JOURNAL OF AEROSPACE INFORMATION SYSTEMS
Vol. 14, No. 3, March 2017

Retrospective Cost Adaptive Control of Generic Transport Model Under Uncertainty and Failure

Ahmad Ansari* and Dennis S. Bernstein†
University of Michigan, Ann Arbor, Michigan 48109

DOI: 10.2514/1.I010454

This paper presents a case study on the application of retrospective cost adaptive control to the NASA generic transport model under conditions of uncertainty and failure. To apply retrospective cost adaptive control to the generic transport model, a collection of control architectures is defined, where each architecture has a decentralized adaptation structure in the sense that a performance (error) variable is assigned to each control channel for online decentralized adaptation. Sensor signals may be shared among the channels to account for channel coupling. Then, a series of cases that examine the performance of retrospective cost adaptive control under various conditions is considered, including trim finding, startup, unknown time-varying aircraft dynamics (such as icing and mass variation), unknown flight envelope (including the possibility of conflicting commands), known sensing constraints (a failed air data system), known actuation constraints (requiring differential thrust), unknown sensor failure (stuck and biased airspeed measurement), and unknown actuator failure (severe stroke and rate saturation, dead zone, and jam). For each case, a range of scenarios is considered that captures various commands and performance objectives. Disturbance rejection is included in the form of unknown wind shear.

Nomenclature

$\boldsymbol{a}$	=	ailerons
$\vec{a}$	=	desired path on ground
$\boldsymbol{e}$	=	elevator
$e_{\parallel}$	=	signed distance error parallel to $\vec{a}$
$e_{\perp}$	=	signed distance error perpendicular to $\vec{a}$
$\mathrm{F_E}$	=	Earth frame
$\mathrm{F_{AC}}$	=	aircraft body-fixed frame
$\mathrm{G_f}$	=	filter transfer function applied to ϕ and u
h	=	altitude
i	=	channels 1–4 for $\boldsymbol{T}$, $\boldsymbol{e}$, $\boldsymbol{a}$, and $\boldsymbol{r}$, respectively
k_0	=	relative degree of controller
n_c	=	order of controller
P, Q, R	=	angular velocity of F_{AC} relative to F_E in direction $\hat{\imath}_{\mathrm{AC}}, \hat{\jmath}_{\mathrm{AC}}, \hat{k}_{\mathrm{AC}}$
R_u	=	control penalty
R_z	=	performance penalty
R_θ	=	controller-coefficient penalty
$\boldsymbol{r}$	=	rudder
$\boldsymbol{T}$	=	throttle
U, V, W	=	aircraft speed relative to air in direction $\hat{\imath}_{\mathrm{AC}}, \hat{\jmath}_{\mathrm{AC}}, \hat{k}_{\mathrm{AC}}$
u	=	input
$u_{i,\mathrm{actual}}, u_{i,\mathrm{req}}$	=	ith actual/requested actuator setting
$u_{i,\mathrm{trim}}$	=	ith constant actuator setting for trim flight
V_{AC}	=	airspeed
V_{g}	=	ground speed
X	=	north coordinate
Y	=	east coordinate
y	=	measurement
y_{trim}	=	measurement at trim flight
z	=	performance variable
$\boldsymbol{z}$	=	Z transform
α	=	angle of attack
β	=	sideslip angle
γ	=	flight-path angle
$\delta u_{i,\mathrm{actual}}$	=	ith actual actuator setting increment
$\delta u_{i,\mathrm{req}}$	=	ith requested actuator setting increment specified by the adaptive controller
δy	=	measurement increment from the trim flight
δy_{cmd}	=	incremental command from the trim flight

Received 2 February 2016; revision received 12 October 2016; accepted for publication 13 January 2017; published online 10 March 2017.

*Ph.D. Candidate, 1320 Beal Avenue, Aerospace Engineering Department.
†Professor, 1320 Beal Avenue, Aerospace Engineering Department.

η	=	ground velocity alignment error with $\vec{a}$
Θ, Φ, Ψ	=	pitch, roll, yaw angles
θ	=	controller coefficients
τ	=	turn rate
ϕ	=	feedback vector

I. Introduction

ALONG with aerodynamics, lightweight structures, and propulsion, feedback control is one of the key technologies that enable aviation. Feedback control uses inertial and noninertial sensors to provide measurements of position, velocity, acceleration, attitude, and angular rates to specify thrust and aerodynamic surfaces to apply corrective forces and moments. Closed-loop control enables the operation of autopilots, which can either assist the pilot or assume complete control of aircraft operation.

The standard approach to feedback control of aircraft is based on classical control techniques [1–3]. With the advent of optimal control methods, state-space-based control techniques have also been successful [4,5]. Although classical optimal control does not account for model uncertainty, aircraft flight control has benefited from advances in robust control [6,7].

For conventional aircraft operating under emergency flight conditions, as well as for unconventional aircraft, recent research has focused on adaptive control techniques [8–13]. The failure of the Honeywell MH-96 self-adaptive controller used on the X-15-3 was analyzed in [12], and L_1 adaptive control with an uncertain flight envelope has been flight tested on the NASA AirStar scaled aircraft [13].

The goal of the present paper is to investigate the performance of an alternative technique for adaptive flight control, specifically, retrospective cost adaptive control (RCAC). RCAC is a direct discrete-time adaptive control technique for stabilization, command following, and disturbance rejection [14]. As a discrete-time approach, RCAC is motivated by the desire to implement control algorithms that operate at the sensor sample rate without the need for controller discretization. This also means that the required modeling information can be estimated based on data sampled at the same rate as the control update.

RCAC was originally motivated by the notion of retrospectively optimized control, where past controller coefficients used to generate past control inputs are reoptimized in the sense that if the reoptimized coefficients had been used over a previous window of operation, then the performance would have been better. Unlike signal processing applications such as estimation and identification, however, it is impossible to change past control inputs, and thus the reoptimized controller coefficients are used only to generate the next control input.

RCAC was originally developed within the context of active noise control experiments [15]. The algorithm used in [15] is gradient based, where the gradient direction and step size are based on different cost functions. In subsequent work [16], the gradient algorithm was replaced by batch least-squares optimization. In both [15,16], the modeling information is given by Markov parameters (impulse response components) of the open-loop transfer function G_{zu} from the control input u to the performance variable z. More recently, in [17], a recursive least-squares algorithm was used, along with knowledge of the nonminimum-phase (NMP) zeros of G_{zu}. The approaches in [15–17] are closely related in the sense that all of the NMP zeros outside of the spectral radius of G_{zu} are approximate zeros of a polynomial, for which the coefficients are Markov parameters of G_{zu}; this polynomial is a truncated Laurent expansion of G_{zu} about infinity. RCAC uses a filter G_f to define the retrospective cost by filtering the difference between the actual past control inputs and the reoptimized control inputs. To construct G_f, Markov parameters are used in [15,16,18] and NMP zeros are used in [17,19].

In the present paper, we interface RCAC with the NASA generic transport model (GTM) simulation model [20–23]. This aircraft model includes an aerodynamic database, trim function, and interface to facilitate feedback control from realistic sensors to thrust and control surfaces. For RCAC, which operates at a fixed sample rate, A/D and D/A operations are implemented through Simulink blocks.

The goal of this paper is to explore the ability of RCAC to control the GTM in various operational scenarios with minimal modeling information. To set the stage for this study, we begin by presenting notation relevant to the GTM simulation; this notation is used to precisely define the control architectures and scenarios considered for adaptive control. A summary of the RCAC algorithm, including the definition of the retrospective cost, controller structure, and modeling information that RCAC requires, is given in Appendix B. The RCAC algorithm is a multiple-input/multiple-output control law, where the performance variable, measurements for feedback, and control inputs may be vectors. Appendix A provides the tuning parameters for the various architectures. Except for the evaluation of warmup strategies, these parameters are fixed for each architecture.

To apply RCAC to the GTM, we define a collection of architectures that assign specified performance (error) variables to control inputs. This means that each RCAC implementation is a decentralized controller, although the individual channel controllers may share measurement signals to account for channel coupling. We then consider a series of cases that examine the performance of RCAC under various conditions, including trim finding, startup, unknown time-varying aircraft dynamics (such as icing and mass variation), unknown flight envelope (including the possibility of conflicting commands), known sensing constraints (such as a failed air data system), known actuation constraints (requiring differential thrust), unknown sensor failure (stuck and biased airspeed measurement), and unknown actuator failure (such as anomalous stroke and rate saturation, dead zone, and jam). Some of these cases are considered in [24–28]. For each case, we consider a range of scenarios that capture various commands and performance objectives. Disturbance rejection is included in the form of unknown wind shear.

All of the applications of RCAC assume extremely limited modeling information about the GTM. In particular, no information about the aerodynamics of the GTM are used. In addition, each of the six architectures considered in this paper uses limited measurements, and none of the architectures assumes that full-state measurements are available. Consequently, full-state-feedback controllers such as linear-quadratic-regulator can be used with these architectures.

An additional distinction between the RCAC controllers and standard practice concerns the use of integrators. Most applications of flight control insert an integrator in each loop. The examples in this paper show that RCAC can automatically develop an integrator in response to trim commands. In addition, many of the scenarios considered in this paper involve commands for transitioning the aircraft from a given trim to another trim. This transition is performed by RCAC under fixed tuning parameters, without manual intervention to facilitate the transition. The goal is to determine the ability of RCAC to perform trim-to-trim transitions without gain scheduling, which is the traditional approach to trim transition, as well as without an inner-/outer-loop controller structure.

II. Assumptions, Notation, and Terminology

The Earth frame and aircraft body-fixed frame are denoted by F_E and F_{AC}, respectively. We assume that F_E is an inertial frame and the Earth is flat. The origin O_E of F_E is any convenient point fixed on the Earth. F_E is defined with the axes $\hat{i}_E$ and $\hat{j}_E$ horizontal, whereas the axis $\hat{k}_E$ points downward. F_{AC} is defined with $\hat{i}_{AC}$ pointing out the nose of the aircraft, $\hat{j}_{AC}$ pointing out the right wing, and $\hat{k}_{AC}$ pointing downward, that is, $\hat{k}_{AC} = \hat{i}_{AC} \times \hat{j}_{AC}$. F_{AC} and F_E are related by

$$\mathrm{F}_{\mathrm{AC}} = \vec{R}_{\mathrm{AC/E}}\mathrm{F}_{\mathrm{E}} \tag{1}$$

where $\vec{R}_{\mathrm{AC/E}}$ is a physical rotation matrix represented by a 3-2-1 Euler rotation sequence, involving two intermediate frames $\mathrm{F}_{\mathrm{E}'}$ and $\mathrm{F}_{\mathrm{E}''}$. In particular,

$$\vec{R}_{\mathrm{AC/E}} = \vec{R}_{\hat{i}_{\mathrm{E}''}}(\Phi)\vec{R}_{\hat{j}_{\mathrm{E}'}}(\Theta)\vec{R}_{\hat{k}_{\mathrm{E}}}(\Psi) \tag{2}$$

where $\mathrm{F}_{\mathrm{E}'} = \vec{R}_{\mathrm{E}'/\mathrm{E}}\mathrm{F}_{\mathrm{E}}$, $\mathrm{F}_{\mathrm{E}''} = \vec{R}_{\mathrm{E}''/\mathrm{E}'}\mathrm{F}_{\mathrm{E}'}$, and $\vec{R}_{\hat{n}}(\kappa)$ is the Rodrigues rotation about the eigenaxis $\hat{n}$ through the eigenangle κ according to the right-hand rule. At each time instant, let "a" denote the air particle located at a point that is fixed relative to the aircraft and upstream of the pitot tube. The location of the aircraft center of mass c relative to O_{E} at each time instant is given by

$$\vec{r}_{c/\mathrm{O}_{\mathrm{E}}} = \vec{r}_{c/a} + \vec{r}_{a/\mathrm{O}_{\mathrm{E}}} \tag{3}$$

Differentiating Eq. (3) with respect to F_{E} yields

$$\vec{V}_c = \vec{V}_{\mathrm{AC}} + \vec{V}_a \tag{4}$$

where

$$\vec{V}_c \stackrel{\Delta}{=} \vec{v}_{c/\mathrm{O}_{\mathrm{E}}/\mathrm{E}} = \overset{\mathrm{E}\bullet}{\vec{r}}_{c/\mathrm{O}_{\mathrm{E}}}, \qquad \vec{V}_{\mathrm{AC}} \stackrel{\Delta}{=} \vec{v}_{c/a/\mathrm{E}} = \overset{\mathrm{E}\bullet}{\vec{r}}_{c/a}, \qquad \vec{V}_a \stackrel{\Delta}{=} \vec{v}_{a/\mathrm{O}_{\mathrm{E}}/\mathrm{E}} = \overset{\mathrm{E}\bullet}{\vec{r}}_{\mathrm{a}/\mathrm{O}_{\mathrm{E}}} \tag{5}$$

and E• denotes the time derivative with respect to F_{E}. The angular velocity $\vec{\omega}_{\mathrm{AC/E}}$ of F_{AC} relative to F_{E} is obtained from the rotation matrix $\vec{R}_{\mathrm{AC/E}}$ by Poisson's equation

$$\overset{\mathrm{AC}\bullet}{\vec{R}}_{\mathrm{AC/E}} = \vec{R}_{\mathrm{AC/E}}\vec{\omega}^{\times}_{\mathrm{AC/E}} \tag{6}$$

where the operator "×" creates a skew-symmetric physical matrix. We resolve $\vec{V}_{\mathrm{AC}}$ and $\vec{\omega}_{\mathrm{AC/E}}$ in F_{AC} using the notation

$$\begin{bmatrix} U \\ V \\ W \end{bmatrix} \stackrel{\Delta}{=} \vec{V}_{\mathrm{AC}}|_{\mathrm{AC}}, \qquad \begin{bmatrix} P \\ Q \\ R \end{bmatrix} \stackrel{\Delta}{=} \vec{\omega}_{\mathrm{AC/E}}|_{\mathrm{AC}} \tag{7}$$

A. Airspeed V_{AC}

The airspeed V_{AC} is the magnitude of $\vec{V}_{\mathrm{AC}}$ and thus is the speed of the aircraft relative to the local atmosphere, that is,

$$V_{\mathrm{AC}} = |\vec{V}_{\mathrm{AC}}| = \sqrt{U^2 + V^2 + W^2} \tag{8}$$

B. Flight-Path Angle γ

The flight-path angle γ is the angle between $\vec{V}_c$ and its projection onto the horizontal plane spanned by $\hat{i}_{\mathrm{E}}$ and $\hat{j}_{\mathrm{E}}$.

C. Turn Rate τ

We resolve $\vec{\omega}_{\mathrm{AC/E}}$ in F_{E} as

$$\begin{bmatrix} P_{\mathrm{E}} \\ Q_{\mathrm{E}} \\ R_{\mathrm{E}} \end{bmatrix} \stackrel{\Delta}{=} \vec{\omega}_{\mathrm{AC/E}}|_{\mathrm{E}} \tag{9}$$

The turn rate of the aircraft is defined by $\tau \stackrel{\Delta}{=} R_{\mathrm{E}}$.

D. Altitude h

We resolve $\vec{r}_{c/\mathrm{O}_{\mathrm{E}}}$ in F_{E} as

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \stackrel{\Delta}{=} \vec{r}_{c/\mathrm{O}_{\mathrm{E}}}|_{\mathrm{E}} \tag{10}$$

The altitude h of the aircraft is given by $h = -Z$.

E. Angle of Attack α

The projected velocity vector $\vec{V}_{\mathrm{AC,proj}}$ is the projection of $\vec{V}_{\mathrm{AC}}$ onto the plane spanned by $\hat{i}_{\mathrm{AC}}$ and $\hat{k}_{\mathrm{AC}}$. Assuming that $\vec{V}_{\mathrm{AC,proj}}$ is nonzero, $\hat{i}_S$ is the unit vector aligned with $V_{\mathrm{AC,proj}}$, that is, $\hat{i}_S \stackrel{\Delta}{=} \hat{V}_{\mathrm{AC,proj}}$. The angle of attack $\alpha \in (-\pi, \pi]$ is the signed angle from $\hat{\imath}_S$ to $\hat{\imath}_{\mathrm{AC}}$ about $\hat{j}_{\mathrm{AC}}$. The stability frame F_S is then defined by rotating the aircraft frame F_{AC} by the angle $-\alpha$ about $\hat{j}_{\mathrm{AC}}$, that is, $F_S \stackrel{\Delta}{=} R_{\hat{j}_{\mathrm{AC}}}(-\alpha)F_{\mathrm{AC}}$. Using the components of V_{AC} resolved in F_{AC}, α is given by

$$\alpha = \mathrm{atan2}(W, U) \tag{11}$$

F. Sideslip Angle β

Assuming that $\vec{V}_{\mathrm{AC}}$ is nonzero, $\hat{i}_{\mathrm{W}}$ is the unit vector aligned with $\vec{V}_{\mathrm{AC}}$, that is, $\hat{i}_{\mathrm{W}} \triangleq \hat{V}_{\mathrm{AC}}$. The sideslip angle $\beta \in (-\pi, \pi]$ is the signed angle from $\hat{i}_{\mathrm{S}}$ to $\hat{i}_{\mathrm{W}}$ about $\hat{k}_{\mathrm{S}}$. The wind frame $\mathrm{F_W}$ is obtained by rotating the stability frame $\mathrm{F_S}$ by the angle β about $\hat{k}_{\mathrm{S}}$, that is, $\mathrm{F_W} \triangleq \vec{R}_{\hat{k}_{\mathrm{S}}}(\beta)\mathrm{F_S}$. Using the components of $\vec{V}_{\mathrm{AC}}$ resolved in $\mathrm{F_{AC}}$, β is given by

$$\beta = \mathrm{atan2}(V, \sqrt{U^2 + W^2}) \tag{12}$$

G. Ground Speed V_{g}

Using Eq. (10), the velocity vector $\vec{V}_c$ is resolved in F_E as

$$\vec{V}_c|_{\mathrm{E}} = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} \tag{13}$$

The ground velocity $\vec{V}_{\mathrm{g}}$ is the projection of $\vec{V}_c$ onto the horizontal plane spanned by $\hat{i}_{\mathrm{E}}$ and $\hat{j}_{\mathrm{E}}$, that is,

$$\vec{V}_{\mathrm{g}}|_{\mathrm{NE}} = \begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix} \tag{14}$$

where the north–east frame $\mathrm{F_{NE}}$ is defined with the direction $\hat{i}_{\mathrm{E}}$ representing north and the direction $\hat{j}_{\mathrm{E}}$ representing east. The ground speed V_{g} is the magnitude of $\vec{V}_{\mathrm{g}}$, that is,

$$V_{\mathrm{g}} = |\vec{V}_{\mathrm{g}}| = \sqrt{\dot{X}^2 + \dot{Y}^2} \tag{15}$$

Note that $\dot{Z} = -\dot{h}$.

