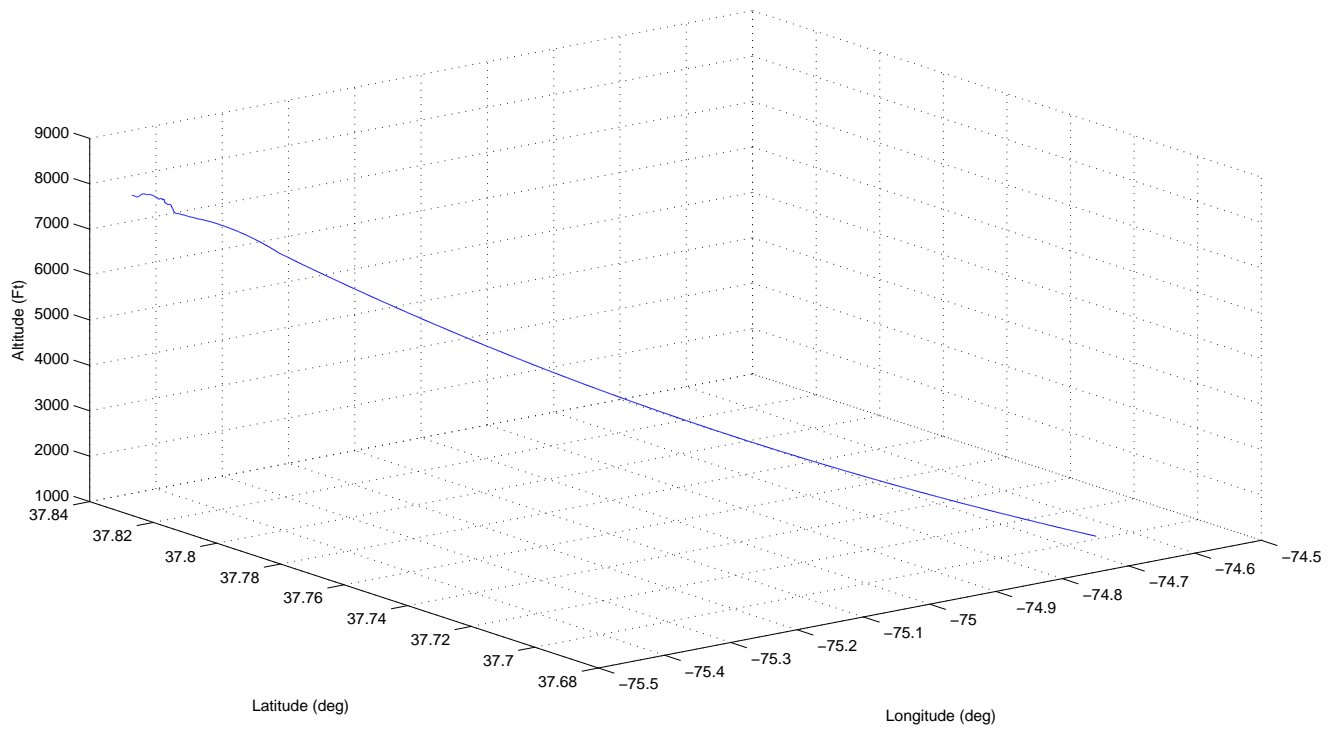


Figure 44. Example VIII.10: Roll, heading and airspeed command following trajectory. The aircraft yaws to attain a heading of 100 deg while rolling 10 deg and increasing the airspeed to 150 knots. Altitude is not commanded in this example.

Acknowledgments

This work was supported in part by the NASA IRAC Grant NNX08AB92A. We wish to thank Ming-Jui Yu for assistance in producing the RCAC GTM results in this paper.