H. Ground Position R_{g}

The ground position $\vec{r}_{\mathrm{g}}$ of the aircraft is the projection of $\vec{r}_{c/\mathrm{O_E}}$ onto the horizontal plane spanned by $\mathrm{F_{NE}}$, that is,

$$\vec{r}_{\mathrm{g}}(t) = X(t)\hat{i}_{\mathrm{E}} + Y(t)\hat{j}_{\mathrm{E}} \tag{16}$$

We define

$$R_{\mathrm{g}}(t) \triangleq \vec{r}_{\mathrm{g}}(t)|_{\mathrm{NE}} = \begin{bmatrix} X(t) \\ Y(t) \end{bmatrix} \tag{17}$$

Likewise, the desired ground position $\vec{r}_{\mathrm{g,des}}$ of the aircraft is given by

$$R_{\mathrm{g,des}}(t) \triangleq \vec{r}_{\mathrm{g,des}}(t)|_{\mathrm{NE}} = \begin{bmatrix} X_{\mathrm{des}}(t) \\ Y_{\mathrm{des}}(t) \end{bmatrix} \tag{18}$$

Using Eqs. (17) and (18), at each time t we define

$$\vec{a}(t) \triangleq \vec{r}_{\mathrm{g,des}}(t + T_{\mathrm{g}}) - \vec{r}_{\mathrm{g,des}}(t), \qquad \vec{b}(t) \triangleq \vec{r}_{\mathrm{g,des}}(t) - \vec{r}_{\mathrm{g}}(t), \qquad \vec{c} \triangleq \frac{\vec{a} \cdot \vec{b}}{|\vec{a}|^2}\vec{a}, \qquad \vec{d} \triangleq \vec{b} - \vec{c} \tag{19}$$

where T_{g} is the ground track interval, and $\vec{c}$ and $\vec{d}$ are parallel and perpendicular to $\vec{a}$, respectively, as shown in Fig. 1. Using Eqs. (14) and (19), we define

$$e_{\parallel} \triangleq \mathrm{sgn}[\vec{c} \cdot \vec{a}]|\vec{c}| \tag{20}$$

$$e_{\perp} \triangleq \mathrm{sgn}[(\vec{d} \times \vec{a}) \cdot \hat{k}_{\mathrm{E}}]|\vec{d}| \tag{21}$$

$$\eta \triangleq \mathrm{sgn}[(\vec{V}_{\mathrm{g}} \times \vec{a}) \cdot \hat{k}_{\mathrm{E}}]\sin^{-1}\frac{|\vec{V}_{\mathrm{g}} \times \vec{a}|}{|\vec{V}_{\mathrm{g}}||\vec{a}|} \tag{22}$$

where $e_{\parallel}$ and $e_{\perp}$ are signed distance errors relative to the desired path update $\vec{a}$, and the angle η is the velocity alignment error.

I. Wind Shear Model

Figure 2a shows the wind vector in F_{NE}. To model wind shear, we assume that the wind velocity direction is east and the wind speed varies with altitude. Figure 2b shows the wind model parameters, where the wind speed is zero from the ground up to 7920 ft, above which the wind speed increases linearly with a gradient of 0.05 kt/ft, becoming constant above 8320 ft.

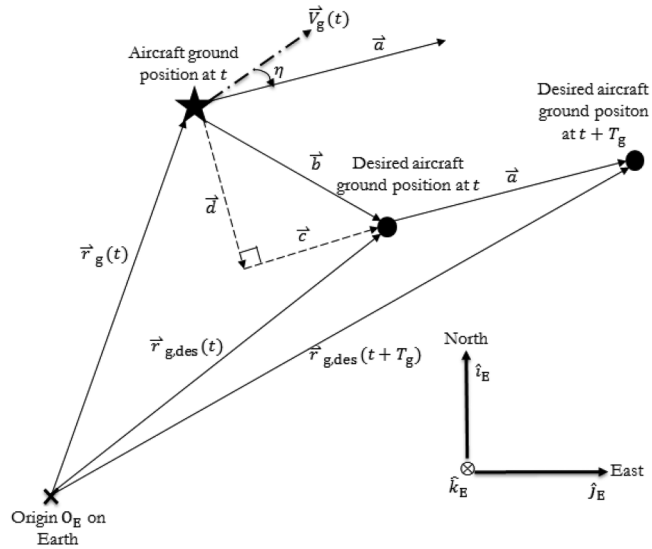

Fig. 1 Ground position R_g. Note that $|\vec{c}|$ and $|\vec{d}|$ represent distance errors relative to the desired path update $\vec{a}$, and the angle η is the velocity alignment error.

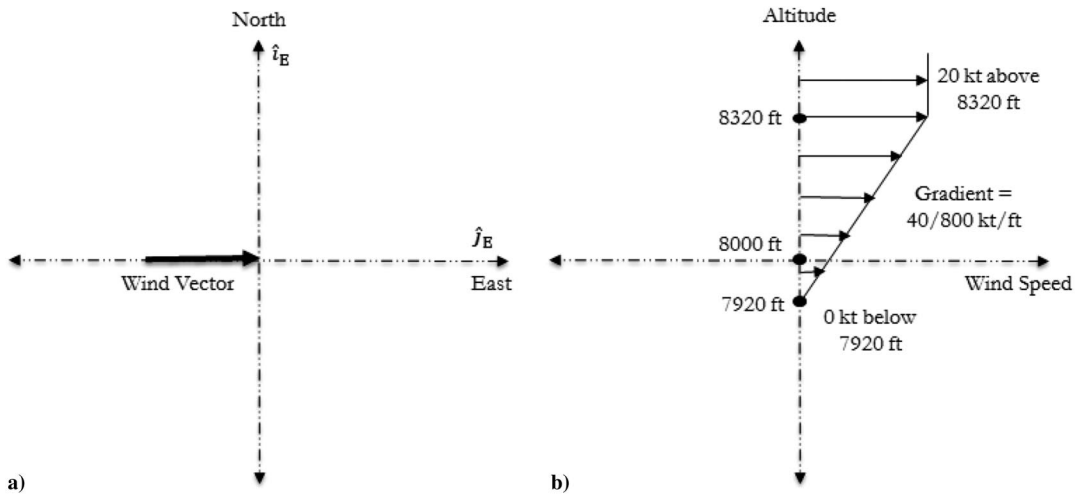

Fig. 2 Wind shear model: a) wind vector in F_{NE}; b) variation of wind speed with altitude. Wind speed varies linearly with a slope of 0.05 kt/ft between 7920 and 8320 ft.

J. Steady and Trim Flight

Steady flight is defined as aircraft flight that satisfies the conditions

$$\overset{AC\bullet}{\vec{V}}_{AC} = 0, \qquad \overset{AC\bullet}{\vec{\omega}}_{AC/E} = 0 \tag{23}$$

These conditions state that the velocity vector $\vec{V}_{AC}$ and the angular velocity vector $\vec{\omega}_{AC/E}$ are constant with respect to the aircraft frame F_{AC}, that is, U, V, W, P, Q, and R are constant. In the case $V_a = 0$, the following steady flight regimes are defined, although not all of these regimes are feasible for many aircraft.

1) In hovering flight, the aircraft maintains a constant location relative to the Earth. In this case, $\vec{V}_{AC} = 0$.

2) In straight-line flight, the aircraft flies along a straight line with its wings level. Straight-line flight may be either ascending ($\gamma > 0$), horizontal ($\gamma = 0$), or descending ($\gamma < 0$). In all of these cases,

$$\overset{E\bullet}{\vec{V}}_{AC} = 0$$

$\vec{V}_{AC}$ is nonzero, and $\vec{\omega}_{AC/E} = 0$.

3) In bullet flight, the aircraft flies along a straight line while rotating about its velocity vector. In this case, $\vec{V}_{AC}$ and $\vec{\omega}_{AC/E}$ are nonzero and parallel.

4) In circular flight, the aircraft flies along a circle. The circle may be either horizontal, vertical, or tilted relative to the Earth, whereas the aircraft attitude along the circle may be either banked or level. In all of these cases, $\vec{V}_{AC}$ and $\vec{\omega}_{AC/E}$ are nonzero and mutually orthogonal.

5) In helical flight, the aircraft flies along a helix. In this case, $\vec{V}_{AC}$ and $\vec{\omega}_{AC/E}$ are nonzero and neither parallel nor mutually orthogonal. A special case of helical flight is a barrel roll, where the wheels of the aircraft can be viewed as rolling along the inside of a cylinder.

In all steady flight regimes, V_{AC}, γ, τ, α, and β are constant. However, h, Ψ, Θ, and Φ may vary with time.

Steady flight in which

$$\overset{\mathrm{AC}\bullet}{\vec{g}} = 0$$

that is, the gravity vector is constant with respect to F_{AC}, is called *trim flight*. Equivalently, Φ and Θ are constant. Since

$$\overset{\mathrm{AC}\bullet}{\vec{V}}_{\mathrm{AC}} = 0$$

an equivalent condition is that Φ and γ are constant. Therefore, steady flight is trim flight if and only if it is either 1) hovering flight, 2) straight-line flight, 3) circular flight in a horizontal plane, or 4) helical flight around a vertical axis. In trim flight, the total force vector on the aircraft is constant with respect to F_{AC} (zero in hovering and straight-line flight) and the total moment vector on the aircraft is zero. These force and moment conditions imply that the thrust is constant and the control surfaces are set to constant trim angles.

III. Defining Control Architectures

In this section, we define six control architectures. Each control architecture is decentralized with respect to adaptation, where, for each component of the performance variable z in Eq. (B3), a separate RCAC block has a scalar output u given by Eq. (B4). Table 1 defines the control architectures, where each row defines a control architecture and each column represents a performance variable. For each control architecture, each row entry specifies the actuator that is used to follow commands given for the performance variable at the top of the corresponding column. Note that, by following specific commands, each architecture meets certain controlled flight objectives. For example, A_1 can hold an altitude by specifying zero flight-path-angle command but cannot specify the altitude, whereas A_2 can specify an altitude but cannot specify the flight-path angle. These scenarios may be useful under emergency conditions.

The aircraft control inputs are $\boldsymbol{T}$, $\boldsymbol{e}$, $\boldsymbol{a}$, and $\boldsymbol{r}$, which denote throttle, elevator, ailerons, and rudder, respectively. Note that each architecture has either two, three, or four control channels. For each architecture, we use the index $i = 1, 2, 3, 4$ to denote the channels for $\boldsymbol{T}$, $\boldsymbol{e}$, $\boldsymbol{a}$, $\boldsymbol{r}$, respectively. Note that one channel is missing in A_4, and that two channels are missing in A_6. The feedback vector for each channel is defined separately for each architecture. In later sections, each example is done by choosing one control architecture; each architecture is associated with a performance objective as listed in Table 1. For a chosen control architecture, the command may involve transitions among multiple trim conditions. However, none of the examples involve switching between different architectures.

Taking control architecture A_1 as an example, we define the measurement increments in each channel as

$$\delta V_{\mathrm{AC}}(k) \triangleq V_{\mathrm{AC}}(k) - V_{\mathrm{AC,trim}} \tag{24}$$

$$\delta\gamma(k) \triangleq \gamma(k) - \gamma_{\mathrm{trim}} \tag{25}$$

$$\delta\tau(k) \triangleq \tau(k) - \tau_{\mathrm{trim}} \tag{26}$$

$$\delta\beta(k) \triangleq \beta(k) - \beta_{\mathrm{trim}} \tag{27}$$

where the subscript "trim" refers to the initial trim flight, and k denotes the current time step. We choose the sample time to be 0.1 s for all control architectures. The chosen sample rate is approximately 7.86 times the frequency (in Hertz) of the short period mode of the GTM trimmed in horizontal flight at airspeed 100.59 kt and altitude 8000 ft. The performance variable z is given by the error signals

$$z(k) \triangleq \begin{bmatrix} V_{\mathrm{AC}}(k) - V_{\mathrm{AC,cmd}}(k) \\ \gamma(k) - \gamma_{\mathrm{cmd}}(k) \\ \tau(k) - \tau_{\mathrm{cmd}}(k) \\ \beta(k) - \beta_{\mathrm{cmd}}(k) \end{bmatrix} = \begin{bmatrix} \delta V_{\mathrm{AC}}(k) - \delta V_{\mathrm{AC,cmd}}(k) \\ \delta\gamma(k) - \delta\gamma_{\mathrm{cmd}}(k) \\ \delta\tau(k) - \delta\tau_{\mathrm{cmd}}(k) \\ \delta\beta(k) - \delta\beta_{\mathrm{cmd}}(k) \end{bmatrix} \tag{28}$$

where $V_{\mathrm{AC,cmd}}$, γ_{cmd}, τ_{cmd}, and β_{cmd} are the commands and

Table 1 Definition of control architectures

	V_{AC}	γ	τ	h	α	β	$e_{\parallel}$	$e_{\perp}$	η	Example
A_1	$\boldsymbol{T}$	$\boldsymbol{e}$	$\boldsymbol{a}$	—	—	$\boldsymbol{r}$	—	—	—	A.1.1, A.1.2, A.1.3, A.1.4, A.2.1, A.2.2, A.2.3, A.2.4, A.3.1, A.3.2, A.3.3,
				—	—		—	—	—	G.1.1, G.3.1
A_2	$\boldsymbol{T}$	—	$\boldsymbol{a}$	$\boldsymbol{e}$	—	$\boldsymbol{r}$	—	—	—	B.1.1, B.1.2, B.1.3, B.1.4, B.2.1, B.2.2, F.1.1, F.2.1, F.3.1, G.1.2, G.2.1
		—			—		—	—	—	
A_3	$\boldsymbol{T}$	$\boldsymbol{e}$	$\boldsymbol{a}$	—	$\boldsymbol{e}$	$\boldsymbol{r}$	—	—	—	C.1.1, C.1.2
				—			—	—	—	
A_4	$\boldsymbol{T}$	$\boldsymbol{e}$	$\boldsymbol{a}$	—	—	—	—	—	—	C.2.1, C.2.2
				—	—	—	—	—	—	
A_5	— —	—	—	$\boldsymbol{e}$	—	—	$\boldsymbol{T}$	$\boldsymbol{a}$	$\boldsymbol{r}$	D.1.1, D.1.2, G.1.3
		—	—		—	—				
A_6	— —	—	$\boldsymbol{T}$	$\boldsymbol{e}$	—	—	—	—	—	E.1.1, E.1.2
		—			—	—	—	—	—	

For each control architecture, the throttle $\boldsymbol{T}$, elevator e, ailerons a, and rudder r are assigned to specific measurements that define the performance variable in the associated feedback channel. The feedback signals for each channel are defined separately for each architecture.

$$\delta V_{\mathrm{AC,cmd}} \triangleq V_{\mathrm{AC,cmd}} - V_{\mathrm{AC,trim}} \tag{29}$$

$$\delta \gamma_{\mathrm{cmd}} \triangleq \gamma_{\mathrm{AC,cmd}} - \gamma_{\mathrm{AC,trim}} \tag{30}$$

$$\delta \tau_{\mathrm{cmd}} \triangleq \tau_{\mathrm{AC,cmd}} - \tau_{\mathrm{AC,trim}} \tag{31}$$

$$\delta \beta_{\mathrm{cmd}} \triangleq \beta_{\mathrm{AC,cmd}} - \beta_{\mathrm{AC,trim}} \tag{32}$$

are the incremental commands. Note that all increments are defined relative to the initial trim.

Let the requested actuator settings be denoted by

$$\boldsymbol{T}_{\mathrm{req}}(k) \triangleq \boldsymbol{T}_{\mathrm{trim}} + \delta \boldsymbol{T}_{\mathrm{req}}(k) \tag{33}$$

$$\boldsymbol{e}_{\mathrm{req}}(k) \triangleq \boldsymbol{e}_{\mathrm{trim}} + \delta \boldsymbol{e}_{\mathrm{req}}(k) \tag{34}$$

$$\boldsymbol{a}_{\mathrm{req}}(k) \triangleq \boldsymbol{a}_{\mathrm{trim}} + \delta \boldsymbol{a}_{\mathrm{req}}(k) \tag{35}$$

$$\boldsymbol{r}_{\mathrm{req}}(k) \triangleq \boldsymbol{r}_{\mathrm{trim}} + \delta \boldsymbol{r}_{\mathrm{req}}(k) \tag{36}$$

where the first term on the right-hand side denotes the constant actuator setting for the initial trim and the second term on the right-hand side denotes the requested actuator setting increment specified by RCAC. The actual actuator settings are determined by the actuator dynamics, which are modeled as first-order systems with saturation nonlinearities modeled by stroke and rate limits [23,29]; the parameters of the actuator models are given in Table 2.

Thus, the requested actuator settings may not be equal to the actual actuator settings, which are denoted by

$$\boldsymbol{T}_{\mathrm{actual}}(k) \triangleq \boldsymbol{T}_{\mathrm{trim}} + \delta \boldsymbol{T}_{\mathrm{actual}}(k) \tag{37}$$

$$\boldsymbol{e}_{\mathrm{actual}}(k) \triangleq \boldsymbol{e}_{\mathrm{trim}} + \delta \boldsymbol{e}_{\mathrm{actual}}(k) \tag{38}$$

$$\boldsymbol{a}_{\mathrm{actual}}(k) \triangleq \boldsymbol{a}_{\mathrm{trim}} + \delta \boldsymbol{a}_{\mathrm{actual}}(k) \tag{39}$$

$$\boldsymbol{r}_{\mathrm{actual}}(k) \triangleq \boldsymbol{r}_{\mathrm{trim}} + \delta \boldsymbol{r}_{\mathrm{actual}}(k) \tag{40}$$

where the left-hand side denotes the actual actuator setting and the second term on the right-hand side denotes the actual actuator setting increment. Except for the examples in case G concerning unknown actuator failure, we assume that the actual actuator settings are known to RCAC.

For each channel, RCAC updates a strictly proper dynamic controller represented in input–output form as

$$\delta u_{i,\mathrm{req}}(k) = \phi_i(k)^{\mathrm{T}} \theta_i(k) \tag{41}$$

where $\delta u_{i,\mathrm{req}}$ denotes the requested actuator setting increment for the ith channel, where, for example, the corresponding actuator in the ith channel of control architecture A_1 is $\delta \boldsymbol{T}_{\mathrm{req}}$, $\delta \boldsymbol{e}_{\mathrm{req}}$, $\delta \boldsymbol{a}_{\mathrm{req}}$, and $\delta \boldsymbol{r}_{\mathrm{req}}$ for $i = 1, 2, 3, 4$, respectively. For channel i in control architecture A_1, the components of the feedback vector ϕ_i include values of the actual actuator setting increment $u_{i,\mathrm{actual}}$, which is $\delta \boldsymbol{T}_{\mathrm{actual}}$, $\delta \boldsymbol{e}_{\mathrm{actual}}$, $\delta \boldsymbol{a}_{\mathrm{actual}}$, and $\delta \boldsymbol{r}_{\mathrm{actual}}$ for $i = 1, 2, 3, 4$, respectively, as well as the additional signals

Table 2 Parameters of actuator models for throttle, elevator, ailerons, and rudder

	Stroke limits	Rate limits	Bandwidth
Throttle $\boldsymbol{T}$	[0,100]%	∞	20π Hz
Elevator $\boldsymbol{e}$	[−20, 30] deg	[−300, 300] deg/s	20π Hz
Ailerons $\boldsymbol{a}$	[−20, 20] deg	[−300, 300] deg/s	20π Hz
Rudder $\boldsymbol{r}$	[−30, 30] deg	[−300, 300] deg/s	20π Hz

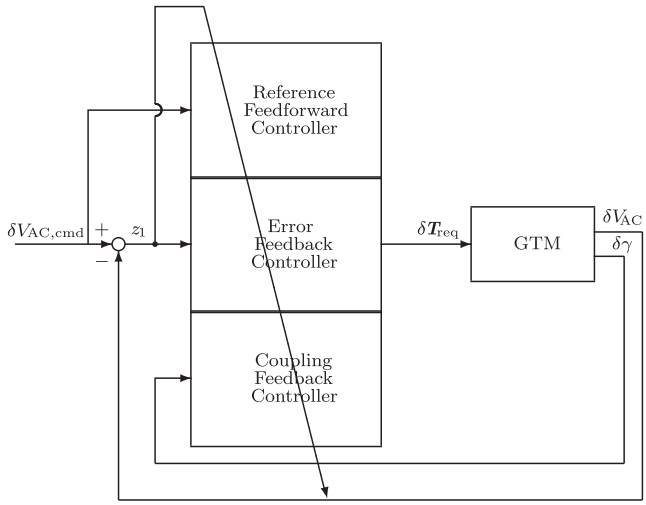

Fig. 3 Block diagram for channel 1 in A_1, which uses thrust to follow the airspeed command. The error signal z_1 for the control is the difference $\delta V_{\mathrm{AC,cmd}} - \delta V_{\mathrm{AC}}$. This channel also uses the flight-path angle for feedback.

$$
\begin{aligned}
\phi_1(k) &\triangleq [\delta \boldsymbol{T}_{\mathrm{actual}}(k-1) \ \cdots \ \delta \boldsymbol{T}_{\mathrm{actual}}(k-n_c) \quad \delta V_{\mathrm{AC,cmd}}(k-1) \ \cdots \ \delta V_{\mathrm{AC,cmd}}(k-n_c) \\
&\qquad z_1(k-1) \ \cdots \ z_1(k-n_c) \qquad \delta\gamma(k-1) \ \cdots \ \delta\gamma(k-n_c)]^{\mathrm{T}}, \\
\phi_2(k) &\triangleq [\delta \boldsymbol{e}_{\mathrm{actual}}(k-1) \ \cdots \ \delta \boldsymbol{e}_{\mathrm{actual}}(k-n_c) \qquad \delta\gamma_{\mathrm{cmd}}(k-1) \ \cdots \ \delta\gamma_{\mathrm{cmd}}(k-n_c) \\
&\qquad z_2(k-1) \ \cdots \ z_2(k-n_c) \qquad \delta V_{\mathrm{AC}}(k-1) \ \cdots \ \delta V_{\mathrm{AC}}(k-n_c)]^{\mathrm{T}}, \\
\phi_3(k) &\triangleq [\delta \boldsymbol{a}_{\mathrm{actual}}(k-1) \ \cdots \ \delta \boldsymbol{a}_{\mathrm{actual}}(k-n_c) \qquad \delta\tau_{\mathrm{cmd}}(k-1) \ \cdots \ \delta\tau_{\mathrm{cmd}}(k-n_c) \\
&\qquad z_3(k-1) \ \cdots \ z_3(k-n_c) \qquad \delta\beta(k-1) \ \cdots \ \delta\beta(k-n_c)]^{\mathrm{T}}, \\
\phi_4(k) &\triangleq [\delta \boldsymbol{r}_{\mathrm{actual}}(k-1) \ \cdots \ \delta \boldsymbol{r}_{\mathrm{actual}}(k-n_c) \qquad \delta\beta_{\mathrm{cmd}}(k-1) \ \cdots \ \delta\beta_{\mathrm{cmd}}(k-n_c), \\
&\qquad z_4(k-1) \ \cdots \ z_4(k-n_c) \qquad \delta\tau(k-1) \ \cdots \ \delta\tau(k-n_c)]^{\mathrm{T}}
\end{aligned}
\tag{42}
$$

where $z = [\,z_1 \quad z_2 \quad z_3 \quad z_4\,]^{\mathrm{T}}$ and z_1, z_2, z_3, z_4 are the components of z in Eq. (28), which is the vector of command-following errors. The vector θ_i of controller coefficients has the same size as ϕ_i. Figure 3 shows the block diagram for channel 1, which uses thrust to follow the airspeed command. In doing so, A_1 uses the airspeed command as a feedforward signal, airspeed command-following error as a feedback signal, and flight-path angle as coupling feedback signals. The remaining channels of A_1 use similar structures to follow flight-path-angle, turn-rate, and sideslip-angle commands.

For architecture A_1, we use

$$G_{\mathrm{f}}(z) = \mathrm{diag}(1/z^4, -1/z^4, -1/z^4, 1/z^4)$$

and the tuning parameters

$$n_c = 8, R_z = I_4, R_u = \mathrm{diag}(0.5, 1, 10, 10), R_\theta = \mathrm{diag}(10^{-5} I_{4n_c}, 10^{-4} I_{4n_c}, 10^{-3} I_{4n_c}, 10^{-3} I_{4n_c})$$

The same notation is used for the remaining control architectures. The feedback vector ϕ_i, filter G_{f}, and tuning parameters n_c, R_z, R_u, and R_θ for control architectures A_2, A_3, A_4, A_5, and A_6 are given in Appendix A. Except for the evaluation of warmup strategies in Scenario A.1, the tuning parameters are fixed for each control architecture.

IV. Comparison of Trimgtm and RCAC for Determining Trim

One of the most fundamental tasks in aircraft flight control is to determine states and actuator settings that define trim. This problem continues to be of interest [9,30–34]. Given initial states and actuator settings, the GTM applies the function trimgtm to the aircraft equations of motion and aerodynamic lookup table to determine the closest trim states and actuator settings. We use trimgtm to determine initial trim states and actuator settings for all simulations. We also test the ability of RCAC to determine trim states and actuator settings. To do this, we initialize the states and actuators of the GTM with values that are not necessarily trim states and actuator settings. We then use a subset of these initial aircraft states as commands and use RCAC to control the GTM. The goal is to determine whether RCAC can reach a nearby trim state. We then compare the attained trim states and actuator settings with the values obtained by trimgtm for the same initial states and actuator settings. Note that using RCAC to determine trim states and actuator settings can be viewed as using RCAC to fly the aircraft from an initial nontrim flight condition to a trim. In so doing, RCAC uses no knowledge of the aircraft equations of motion or the aerodynamic lookup table.

For illustration, we initialize the GTM with the states and actuator settings

$$
\begin{aligned}
&U(0) = 126.4 \text{ ft/s}, (0) = 0 \text{ ft/s}, (0) = 6.6 \text{ ft/s}, (0) = 0 \text{ deg/s}, (0) = 0 \text{ deg/s}, \\
&R(0) = 0 \text{ deg/s}, (0) = 8000 \text{ ft}, \Phi(0) = 0 \text{ deg}, \Theta(0) = 4 \text{ deg}, \Psi(0) = 45 \text{ deg}, \\
&\boldsymbol{T}(0) = 20\%, \boldsymbol{e}(0) = 0 \text{ deg}, \boldsymbol{a}(0) = 0 \text{ deg}, \boldsymbol{r}(0) = 0 \text{ deg}
\end{aligned}
\tag{43}
$$

The initial airspeed, flight-path angle, turn rate, sideslip angle, and angle of attack corresponding to Eq. (43) are

$$V_{\mathrm{AC}}(0) = 75 \text{ kt}, \gamma(0) = 1 \text{ deg}, \tau(0) = 0 \text{ deg/s}, \beta(0) = 0 \text{ deg}, \alpha(0) = 3 \text{ deg} \tag{44}$$

which represents straight-line flight.

We first use trimgtm to determine nearby trim. From the initial flight Eqs. (43) and (44), trimgtm yields the trim states and actuator settings

$$V_{\mathrm{AC,trim}} = 75.04 \text{ kt}, \quad \gamma_{\mathrm{trim}} = 0.96 \text{ deg}, \quad \tau_{\mathrm{trim}} = 0.485 \text{ deg/s}, \quad \beta_{\mathrm{trim}} = -0.04 \text{ deg}, \quad \alpha_{\mathrm{trim}} = 5.15 \text{ deg},$$
$$\boldsymbol{T}_{\mathrm{trim}} = 21.91\%, \boldsymbol{e}_{\mathrm{trim}} = 1.12 \text{ deg}, \quad \boldsymbol{a}_{\mathrm{trim}} = 0.33 \text{ deg}, \quad \boldsymbol{r}_{\mathrm{trim}} = -0.06 \text{ deg} \tag{45}$$

Note that Eq. (45) is a helical trim flight.

Next we use RCAC with the commands (44) to fly the aircraft from the initial flight Eqs. (43) and (44) to a nearby trim. To do so, we use control architecture 1. Figure 4 shows that RCAC reaches trim flight with the trim states and actuator settings

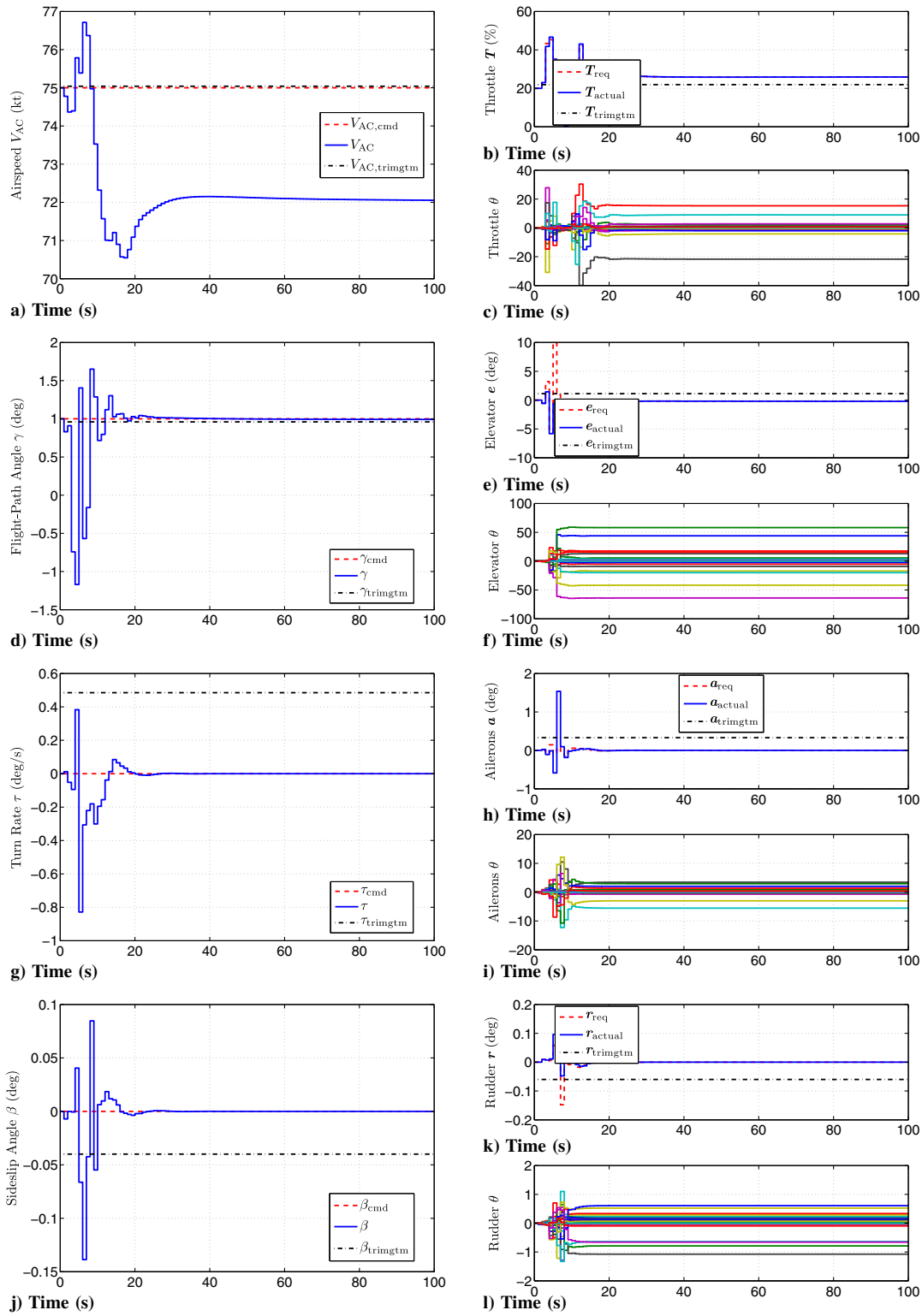

Fig. 4 Control architecture A_1 for attaining trim flight from the initial states and actuator settings (43). Considering the nontrim flight $V_{AC}(0) = 75$ kt, $\gamma(0) = 1$ deg, $\tau(0) = 0$ deg/s, and $\beta(0) = 0$ deg, RCAC gives a nearby ascending straight-line trim flight, whereas trimgtm gives a helical trim flight. Consequently, the trim computed by trimgtm and trim reached by RCAC are qualitatively different.

$$V_{\mathrm{AC,trim}} = 72.05 \text{ kt}, \quad \gamma_{\mathrm{trim}} = 0.99 \text{ deg}, \quad \tau_{\mathrm{trim}} = 0 \text{ deg/s}, \quad \beta_{\mathrm{trim}} = 0 \text{ deg},$$
$$\alpha_{\mathrm{trim}} = 6.85 \text{ deg}, \quad \boldsymbol{T}_{\mathrm{trim}} = 25.90\%, \boldsymbol{e}_{\mathrm{trim}} = -0.21 \text{ deg}, \quad \boldsymbol{a}_{\mathrm{trim}} = 0 \text{ deg}, \quad \boldsymbol{r}_{\mathrm{trim}} = 0 \text{ deg} \tag{46}$$

Note that Eq. (46) is a straight-line trim flight. Comparing Eqs. (45) and (46), note that the trim reached by RCAC is qualitatively close to the commands (44).

V. Case A: Control with Minimal Modeling

A. Scenario A.1: Adaptation Warmup Strategies

Unless stated otherwise, at the start of each scenario, the aircraft is assumed to be flying in an initial trim without the use of feedback control. The components of the controller-coefficient vector θ are thus initially set to zero, that is, $\theta(0) = 0$. Although $\theta(0)$ can be set arbitrarily, we choose $\theta(0) = 0$ to reflect the absence of additional modeling information. RCAC must therefore adapt the components of θ from their initial zero values to suitable nonzero values. We compare four techniques for doing this. First, we consider the case in which no special effort is made to assist in the transition from open- to closed-loop control. This case is called no warmup. To improve the transient performance, we then consider three approaches to controller warmup, namely, ambient warmup, impulsive warmup, and noise warmup. Unless stated otherwise, the GTM is initialized with the trim

$$V_{\mathrm{AC}}(0) = V_{\mathrm{AC,trim}} = 100.6 \text{ kt}, \quad \gamma(0) = \gamma_{\mathrm{trim}} = 0 \text{ deg}, \quad \tau(0) = \tau_{\mathrm{trim}} = 0 \text{ deg/s},$$
$$\beta(0) = \beta_{\mathrm{trim}} = 0 \text{ deg}, \quad \alpha(0) = \alpha_{\mathrm{trim}} = 3 \text{ deg}, (0) = 8000 \text{ ft} \tag{47}$$

for all of the examples in the paper. The initial actual actuator settings for all actuators are set to the actuator settings for the initial trim, that is,

$$\boldsymbol{T}_{\mathrm{actual}}(0) = \boldsymbol{T}_{\mathrm{trim}} = 22.8\%, \boldsymbol{e}_{\mathrm{actual}}(0) = \boldsymbol{e}_{\mathrm{trim}} = 2.7 \text{ deg}, \quad \boldsymbol{a}_{\mathrm{actual}}(0) = \boldsymbol{a}_{\mathrm{trim}} = 0 \text{ deg}, \boldsymbol{r}_{\mathrm{actual}}(0) = \boldsymbol{r}_{\mathrm{trim}} = 0 \text{ deg}$$

For no warmup and each controller-warmup technique, we use slightly different tuning parameters to compare the performance of each strategy under the most favorable conditions.

Example A.1.1: For no warmup, horizontal straight-line flight with trapezoidal airspeed command, the incremental commands are given by

$$\delta V_{\mathrm{AC,cmd}}(k) = \begin{cases} 0, & k < 700, \\ \min\{5, 0.005(k-700)\} \text{ kt}, & k \geq 700, \end{cases}$$
$$\delta\gamma_{\mathrm{cmd}}(k) = 0 \text{ deg}, \delta\tau_{\mathrm{cmd}}(k) = 0 \text{ deg/s}, \delta\beta_{\mathrm{cmd}}(k) = 0 \text{ deg} \tag{48}$$

where the trapezoidal airspeed command starts at $t = 70$ s. We use control architecture A_1 and the tuning parameters

$$n_c = 10, R_z = I_4, R_u = \mathrm{diag}(0.5, 1, 10, 10), R_\theta = \mathrm{diag}(10^{-3}I_{4n_c}, 10^{-2}I_{4n_c}, 10^{-2}I_{4n_c}, 10^{-3}I_{4n_c})$$

Figure 5 shows that no adaptation occurs until $t = 70$ s when the trapezoidal command starts. This is because the simulation starts from a trim with zero command-following error and no disturbance is introduced. Note that the maximum command-following errors for airspeed, flight-path angle, turn rate, and sideslip angle are 4.1 kt, 15.0 deg, 5.3 deg/s, and 0.6 deg, respectively. In the next three examples, the goal is to improve the transient response by using a warmup strategy. ■

Example A.1.2: For the same problem as in Example A.1.1, but for ambient noise warmup, horizontal straight-line flight with trapezoidal airspeed command, we introduce zero-mean white noise with standard deviation 0.001 kt into the wind speed for the entire simulation starting from $t = 0$ s. All tuning parameters are the same as in the case of no warmup (Example A.1.1) except for R_θ, which for this example is $R_\theta = \mathrm{diag}(10^{-3}I_{3n_c}, 10^{-4}I_{n_c}, 10^{-2}I_{4n_c}, 10^{-2}I_{4n_c}, 10^{-3}I_{4n_c})$.

Comparing with Example A.1.1, Fig. 6 shows that the transient error is smaller with ambient noise warmup. As expected, Figs. 6g and 6j show that the turn rate and sideslip angle are corrupted by the ambient noise. ■

Example A.1.3: For the same problem as in Example A.1.1, but for actuator impulse warmup, horizontal straight-line flight with trapezoidal airspeed command, we introduce impulses into T_{req} and e_{req} at $t = 10$ s for 0.1 s with amplitude 1% and 1 deg, respectively. All tuning parameters are the same as in the case of no warmup (Example A.1.1) except for n_c and R_θ, which for this example are $n_c = 12$ and $R_\theta = \mathrm{diag}(10^{-2}I_{4n_c}, 10^{-3}I_{4n_c}, 10^{-1}I_{4n_c}, 10^{-3}I_{4n_c})$.

Comparing with Example A.1.1, Fig. 7 shows that the transient error is smaller with actuator impulse warmup. ■

Example A.1.4: For the same problem as in Example A.1.1, but for actuator noise warmup, horizontal straight-line flight with trapezoidal airspeed command, we introduce zero-mean white noise with standard deviation 0.001% into T_{req} and zero-mean white noise with standard deviation 0.001 deg into e_{req}, a_{req}, and r_{req} from $t = 10$ to 70 s. All tuning parameters are the same as in the case of no warmup (Example A.1.1) except for n_c and R_θ, which for this example are $n_c = 8$ and $R_\theta = \mathrm{diag}(10^{-3}I_{3n_c}, 10^{-4}I_{n_c}, 10^{-3}I_{4n_c}, I_{4n_c}, 10^{-3}I_{4n_c})$.