References

- ¹K. J. Astrom and B. Wittenmark, *Adaptive Control*, 2nd ed., Addison-Wesley, 1995.
- ²G. C. Goodwin and K. S. Sin, *Adaptive Filtering, Prediction and Control*, Prentice Hall, 1984.
- ³P.A Ioannou, and J. Sun, *Robust Adaptive Control*, Prentice Hall, 1996.
- ⁴G. Tao, *Adaptive Control Design and Analysis*, Wiley, 2003.
- ⁵R. Venugopal and D. S. Bernstein. "Adaptive Disturbance Rejection Using ARMARKOV System Representations," *IEEE Trans. Contr. Sys. Tech.*, Vol. 8, pp. 257–269, 2000.
- ⁶H. Sane, R. Venugopal, and D. S. Bernstein, "Disturbance Rejection Using Self-Tuning ARMARKOV Adaptive Control with Simultaneous Identification," *IEEE Trans. Contr. Sys. Tech.*, Vol. 9, pp. 101-106, 2001.
- ⁷J. B. Hoagg, M. A. Santillo, and D. S. Bernstein, "Discrete-Time Adaptive Command Following and Disturbance Rejection for Minimum Phase Systems with Unknown Exogenous Dynamics," *IEEE Trans. Autom. Contr.*, Vol. 53, pp. 912–928, 2008.
- ⁸M. A. Santillo and D. S. Bernstein, "Adaptive Control Based on Retrospective Cost Optimization," *AIAA J. Guid. Contr. Dyn.*, Vol. 33, pp. 289–304, 2010.
- ⁹R. J. Fuentes, J. B. Hoagg, B. J. Anderton, A. M. D'Amato, and D. S. Bernstein "Investigation of Cumulative Retrospective Cost Adaptive Control for Missile Application," AIAA Guid. Nav. Contr. Conf., Toronto, August 2010, AIAA 2010-7577.
- ¹⁰J. B. Hoagg and D. S. Bernstein, "Retrospective Cost Adaptive Control for Nonminimum-Phase Discrete-Time Systems Part 1: The Ideal Controller and Error System, Part 2: The Adaptive Controller and Stability Analysis," *Proc. Conf. Dec. Contr.*, pp. 893–904, Atlanta, GA, December 2010.
- ¹¹J. B. Hoagg and D. S. Bernstein, "Retrospective Cost Model Reference Adaptive Control for Nonminimum-Phase Discrete-Time Systems, Part 1: The Ideal Controller and Error System," *Proc. Conf. Dec. Contr.*, pp. 2933–2938, San Francisco, CA, June 2011.
- ¹²J. B. Hoagg and D. S. Bernstein, "Retrospective Cost Model Reference Adaptive Control for Nonminimum-Phase Discrete-Time Systems, Part 2: The Adaptive Controller and Stability Analysis," *Proc. Conf. Dec. Contr.*, pp. 2927–2932, San Francisco, CA, June 2011.
- ¹³A. V. Morozov, J. B. Hoagg, and D. S. Bernstein, "Retrospective Adaptive Control of a Planar Multilink Arm with Nonminimum-Phase Zeros," *Proc. Conf. Dec. Contr.*, pp. 3706–3711, Atlanta, GA, December 2010.
- ¹⁴A. V. Morozov, J. B. Hoagg, and D. S. Bernstein, "A Computational Study of the Performance and Robustness Properties of Retrospective Cost Adaptive Control," AIAA Guid. Nav. Contr. Conf., Toronto, August 2010, AIAA-2010-8011-911.
- ¹⁵B. C. Coffey, J. B. Hoagg, and D. S. Bernstein, "Retrospective Cost Adaptive Control of the NASA GTM Model," AIAA Guid. Nav. Contr. Conf., Toronto, August 2010, AIAA-2010-8404-212.
- ¹⁶A. M. D'Amato, E. D. Sumer, and D. S. Bernstein, "Retrospective Cost Adaptive Control for Systems with Unknown Nonminimum-Phase Zeros," AIAA Guid. Nav. Contr. Conf., Portland, OR, August 2011.