Comparing no warmup, ambient noise warmup, actuator impulse warmup, and actuator noise warmup, Figs. 5–8 show that the peak transient errors averaged over all four channels are 6.25, 2.57, 3.05, and 2.22 for no warmup, ambient noise warmup, actuator impulse warmup, and actuator noise warmup, respectively. Note that the smallest value is obtained with actuator noise warmup. With actuator noise warmup, the maximum command-following errors for airspeed, flight-path angle, turn rate, and sideslip angle are 4.4 kt, 3.2 deg, 1.2 deg/s, and 0.1 deg, respectively. In addition, unlike the case of ambient noise warmup, the final command-following error using actuator noise warmup is not corrupted by ambient noise, which is more convenient for performance comparison. Thus, we use actuator noise warmup for all subsequent simulations unless mentioned otherwise. ■

B. Scenario A.2: Trim Command Following

Example A.2.1: For ascending straight-line flight with trapezoidal flight-path-angle command and constant airspeed command using control architecture A_1, the incremental commands are given by

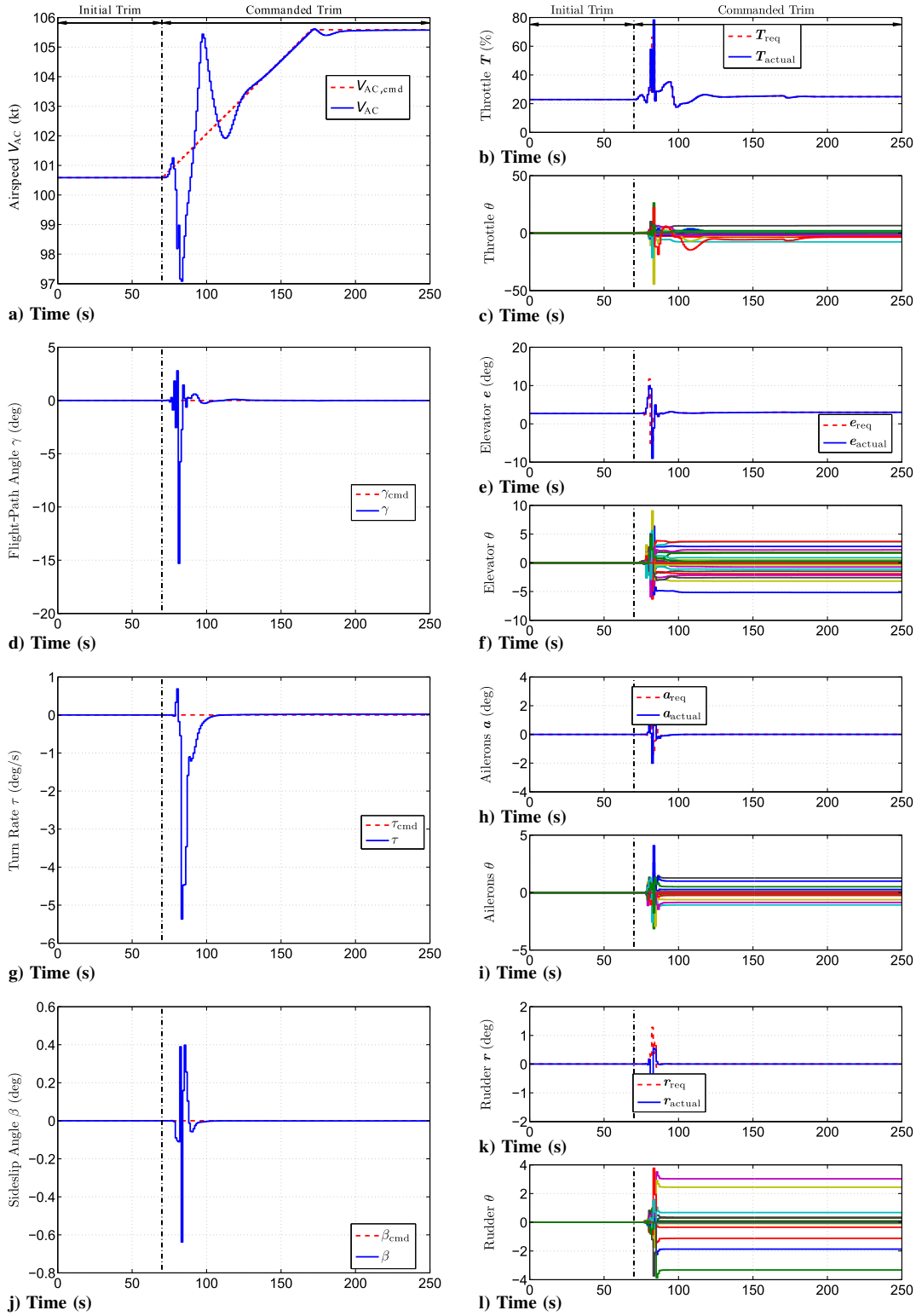

Fig. 5 Example A.1.1: Control architecture A_1 for horizontal straight-line flight with trapezoidal airspeed command without warmup.

$$\delta\gamma_{\text{cmd}}(k) = \begin{cases} 0, & k < 700, \\ \min\{5, 0.005(k-700)\}\ \text{deg}, & k \geq 700, \end{cases}$$
$$\delta V_{\text{AC,cmd}}(k) = 0\ \text{kt}, \delta\tau_{\text{cmd}}(k) = 0\ \text{deg/s}, \delta\beta_{\text{cmd}}(k) = 0\ \text{deg} \tag{49}$$

where the trapezoidal airspeed command starts at $t = 70$ s.

Figure 9d shows that the maximum flight-path-angle command-following error is less than 0.35 deg and that the aircraft maintains constant flight-path angle and airspeed in ascending straight-line flight. Note that the transfer-matrix realization of the controller (41) corresponding to the converged elevator θ at $t = 250$ s shown in Fig. 9f has a pole at one. This means that RCAC automatically develops an integrator for flight-path-angle command following. This is the case in several subsequent examples but is not mentioned explicitly. ■

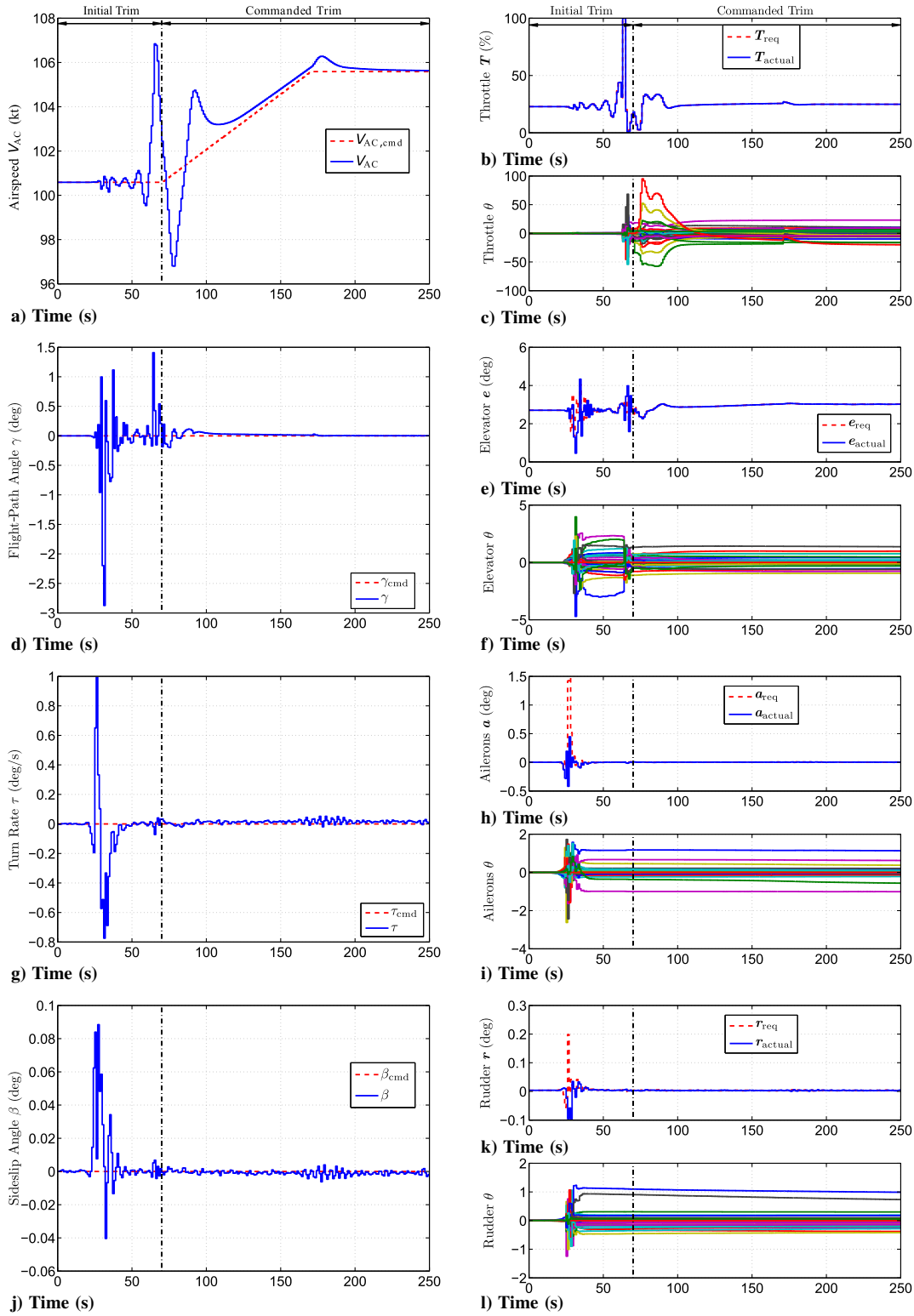

Fig. 6 Example A.1.2: Control architecture A_1 for horizontal straight-line flight with trapezoidal airspeed command and ambient noise warmup. The noise in the wind speed V_a starts at $t = 0$ s and persists for the entire simulation.

Example A.2.2: For horizontal circular flight with trapezoidal turn-rate command, constant airspeed command, and zero sideslip-angle command using control architecture A_1, the incremental commands are given by

$$\delta\tau_{\text{cmd}}(k) = \begin{cases} 0, & k < 700, \\ \min\{5, 0.005(k-700)\}\ \text{deg/s}, & k \geq 700, \end{cases}$$
$$\delta V_{\text{AC,cmd}}(k) = 0\ \text{kt}, \delta\gamma_{\text{cmd}}(k) = 0\ \text{deg}, \delta\beta_{\text{cmd}}(k) = 0\ \text{deg} \tag{50}$$

where the trapezoidal turn-rate command starts at $t = 70$ s.

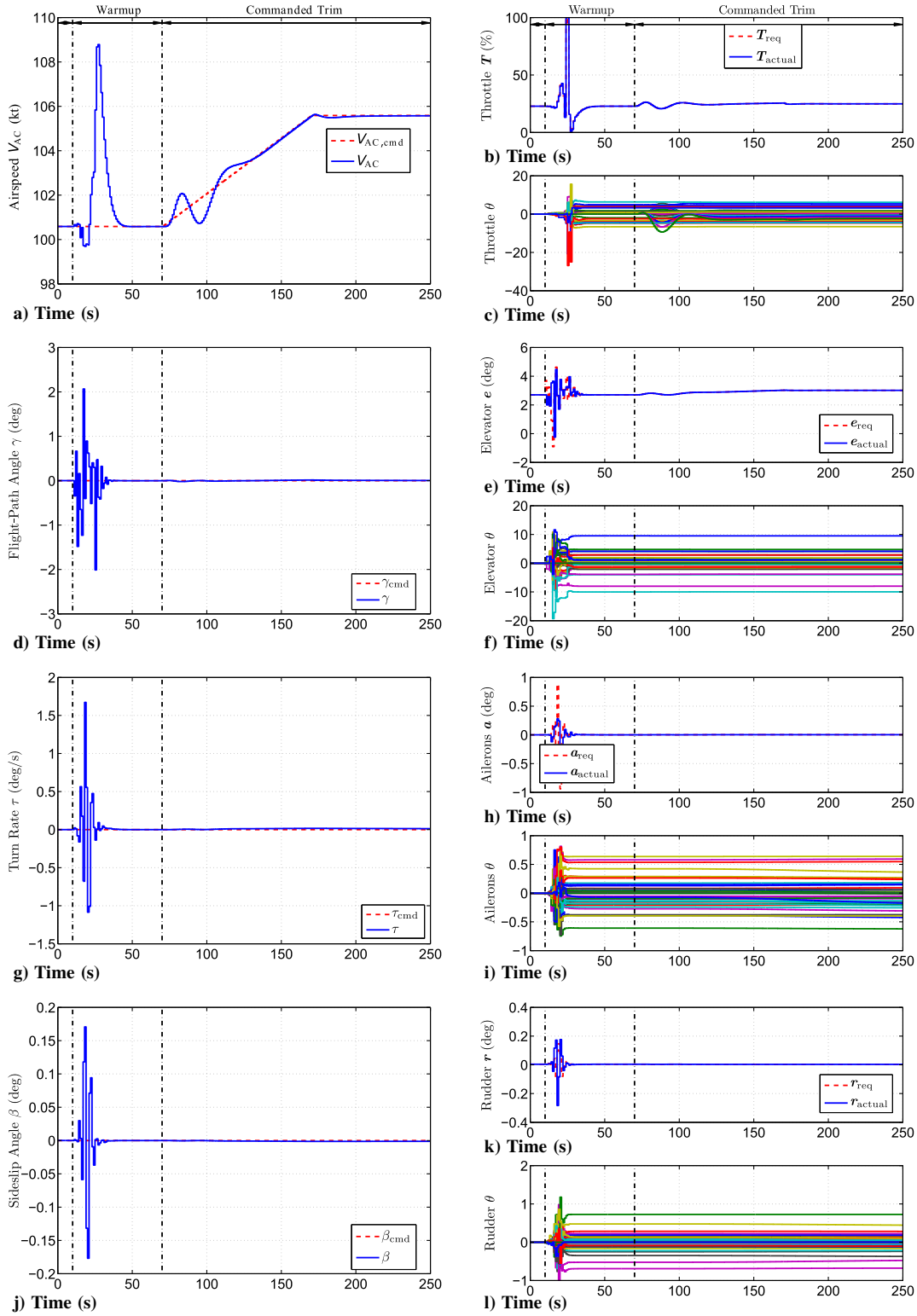

Fig. 7 Example A.1.3: Control architecture A_1 for horizontal straight-line flight with trapezoidal airspeed command and actuator impulse warmup. The impulses in T and e occur at $t = 10$ s.

Figure 10j shows that the maximum sideslip angle is less than 0.07 deg and the aircraft maintains constant turn rate and airspeed in horizontal flight. This shows the ability of RCAC to decouple the lateral channels. In addition, the use of ailerons to follow turn-rate commands implies banked turn, and the roll angle (not shown in Fig. 10) is 24.8 deg at $t = 250$ s. ■

Example A.2.3: For horizontal circular flight with trapezoidal turn-rate command, constant airspeed command, and zero sideslip-angle command, we compare the performance of the converged RCAC controller in Example A.2.2 with the full-state-feedback LQR controller under nominal flight conditions. In doing so, we initialize Eqs. (B16) and (B17) with θ and P of Example A.2.2 at the last time step, and thus no warmup strategy is used. The incremental commands are given by

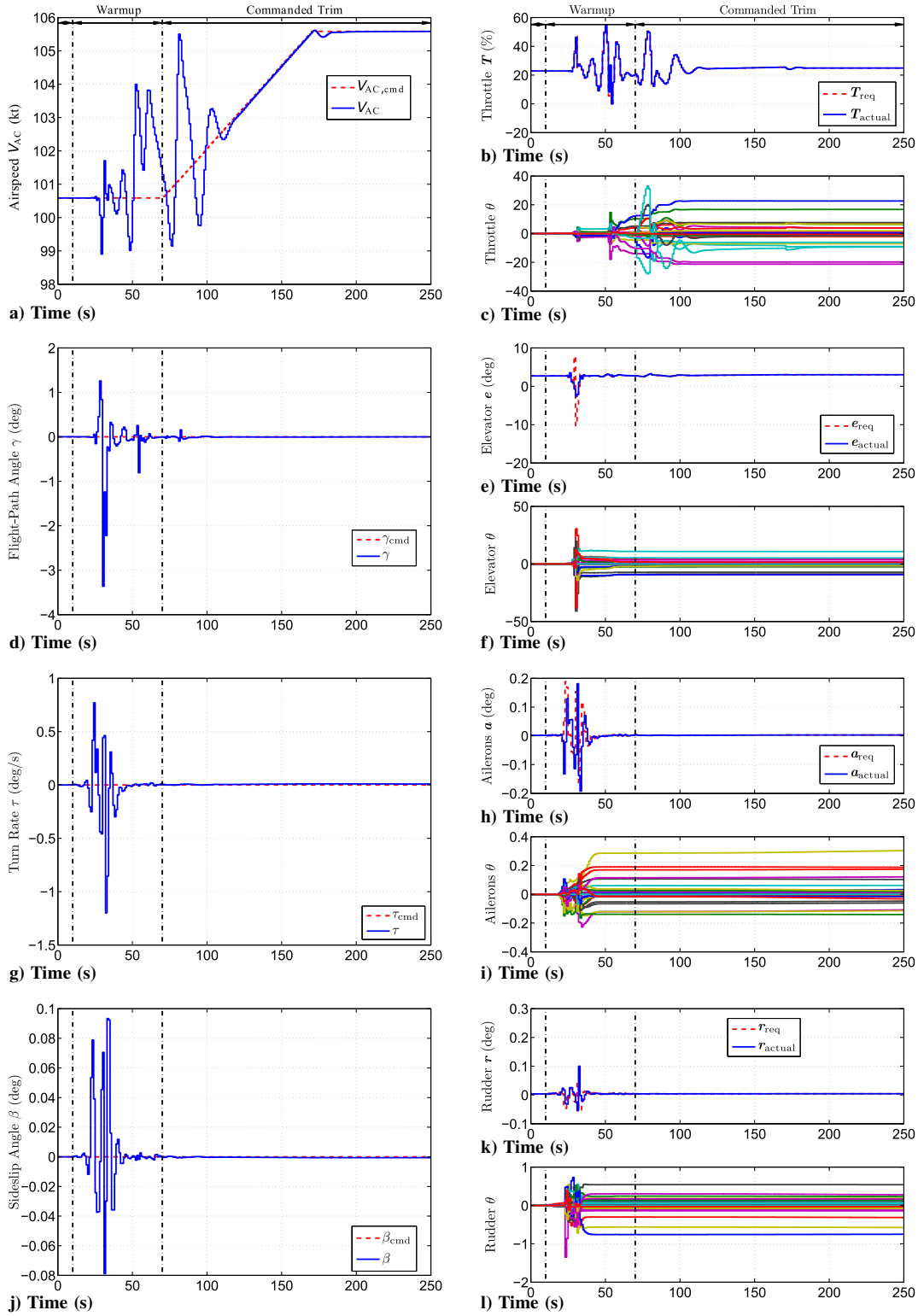

Fig. 8 Example A.1.4: Control architecture A_1 for horizontal straight-line flight with trapezoidal airspeed command and actuator noise warmup. The noise in T, e, a, and r begins at $t = 10$ s and ends at $t = 70$ s.

$$\delta\tau_{\text{cmd}}(k) = \begin{cases} 0, & k < 700, \\ \min\{5, 0.025(k-700)\}\ \text{deg/s}, & k \geq 700, \end{cases}$$
$$\delta V_{\text{AC,cmd}}(k) = 0\ \text{kt}, \delta\gamma_{\text{cmd}}(k) = 0\ \text{deg}, \delta\beta_{\text{cmd}}(k) = 0\ \text{deg} \tag{51}$$

where the trapezoidal turn-rate command starts at $t = 70$ s. Note that this example is similar to Example A.2.2, except that the incremental turn-rate command ramps up in 20 s rather than 100 s as in Example A.2.2.

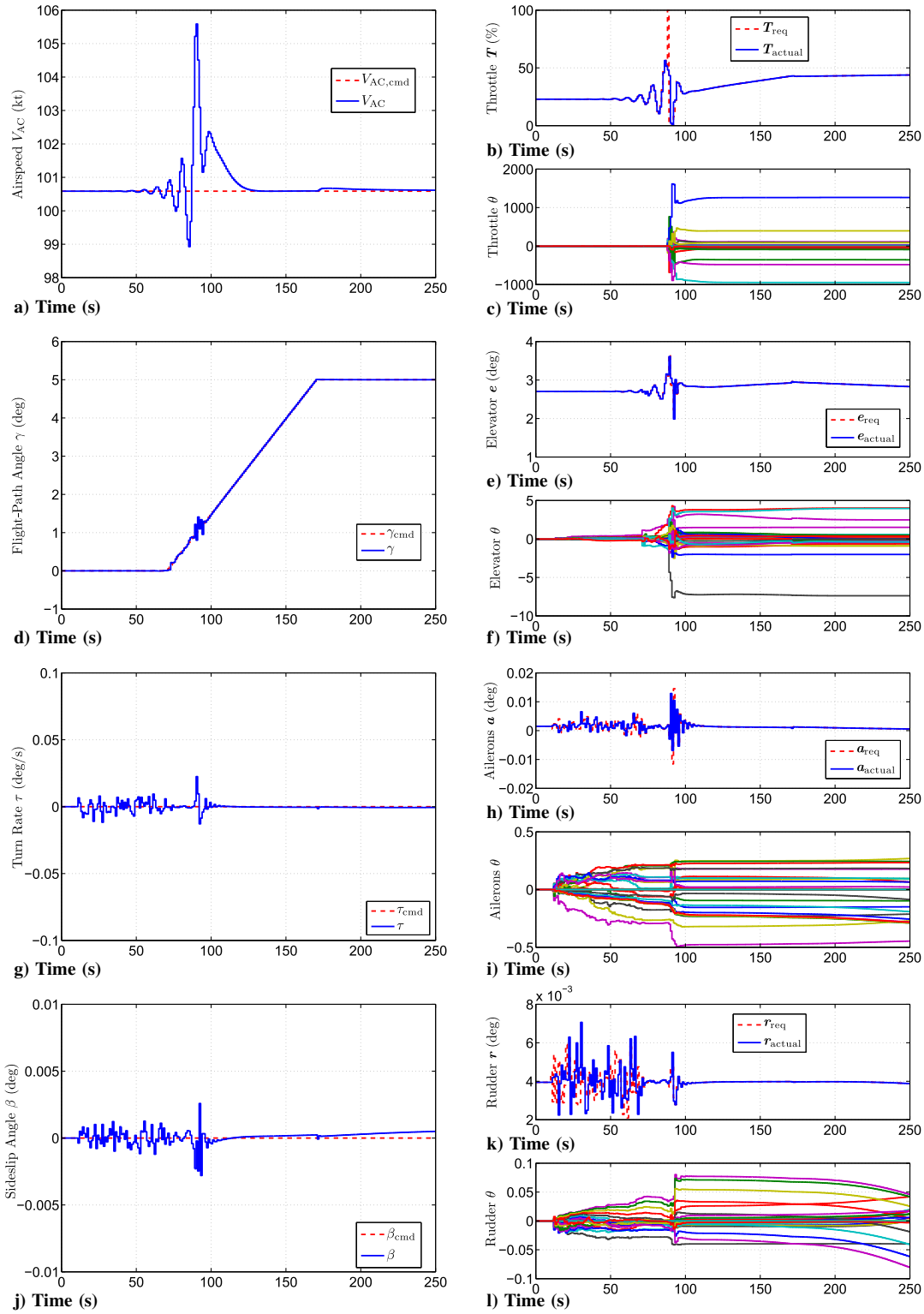

Fig. 9 Example A.2.1: Control architecture A_1 for ascending straight-line flight with trapezoidal flight-path-angle command and constant airspeed command.

To design the LQR controller, we linearize the GTM about the trim (47). The linearization is given by

$$\delta x(k+1) = A\delta x(k) + B\delta u_{\text{req}}(k) \tag{52}$$

$$\delta y(k) = C\delta x(k) \tag{53}$$

where

$$\delta x(k) = [\,\delta U(k) \quad \delta V(k) \quad \delta W(k) \quad \delta P(k) \quad \delta Q(k) \quad \delta R(k) \quad \delta h(k) \quad \delta\Phi(k) \quad \delta\Theta(k)\,]^{\mathrm{T}} \tag{54}$$

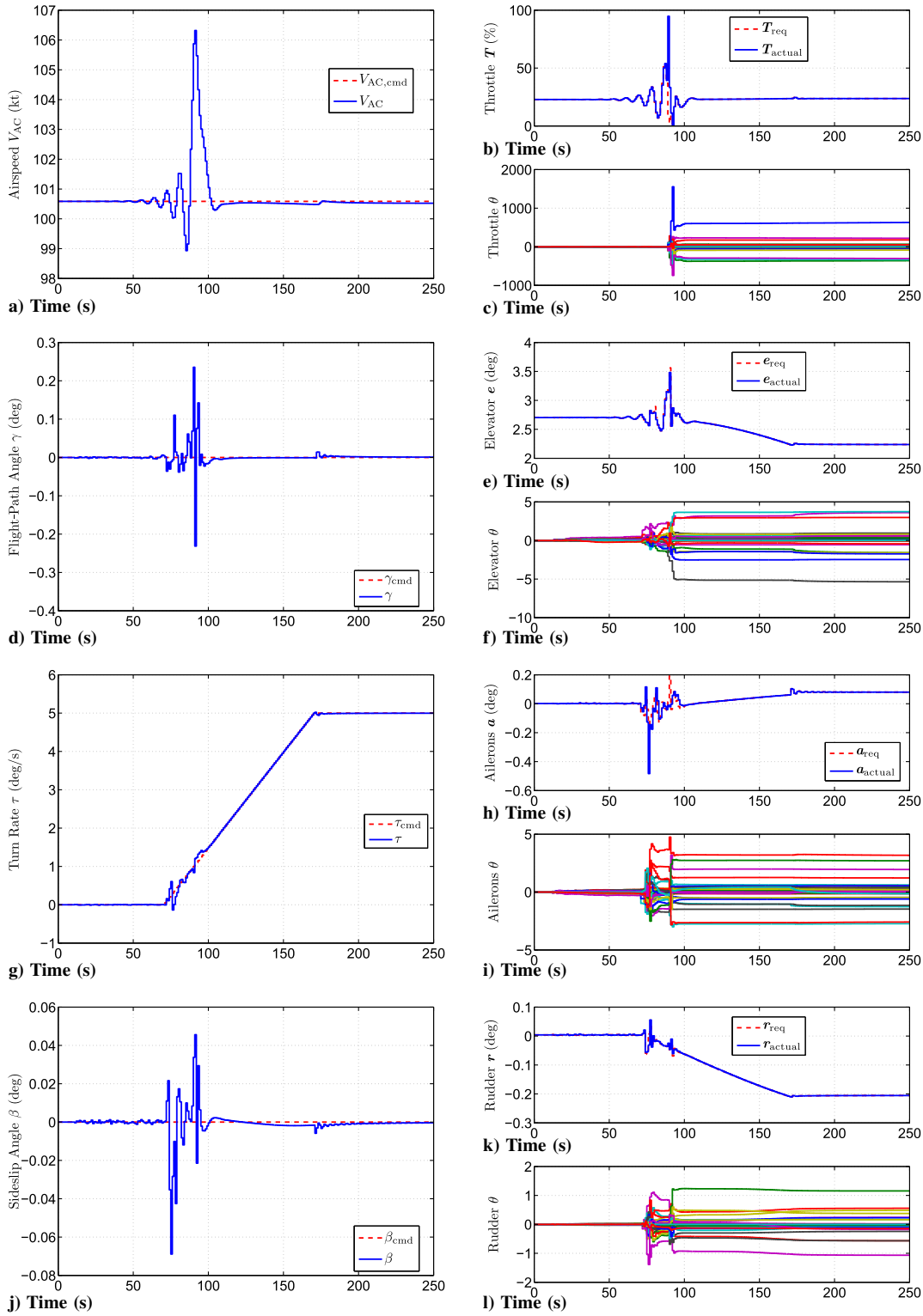

Fig. 10 Example A.2.2: Control architecture A_1 for horizontal circular flight with trapezoidal turn-rate command, constant airspeed command, and zero sideslip-angle command.

$$\delta u_{\mathrm{req}}(k) = [\,\delta T_{\mathrm{req}}(k) \quad \delta e_{\mathrm{req}}(k) \quad \delta a_{\mathrm{req}}(k) \quad \delta r_{\mathrm{req}}(k)\,]^{\mathrm{T}} \tag{55}$$

$$\delta y(k) = [\,\delta V_{\mathrm{AC}}(k) \quad \delta\gamma(k) \quad \delta\tau(k) \quad \delta\beta(k)\,]^{\mathrm{T}} \tag{56}$$

and the matrices A, B, and C are obtained using the GTM function linmodel. The LQR control law with integrator is given by

$$\delta u_{\mathrm{req}}(k) = -K_1\delta x(k) - K_2 y_{\mathrm{int}}(k) \tag{57}$$

where the integrated output $y_{\text{int}}(k)$ is given by

$$y_{\text{int}}(k) = -\frac{T_s}{z-1} z(k) \tag{58}$$

$T_s = 0.1$ s is the sampling time, $z(k)$ is given by Eq. (28), and the gain matrices K_1 and K_2 are obtained using the linearization (A, B, C) with state weighting $R_1 = I_{13\times 13}$ and control weighting $R_2 = \text{diag}(0.1, 0.1, 100, 100)$. Note that, with $z(k)$ given by Eq. (28), the LQR controller has the same objective as the control architecture A_1, namely, following airspeed, flight-path-angle, turn-rate, and sideslip-angle commands. To implement LQR Eq. (57), 13 measurements [nine states Eq. (54) and four outputs Eq. (56)] are needed, whereas, for implementing A_1, eight measurements (four inputs and four outputs) given by Eq. (42) are required.

Figure 11 shows that the maximum RCAC command-following errors for airspeed, flight-path angle, turn rate, and sideslip angle are 0.45 kt, 0.02 deg, 0.15 deg/s, and 0.02 deg, respectively, whereas the maximum LQR command-following errors for airspeed, flight-path angle, turn rate, and sideslip angle are 0.02 kt, 0.06 deg, 0.42 deg/s, and 0.15 deg, respectively. Note that, by using the converged controller coefficients from Fig. 10 as $\theta(0)$, the obtained transients are small relative to Fig. 10. Figure 11i shows that the aileron controller coefficients adapt to follow a larger slope in the turn-rate command compared with Fig. 10g. The RCAC command-following errors at $t = 250$ s for airspeed, flight-path angle, turn rate, and sideslip angle are 0.04 kt, 0.0003 deg, 0.001 deg/s, and 6×10^{-5} deg, respectively, whereas the LQR command-following errors at $t = 250$ s for airspeed, flight-path angle, turn rate, and sideslip angle are 0.0009 kt, 0.003 deg, 0.02 deg/s, and 0.005 deg, respectively. Comparing the maximum command-following errors and command-following errors at $t = 250$ s for RCAC and LQR controllers, RCAC has better performance in following flight-path-angle, turn-rate, and sideslip-angle commands, whereas LQR has better performance in following the airspeed command. The roll angle (not shown in Fig. 11) is 24.79 deg for RCAC and 24.62 deg for LQR at $t = 250$ s.■

Example A.2.4: For helical flight around a vertical axis with trapezoidal turn-rate command, trapezoidal flight-path-angle command, constant airspeed command, and zero sideslip-angle command using control architecture A_1, the incremental commands are given by

$$\begin{aligned} \delta\tau_{\text{cmd}}(k) &= \begin{cases} 0, & k < 700, \\ \min\{5, 0.005(k-700)\}\ \text{deg/s}, & k \geq 700, \end{cases} \\ \delta\gamma_{\text{cmd}}(k) &= \begin{cases} 0, & k < 700, \\ \min\{4, 0.004(k-700)\}\ \text{deg}, & k \geq 700, \end{cases} \\ \delta V_{\text{AC,cmd}}(k) &= 0\ \text{kt}, \delta\beta_{\text{cmd}}(k) = 0\ \text{deg} \end{aligned} \tag{59}$$

where the trapezoidal flight-path-angle and turn-rate commands start at $t = 70$ s.

Figure 12j shows that the maximum sideslip angle is less than 0.06 deg and the aircraft maintains constant airspeed during helical flight. This shows the ability of RCAC to decouple the lateral channels. In addition, the use of ailerons to control turn rate implies banked helical flight, and the roll and pitch angles (not shown in Fig. 12) are 24.9 and 7.2 deg, respectively, at $t = 250$ s. ■

C. Scenario A.3: Robustness of RCAC Tuning to the Initial Trim

We now investigate the robustness of RCAC to variations in the initial trim and trim command using control architecture A_1. Because the force and moment coefficients of the GTM are functions of α, β, P, Q, and R, changing $\alpha(0) = \alpha_{\text{trim}}$ results in different initial vehicle dynamics. For all of the examples in this scenario, the incremental commands are given by

$$\begin{aligned} \delta\tau_{\text{cmd}}(k) &= \begin{cases} 0, & k < 700, \\ \min\{5, 0.005(k-700)\}\ \text{deg/s}, & k \geq 700, \end{cases} \\ \delta V_{\text{AC,cmd}}(k) &= 0\ \text{kt}, \delta\gamma_{\text{cmd}}(k) = 0\ \text{deg}, \delta\beta_{\text{cmd}}(k) = 0\ \text{deg} \end{aligned} \tag{60}$$

where the trapezoidal turn-rate command starts at $t = 70$ s.

Example A.3.1: For the robustness of RCAC tuning, given an increase of 16 kt in $V_{\text{AC}}(0)$ relative to the trim given by Eq. (47), the GTM is initialized with the trim $\alpha(0) = \alpha_{\text{trim}} = 2$ deg, $V_{\text{AC}}(0) = V_{\text{AC,trim}} = 116.6$ kt, $\gamma(0) = \gamma_{\text{trim}} = 0$ deg, $\tau(0) = \tau_{\text{trim}} = 0$ deg/s, and $\beta(0) = \beta_{\text{trim}} = 0$ deg with $h(0) = 8000$ ft. In this example, the control surfaces have more authority than in Example A.2.2, where $V_{\text{AC,trim}} = 100.6$ kt due to the increased dynamic pressure.

Figure 13 shows that the maximum command-following errors for airspeed, flight-path angle, turn rate, and sideslip angle are 3.2 kt, 4 deg, 2.1 deg/s, and 0.5 deg, respectively, and the aircraft maintains constant turn rate and constant airspeed in horizontal flight after 170 s. This shows the robustness of RCAC to variations in the initial trim and trim command. ■

Example A.3.2: For robustness of RCAC tuning, given a decrease of 18.6 kt in $V_{\text{AC}}(0)$ relative to the trim given by Eq. (47), the GTM is initialized with the trim $\alpha(0) = \alpha_{\text{trim}} = 5$ deg, $V_{\text{AC}}(0) = V_{\text{AC,trim}} = 82$ kt, $\gamma(0) = \gamma_{\text{trim}} = 0$ deg, $\tau(0) = \tau_{\text{trim}} = 0$ deg/s, and $\beta(0) = \beta_{\text{trim}} = 0$ deg with $h(0) = 8000$ ft. In this example, the control surfaces have less authority than in Example A.2.2, where $V_{\text{AC,trim}} = 100.6$ kt due to the decreased dynamic pressure.

Figure 14 shows that the maximum RCAC command-following errors for airspeed, flight-path angle, turn rate, and sideslip angle are 4.4 kt, 1 deg, 0.2 deg/s, and 0.02 deg, respectively, and the aircraft maintains constant turn rate and constant airspeed in horizontal flight after 170 s. This shows the robustness of RCAC to variations in the initial trim and trim command. For this example, as shown in Fig. 14, LQR given by Eq. (57) is not able to follow the incremental commands and diverges around $t = 800$ s. ■

Example A.3.3: Robustness of RCAC tuning, given a decrease of 24.6 kt in $V_{\text{AC}}(0)$ relative to the trim given by Eq. (47), the GTM is initialized with the trim $\alpha(0) = \alpha_{\text{trim}} = 6$ deg, $V_{\text{AC}}(0) = V_{\text{AC,trim}} = 76$ kt, $\gamma(0) = \gamma_{\text{trim}} = 0$ deg, $\tau(0) = \tau_{\text{trim}} = 0$ deg/s, and $\beta(0) = \beta_{\text{trim}} = 0$ deg with $h(0) = 8000$ ft. In this example, the control surfaces have less authority than in Example A.3.2, where $V_{\text{AC,trim}} = 82$ kt due to the decreased dynamic pressure.

Figure 15 shows that the maximum command-following errors for airspeed, flight-path angle, turn rate, and sideslip angle are 3.3 kt, 1.9 deg, 11.7 deg/s, and 6.2 deg, respectively, and the aircraft maintains constant turn rate and airspeed in horizontal flight after 200 s. Note that the decrease of 24.6 kt in $V_{\text{AC}}(0)$ causes an increase in the transient response; however, after 190 s, the command-following errors become small.■

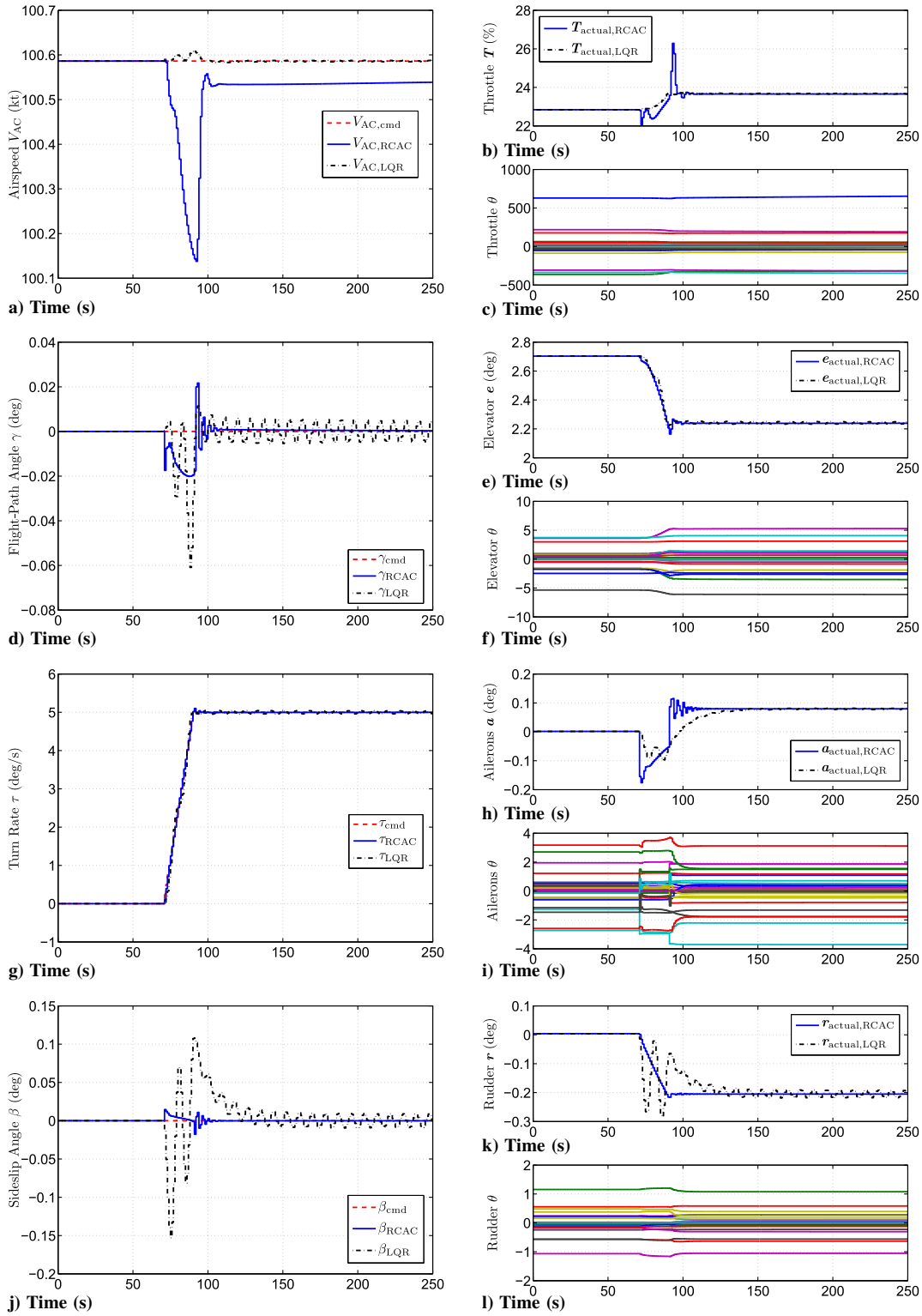

Fig. 11 Example A.2.3: Control architecture A_1 for horizontal circular flight with trapezoidal turn-rate command, constant airspeed command, and zero sideslip-angle command. The performance of the converged RCAC controller in Example A.2.2 is compared with the full-state-feedback LQR controller under nominal flight conditions.

VI. Case B: Control with Unknown Time-Varying Aircraft Dynamics

A. Scenario B.1: Changing Mass, Moments of Inertia, and C.G. Location During Flight

We now investigate scenarios where the dynamics of the aircraft change in an unknown way during flight. In particular, we consider changes in the aircraft mass, moments of inertia, and c.g. location along $\hat{i}_{AC}$ during flight. For all of the examples in this scenario, we use control architecture A_2 with the following incremental commands

$$\delta V_{AC,cmd}(k) = 0 \text{ kt}, \delta h_{cmd}(k) = 0 \text{ ft}, \delta\tau_{cmd}(k) = 0 \text{ deg/s}, \delta\beta_{cmd}(k) = 0 \text{ deg} \tag{61}$$

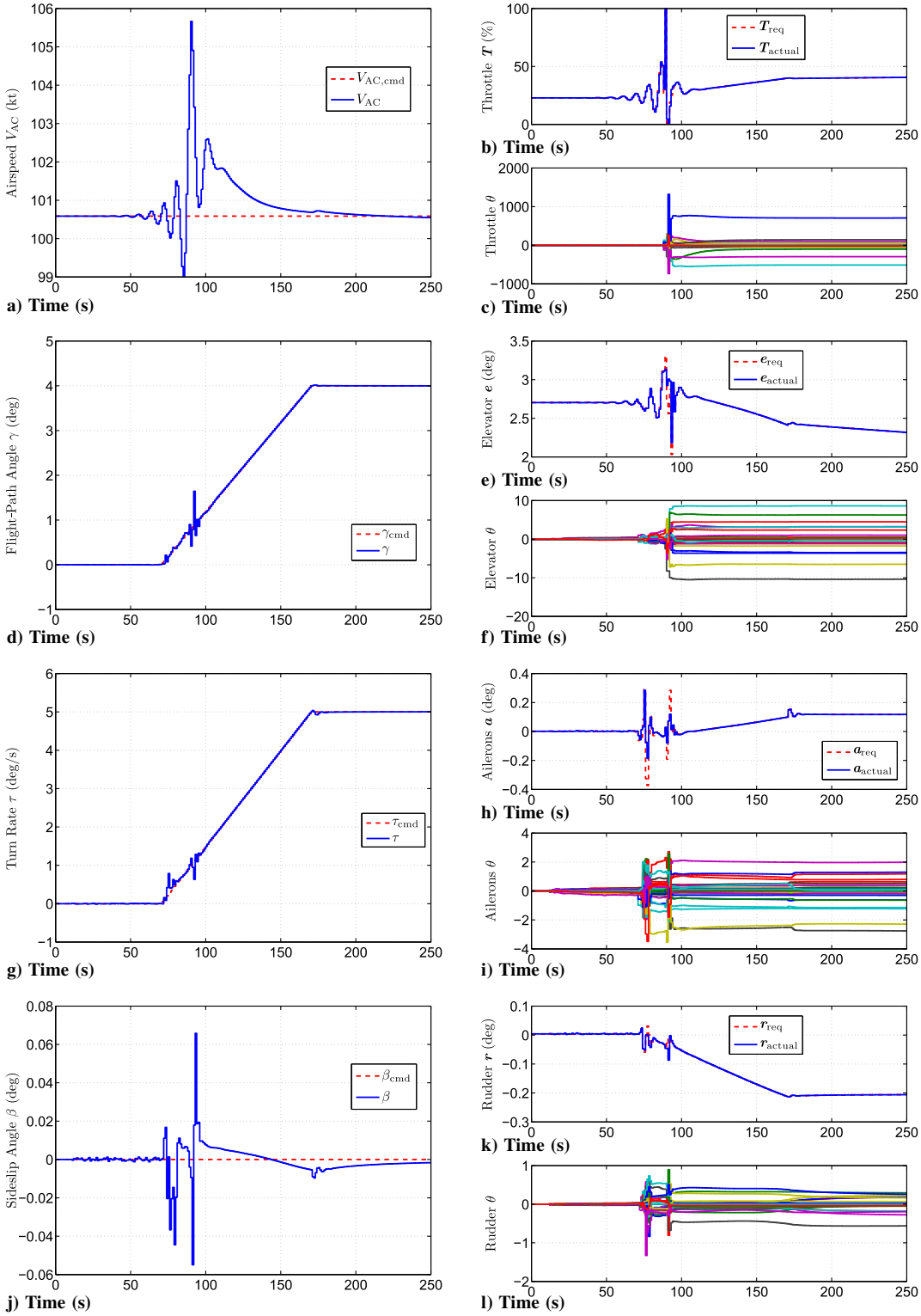

Fig. 12 Example A.2.4: Control architecture A_1 for helical flight around a vertical axis with trapezoidal turn-rate command, trapezoidal flight-path angle, constant airspeed command, and zero sideslip-angle command.

We let the aircraft fly in a straight line with constant altitude and airspeed, while the aircraft dynamics change during flight. For all of the examples in this scenario, the center of pressure and c.g. are initially collocated.

Example B.1.1: For time-varying dynamics with c.g. moving forward, we first fly the aircraft with fixed c.g. location for a baseline comparison, and then we linearly vary the c.g. location along $\hat{i}_{AC}$ with a rate of 0.0008 ft/s during flight, so that, at 250 s, the c.g. has moved forward by 0.2 ft, and does not change after 250 s.

Figure 16 shows that, with fixed c.g. location, the maximum command-following errors for airspeed, altitude, turn rate, and sideslip angle are 4.12 kt, 8.01 ft, 0.07 deg/s, and 0.01 deg, respectively. Note that the transients in Figs. 16a, 16d, 16g, and 16j are due to the actuator noise warmup, from $t = 10$ to 70 s, to adapt to suitable controller coefficients θ. Figures 16c, 16f, 16i, and 16l show that θ converges after the actuator

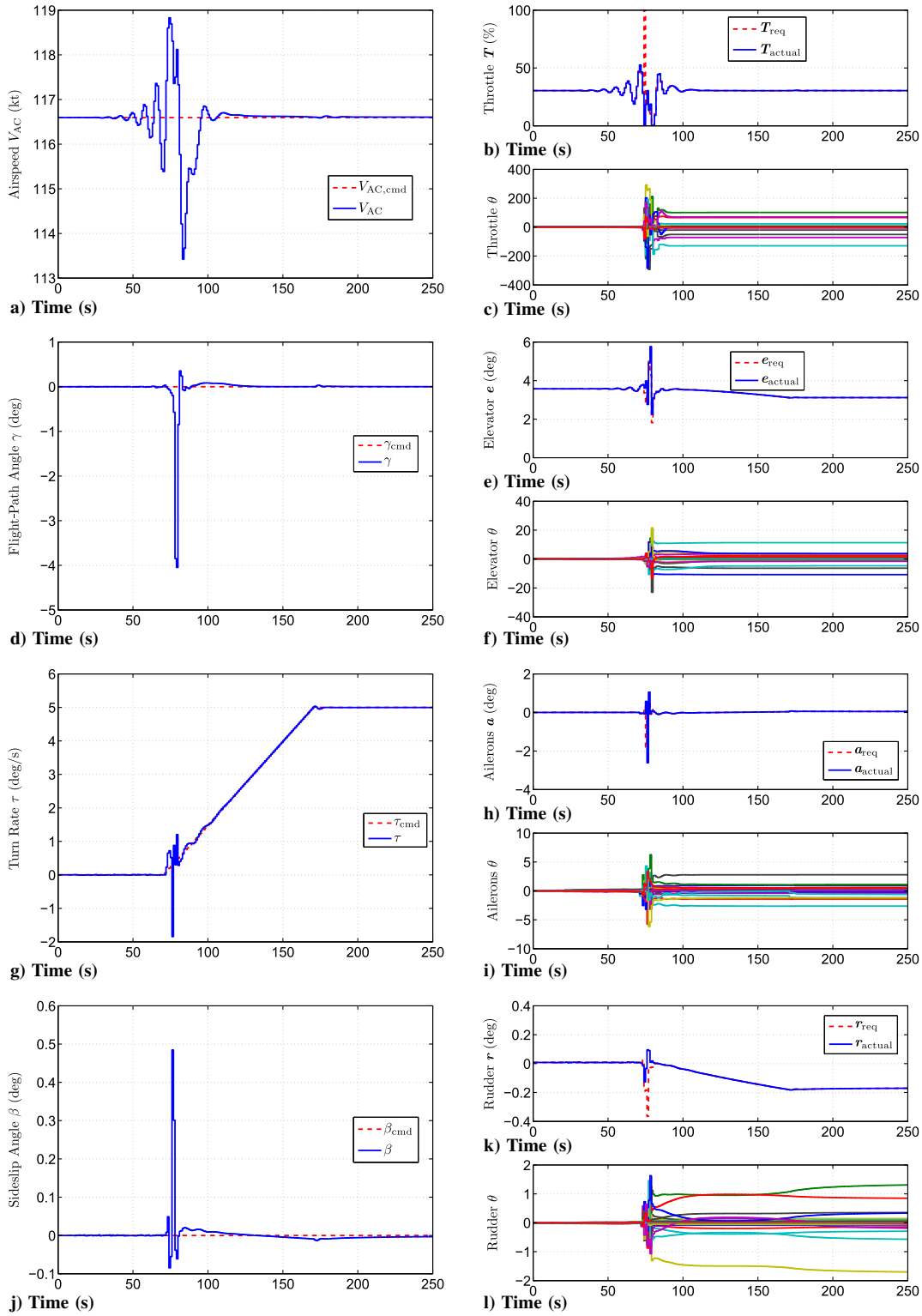

Fig. 13 Example A.3.1: Control architecture A_1 for horizontal circular flight with a modified initial trim. $V_{AC}(0)$ is increased by 16 kt relative to the trim (47).

noise warmup and subsequently brings the aircraft back to its initial trim with zero command-following errors for airspeed, altitude, turn rate, and sideslip angle.

Figure 17 shows that, with moving c.g. location, the maximum command-following errors for airspeed, altitude, turn rate, and sideslip angle are 4.9 kt, 32.5 ft, 0.05 deg/s, and 0.02 deg, respectively, and the aircraft maintains constant altitude and airspeed in straight-line flight. Note that the requested actuator settings in all four channels become constant after 250 s. ■

Example B.1.2: For time-varying dynamics with c.g. moving aft, we linearly vary the c.g. location along $-\hat{i}_{AC}$ with a rate of 0.0008 ft/s, so that, at 250 s, the c.g. has moved aft by 0.2 ft and does not change after 250 s.

Figure 18 shows that the maximum command-following errors for airspeed, altitude, turn rate, and sideslip angle are 4.2 kt, 28.1 ft, 0.52 deg/s, and 0.17 deg, respectively, and the aircraft maintains constant airspeed and altitude with 24.5 ft offset in straight-line flight. Note that the requested actuator settings in all four channels become constant after 250 s. ■

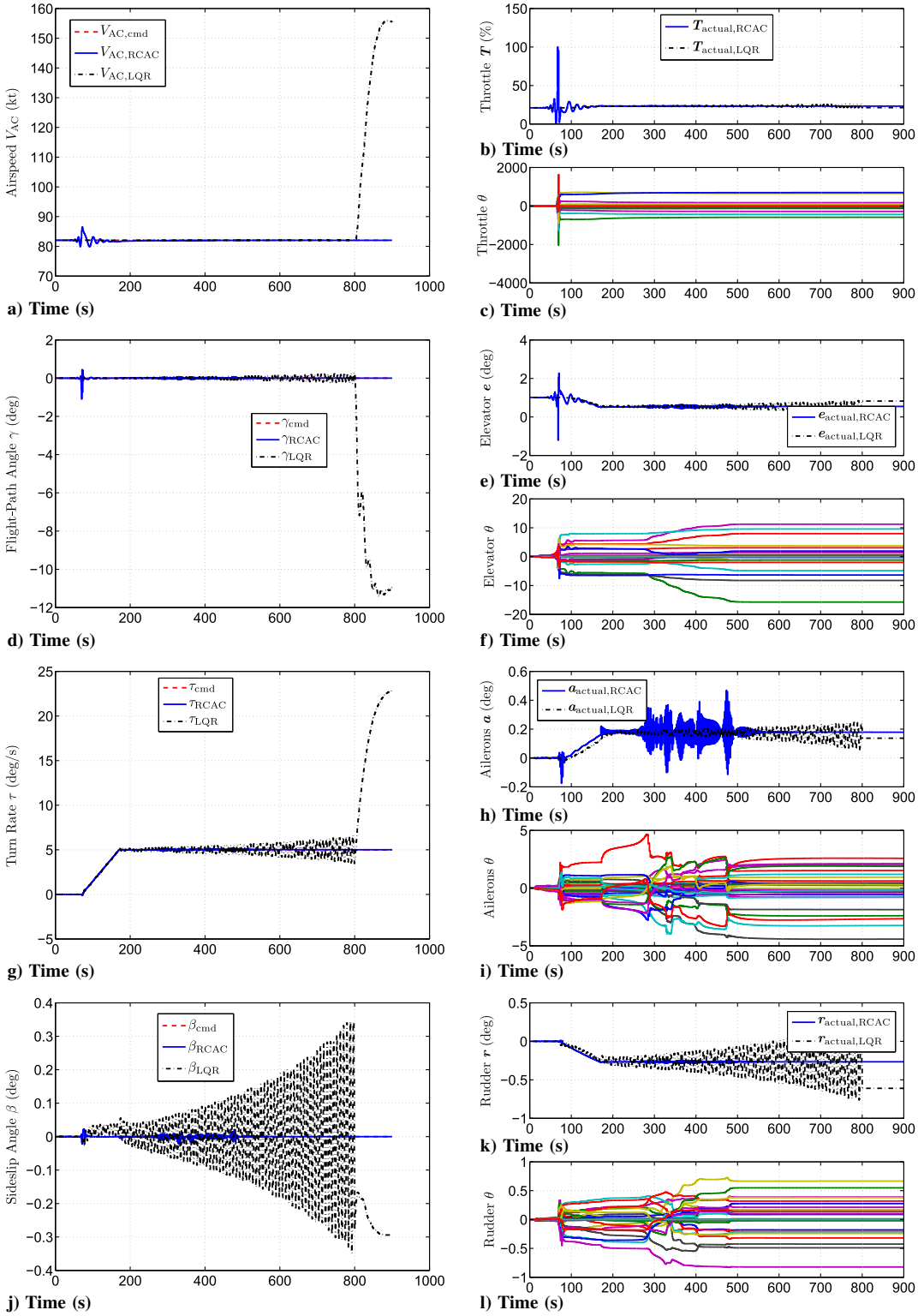

Fig. 14 Example A.3.2: Control architecture A_1 for horizontal circular flight with a modified initial trim. $V_{AC}(0)$ is decreased by 18.6 kt relative to the trim (47). LQR is not able to follow the incremental commands and diverges around $t = 800$ s.

Example B.1.3: For time-varying dynamics with decreasing mass, decreasing moments of inertia, and c.g. moving aft, we vary the c.g. location as in Example B.1.2 and we linearly decrease the mass and moments of inertia. The initial mass and moments of inertia of the GTM are 1.54 slug and $[1.32, 4.25, 5.4, 0.1125]$ lbf-ft^2/s, respectively. At 250 s, the mass decreases by 0.1 slug and the moments of inertia $[I_{xx}, I_{yy}, I_{zz}, I_{xz}]$ decrease by $[0.083, 0.26, 0.34, 0.0075]$ lbf $\cdot$ ft^2/s. After 250 s, the mass and moments of inertia are constant.

Figure 19 shows that the maximum command-following errors for airspeed, altitude, turn rate, and sideslip angle are 3.3 kt, 21.9 ft, 0.4 deg/s, and 0.05 deg, respectively, and the aircraft maintains constant airspeed and altitude. Note that the requested actuator settings in all four channels become constant after 250 s. ■

Example B.1.4: For time-varying dynamics with decreasing mass, decreasing moments of inertia, and c.g. moving forward, we vary the c.g. location as in Example B.1.1 and we vary the mass and moments of inertia as in Example B.1.3.

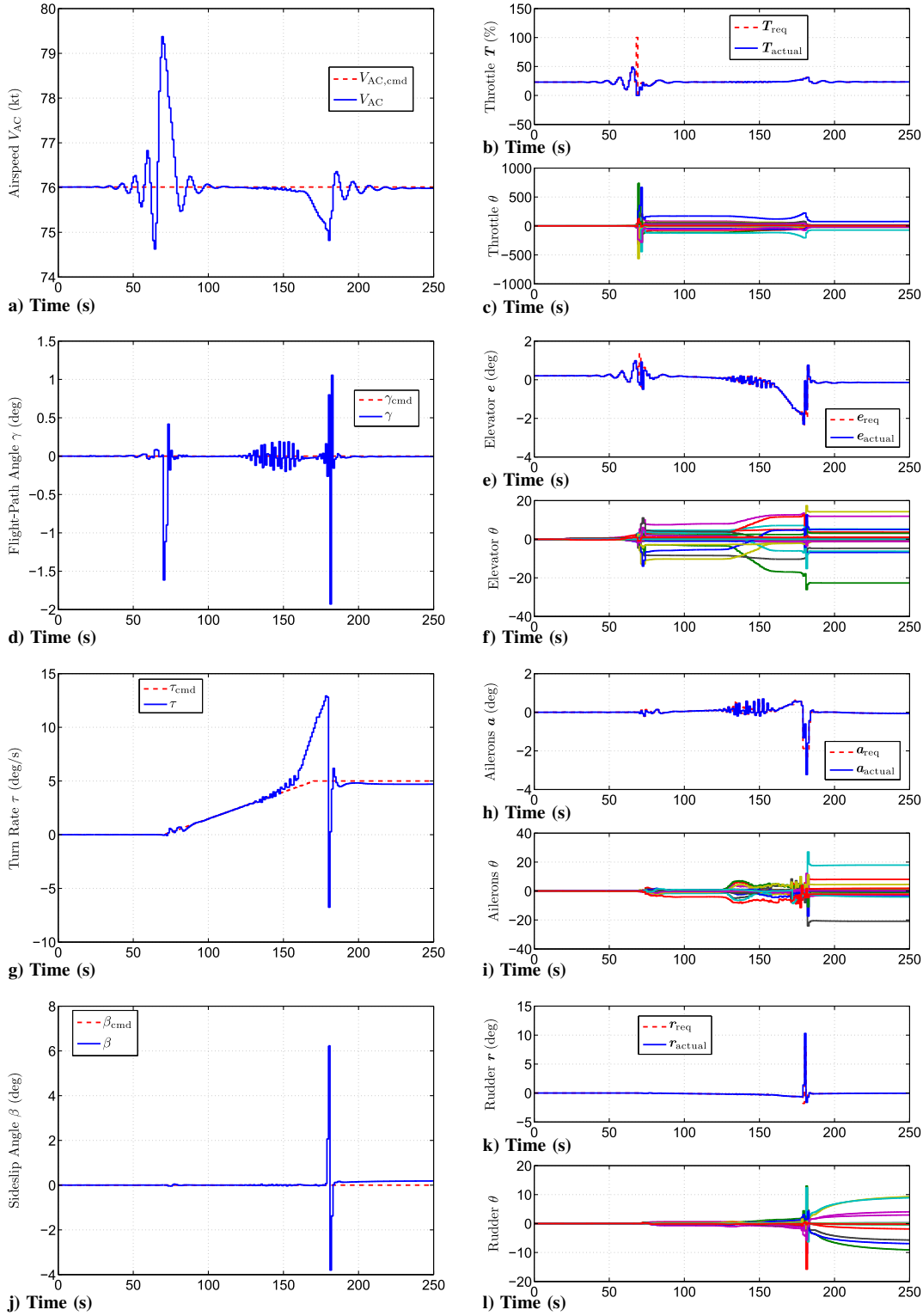

Fig. 15 Example A.3.3: Control architecture A_1 for horizontal circular flight with a modified initial trim. $V_{AC}(0)$ is decreased by 24.6 kt relative to the trim (47). Note that the decrement of 24.6 kt in $V_{AC}(0)$ causes an increase in the transient response; however, after 190 s, the command-following errors become small.

Figure 20 shows that the maximum command-following errors for airspeed, altitude, turn rate, and sideslip angle are 5.2 kt, 32.8 ft, 0.06 deg/s, and 0.02 deg, respectively, and the aircraft maintains constant airspeed and altitude. Note that the requested actuator settings in all four channels become constant after 250 s. ■

B. Scenario B.2: Icing

To model the increased drag and reduced lift due to icing, we linearly increase the aerodynamic force coefficient by 40% in the direction $\hat{i}_{AC}$ until 250 s, and we linearly decrease the aerodynamic force coefficient by 30% in the direction $-\hat{k}_{AC}$ until 250 s, respectively. After 250 s, we do not further alter the forces. This corresponds to a scenario where the ice forms in 250 s and then maintains its profile. For all of the examples in this scenario, we use control architecture A_2.

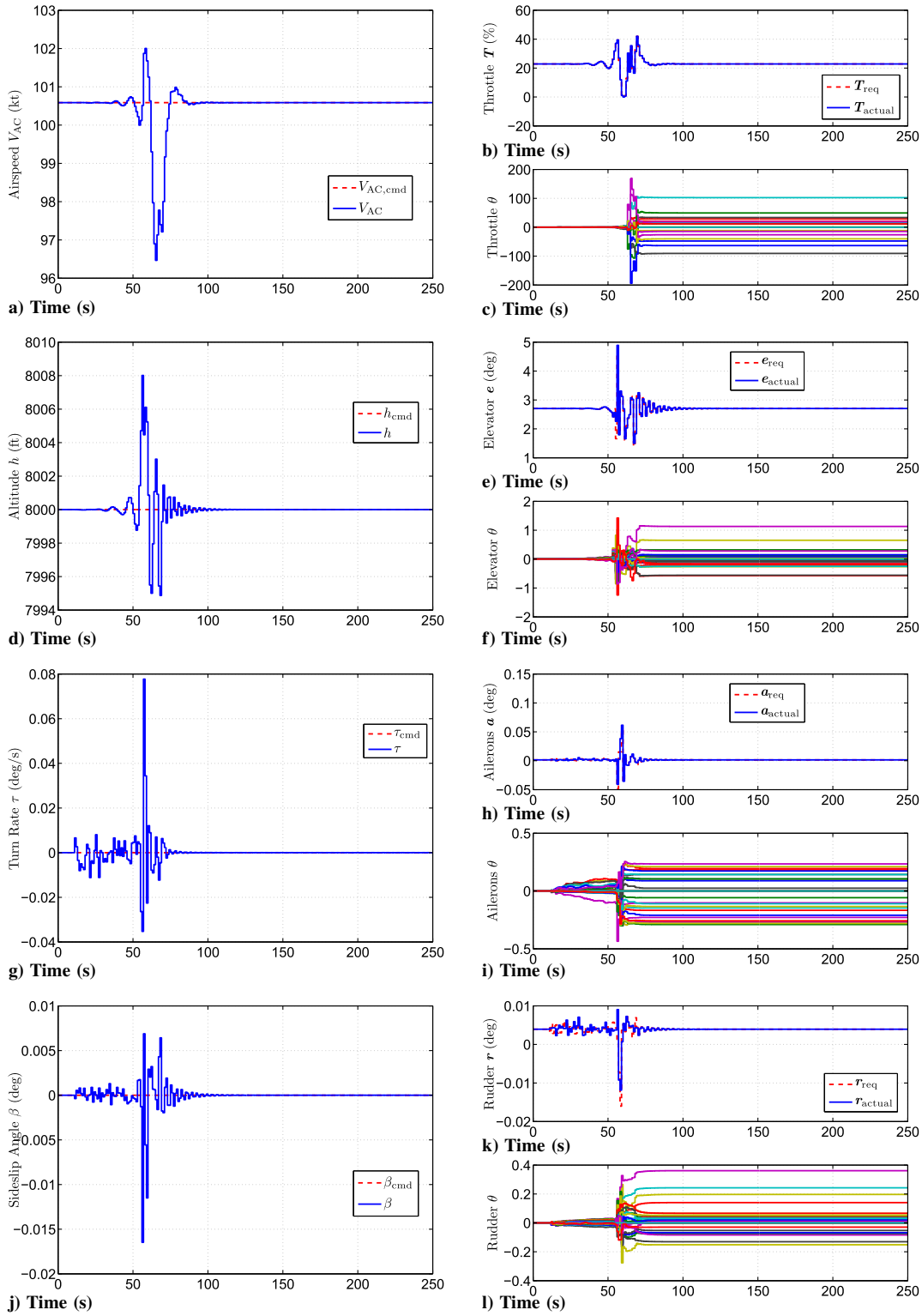

Fig. 16 Example B.1.1: Control architecture A_2 for straight-line flight with constant altitude and airspeed commands. This example is with fixed c.g. location to provide a baseline comparison for the subsequent examples involving the moving c.g. The transients are due to the actuator noise warmup, from $t = 10$ to 70 s.

Example B.2.1: For horizontal straight-line flight in the presence of icing with constant altitude and airspeed command, the incremental commands are given by

$$\delta V_{\mathrm{AC,cmd}}(k) = 0 \text{ kt}, \delta h_{\mathrm{cmd}}(k) = 0 \text{ ft}, \delta\tau_{\mathrm{cmd}}(k) = 0 \text{ deg/s}, \delta\beta_{\mathrm{cmd}}(k) = 0 \text{ deg} \tag{62}$$

Figure 21 shows that the maximum command-following errors for airspeed, altitude, turn rate, and sideslip angle are 7.7 kt, 38.4 ft, 1.7 deg/s, and 0.2 deg, respectively, and the aircraft maintains constant altitude and airspeed in straight-line flight. To compensate for the effect of icing, RCAC changes the throttle, elevator deflection, and aileron deflection from 22.8%, 2.7 deg, and 0.0014 deg at $t = 0$ s to 24.3%, 1.15 deg, and 0.005 deg at $t = 250$ s, respectively. After 250 s, there is no further ice formation and the requested actuator settings in all four channels become constant. ■

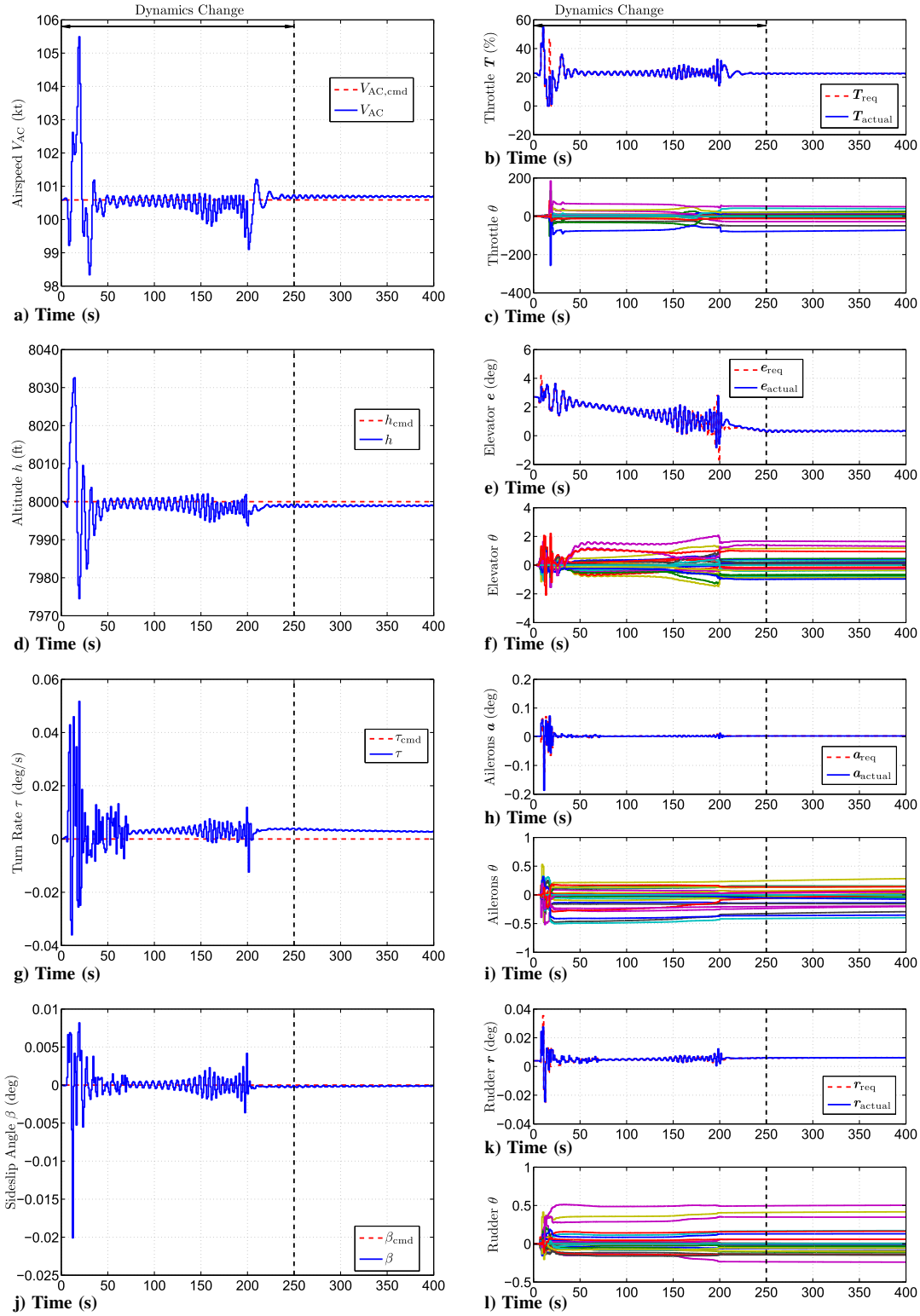

Fig. 17 Example B.1.1: Control architecture A_2 for straight-line flight with constant altitude and airspeed commands. The c.g. moves forward from 0 to 250 s and then remains fixed. Note that the requested actuator settings in all four channels become constant after 250 s.

Example B.2.2: For straight-line flight in the presence of icing with trapezoidal altitude command, followed by horizontal circular flight with trapezoidal turn-rate command and constant altitude command, the incremental commands are given by

$$\delta h_{\mathrm{cmd}}(k) = \begin{cases} 0, & k < 700, \\ \max\{-1000, -1(k-700)\}\mathrm{ft}, & k \geq 700, \end{cases}$$

$$\delta \tau_{\mathrm{cmd}}(k) = \begin{cases} 0, & k < 2700, \\ \min\{5, 0.005(k-2700)\}\ \mathrm{deg/s}, & k \geq 2700, \end{cases}$$

$$\delta V_{\mathrm{AC,cmd}}(k) = 0\ \mathrm{kt}, \delta\beta_{\mathrm{cmd}}(k) = 0\ \ \mathrm{deg} \tag{63}$$

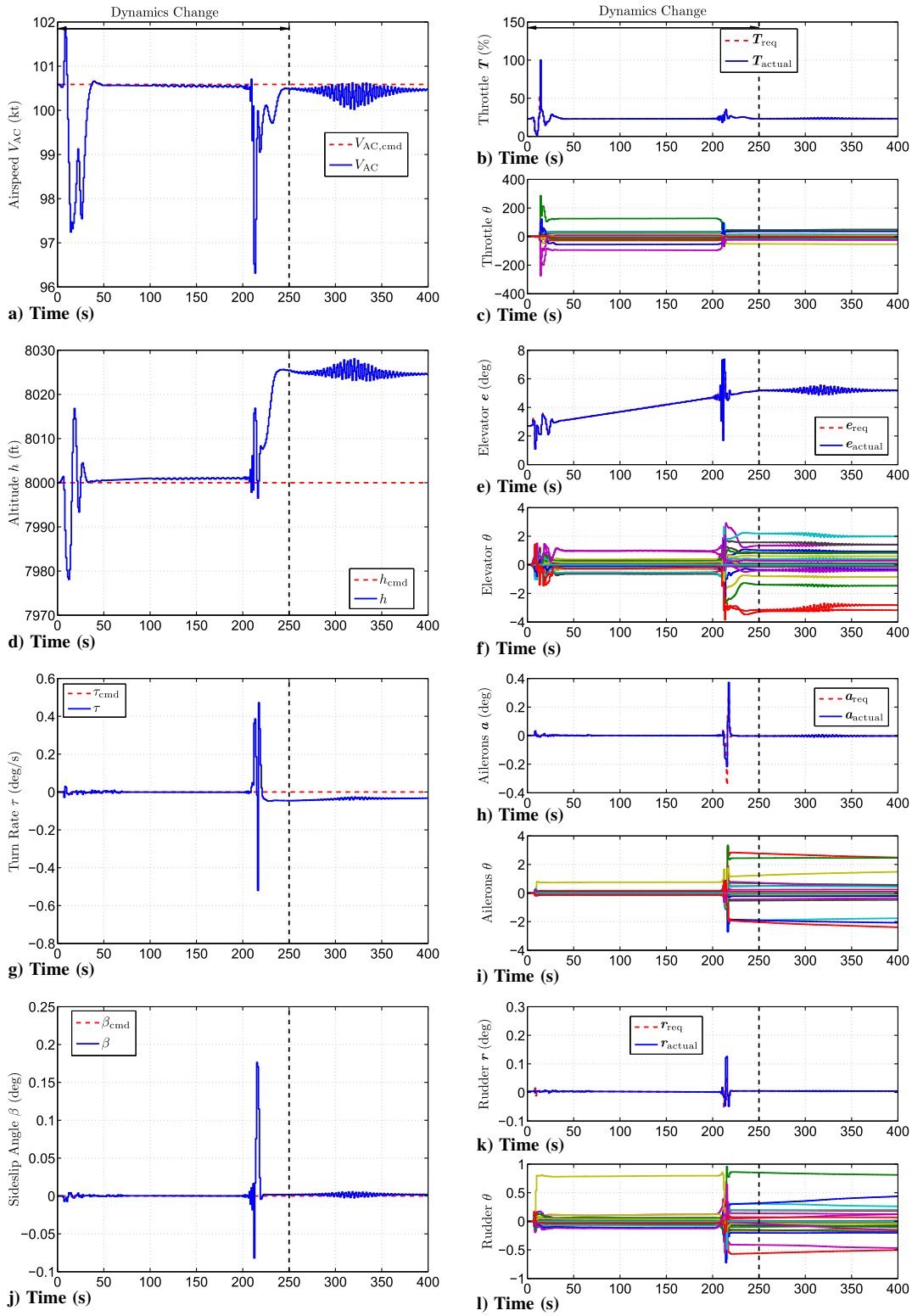

Fig. 18 Example B.1.2: Control architecture A_2 for straight-line flight with constant altitude and airspeed commands. The c.g. moves aft from 0 to 250 s and then remains fixed.

where the trapezoidal altitude command starts at $t = 70$ s and the trapezoidal turn-rate command starts at $t = 270$ s.

Figure 22 shows that the maximum command-following errors for airspeed, altitude, turn rate, and sideslip angle are 7.7 kt, 38.4 ft, 1.7 deg/s, and 0.2 deg, respectively, and the aircraft maintains constant heading and airspeed in straight-line flight until 270 s and afterward transitions to horizontal circular flight. Note that the requested actuator settings in all four channels are constant between 250 and 270 s as well as after 370 s. ■

VII. Case C: Control with Unknown Flight Envelope

A. Scenario C.1: Conflicting Trim Commands

We use architecture A_3, which extends architecture A_1 by including the angle of attack as a performance variable. For given airspeed and turn-rate commands, the commanded angle of attack and flight-path angle may be conflicting due to uncertainty in the achievable trim flight. To test the

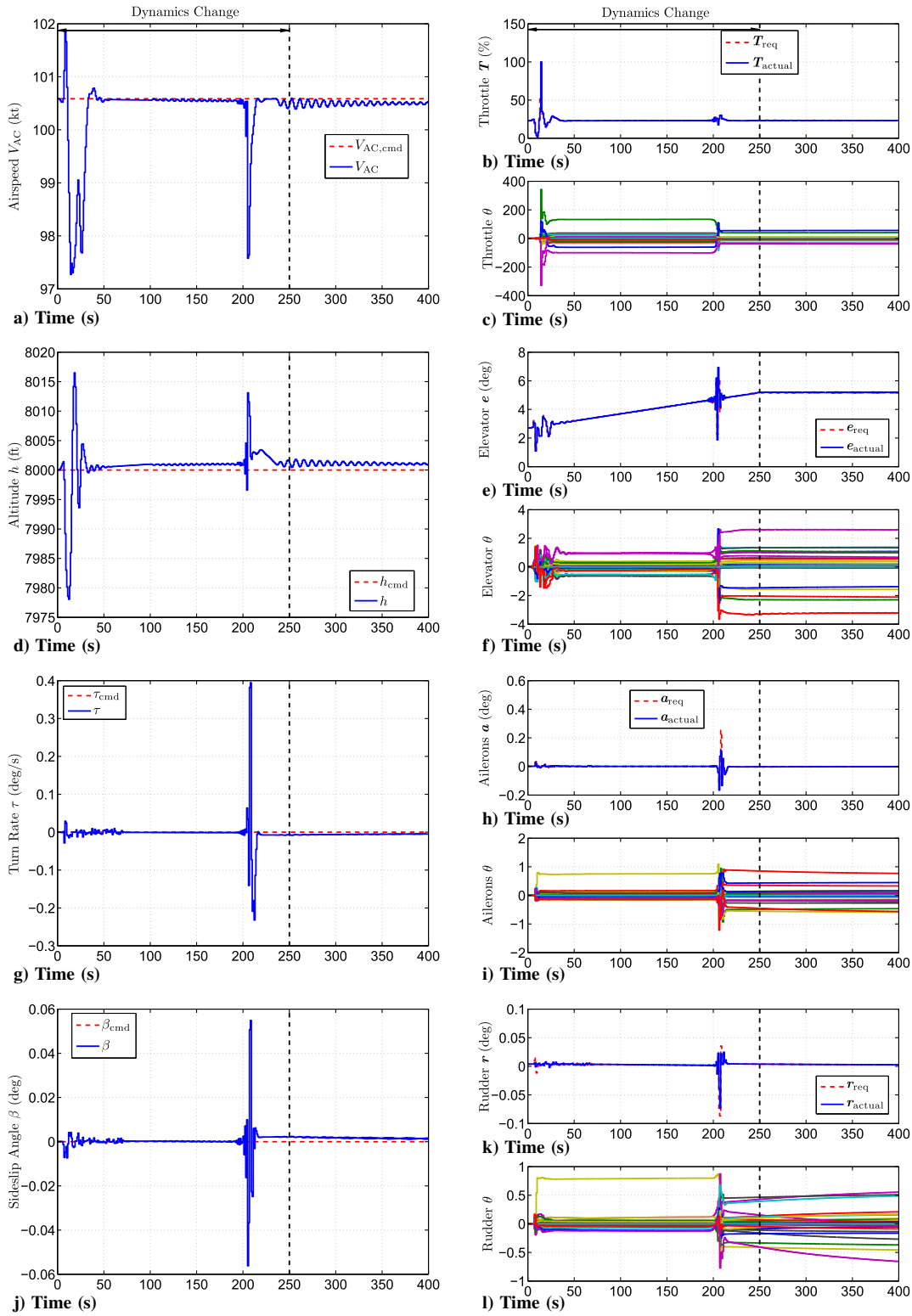

Fig. 19 Example B.1.3: Control architecture A_2 for straight-line flight with constant altitude and airspeed commands. The mass and moments of inertia decrease, and the c.g. moves aft from 0 to 250 s and then becomes constant.

robustness of RCAC to uncertainty in the flight envelope, the goal is thus to investigate the effect of possibly conflicting commands. In A_3, we use the elevator $\boldsymbol{e}$ to control both flight-path angle and angle of attack by adding the corresponding outputs of the adaptive controllers. The requested actuator setting for the elevator control channel (34) thus becomes

$$\boldsymbol{e}_{\mathrm{req}}(k) \stackrel{\Delta}{=} \boldsymbol{e}_{\mathrm{trim}} + \delta\boldsymbol{e}_{\mathrm{req},\gamma}(k) + \delta\boldsymbol{e}_{\mathrm{req},\alpha}(k) \tag{64}$$

where $\delta\boldsymbol{e}_{\mathrm{req},\gamma}(k)$ and $\delta\boldsymbol{e}_{\mathrm{req},\alpha}(k)$ are the outputs of the adaptive controllers for γ and α, respectively.

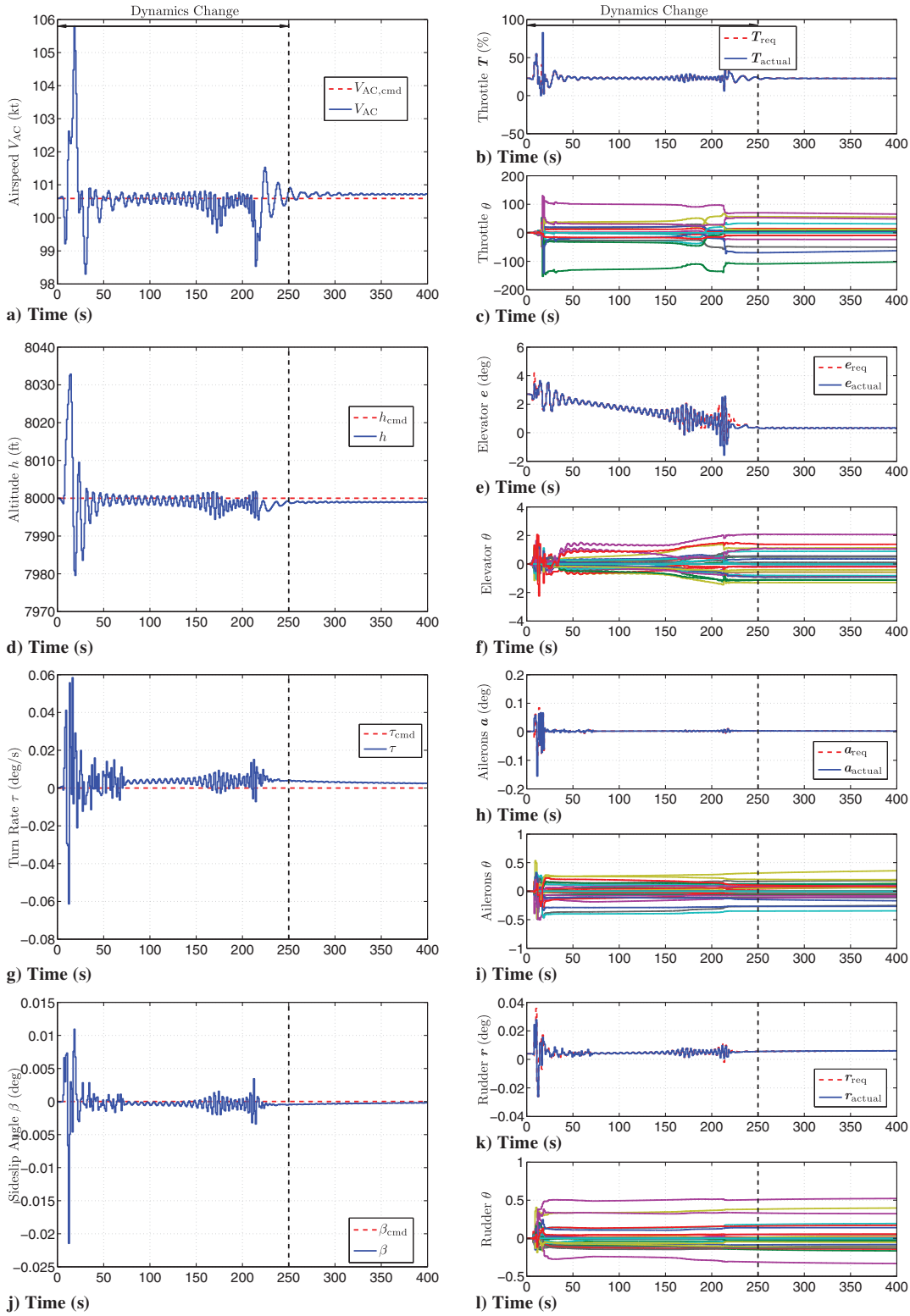

Fig. 20 Example B.1.4: Control architecture A_2 for straight-line flight with constant altitude and airspeed commands. The mass and moments of inertia decrease, and the c.g. moves forward from 0 to 250 s and then becomes constant.

Example C.1.1: For horizontal straight-line flight with zero flight-path-angle and trapezoidal angle-of-attack commands, the incremental commands are given by

$$\delta\alpha_{\text{cmd}}(k) = \begin{cases} 0, & k < 700, \\ \min\{1, 0.001(k-700)\}\,\text{deg}, & k \geq 700, \end{cases}$$

$$\delta V_{\text{AC,cmd}}(k) = 0\ \text{kt}, \delta\gamma_{\text{cmd}}(k) = 0\ \text{deg}, \delta\tau_{\text{cmd}}(k) = 0\ \text{deg/s}, \delta\beta_{\text{cmd}}(k) = 0\ \text{deg} \tag{65}$$

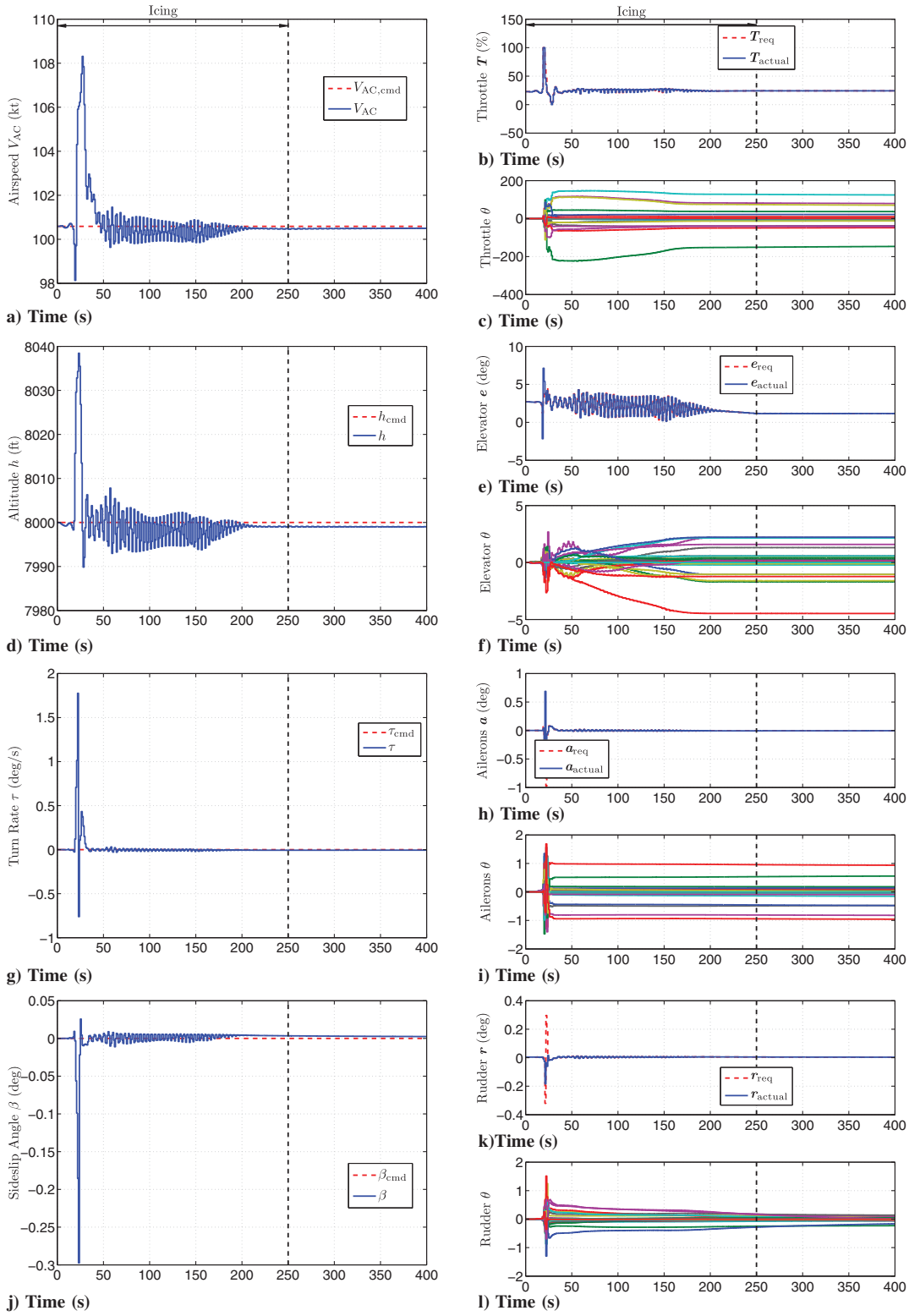

Fig. 21 Example B.2.1: Control architecture A_2 for horizontal straight-line flight with constant altitude and airspeed commands. Ice forms from 0 to 250 s and then maintains its profile. Note that the requested actuator settings in all four channels become constant after 250 s.

where the trapezoidal angle-of-attack command starts at $t = 70$ s. Note that, for the first 70 s, the commanded trim is as in Eq. (47), which indicates no conflict in the commands. After 70 s, the commanded angle of attack increases, whereas the airspeed, flight-path angle, turn rate, and sideslip angle are kept the same as in the trim (47). The angle-of-attack command thus conflicts with the remaining commands.

Figure 23 shows that the adaptive controller trims the aircraft at $V_{AC} = 91.2$ kt, $\gamma = 5$ deg, $\alpha = 4$ deg, $\tau = 0.06$ deg/s, and $\beta = 0.0009$ deg with the command-following errors $V_{AC} - V_{AC,cmd} = 9.35$ kt, $\gamma - \gamma_{cmd} = 5$ deg, $\alpha - \alpha_{cmd} = 0.02$ deg, $\tau - \tau_{cmd} = 0.06$ deg/s, and $\beta - \beta_{cmd} = 0.0009$ deg at $t = 250$ s. The adaptive controller resolves the conflicting commands at the expense of steady-state command-following errors in airspeed and flight-path angle. Note that all of the requested actuator settings are constant after the conflict is resolved. ■

Example C.1.2: For ascending straight-line flight with trapezoidal flight-path-angle and constant angle-of-attack commands, the incremental commands are given by

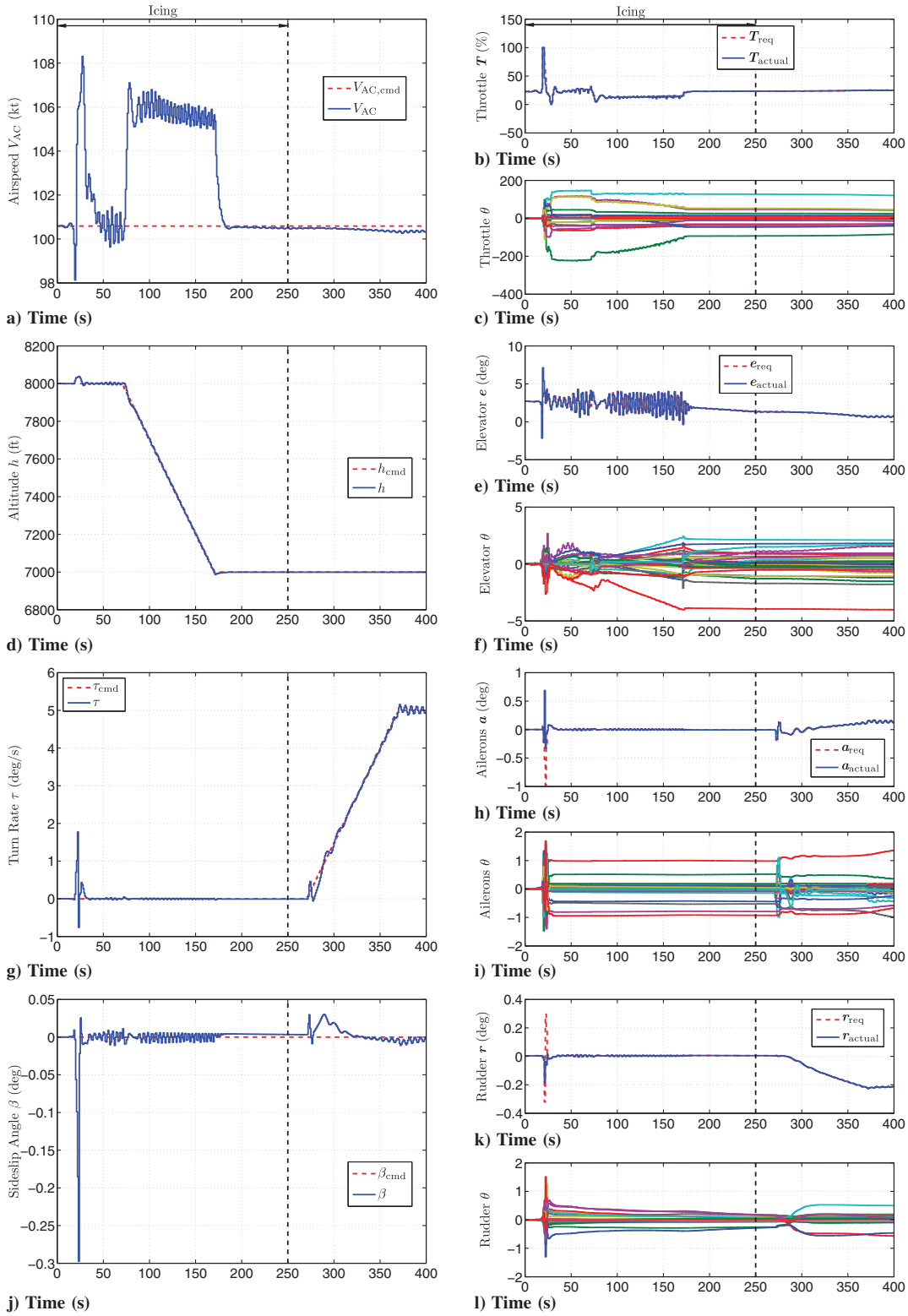

Fig. 22 Example B.2.2: Control architecture A_2 for straight-line flight with trapezoidal altitude command, followed by horizontal circular flight with trapezoidal turn-rate command and constant altitude command. Ice forms from 0 to 250 s and then maintains its profile.

$$\delta\gamma_{\text{cmd}}(k) = \begin{cases} 0, & k < 700, \\ \min\{5, 0.005(k-700)\} \text{ deg}, & k \geq 700, \end{cases}$$
$$\delta V_{\text{AC,cmd}}(k) = 0 \text{ kt}, \delta\alpha_{\text{cmd}}(k) = 0 \text{ deg}, \delta\tau_{\text{cmd}}(k) = 0 \text{ deg/s}, \delta\beta_{\text{cmd}}(k) = 0 \text{ deg} \tag{66}$$

where the trapezoidal flight-path-angle command starts at $t = 70$ s. Note that, for the first 70 s, the commanded trim is as in Eq. (47), which indicates no conflict in the commands. After 70 s, the commanded flight-path angle increases, whereas the airspeed, angle of attack, turn rate, and sideslip angle are kept the same as in the trim (47). The flight-path-angle command thus conflicts with the remaining commands.

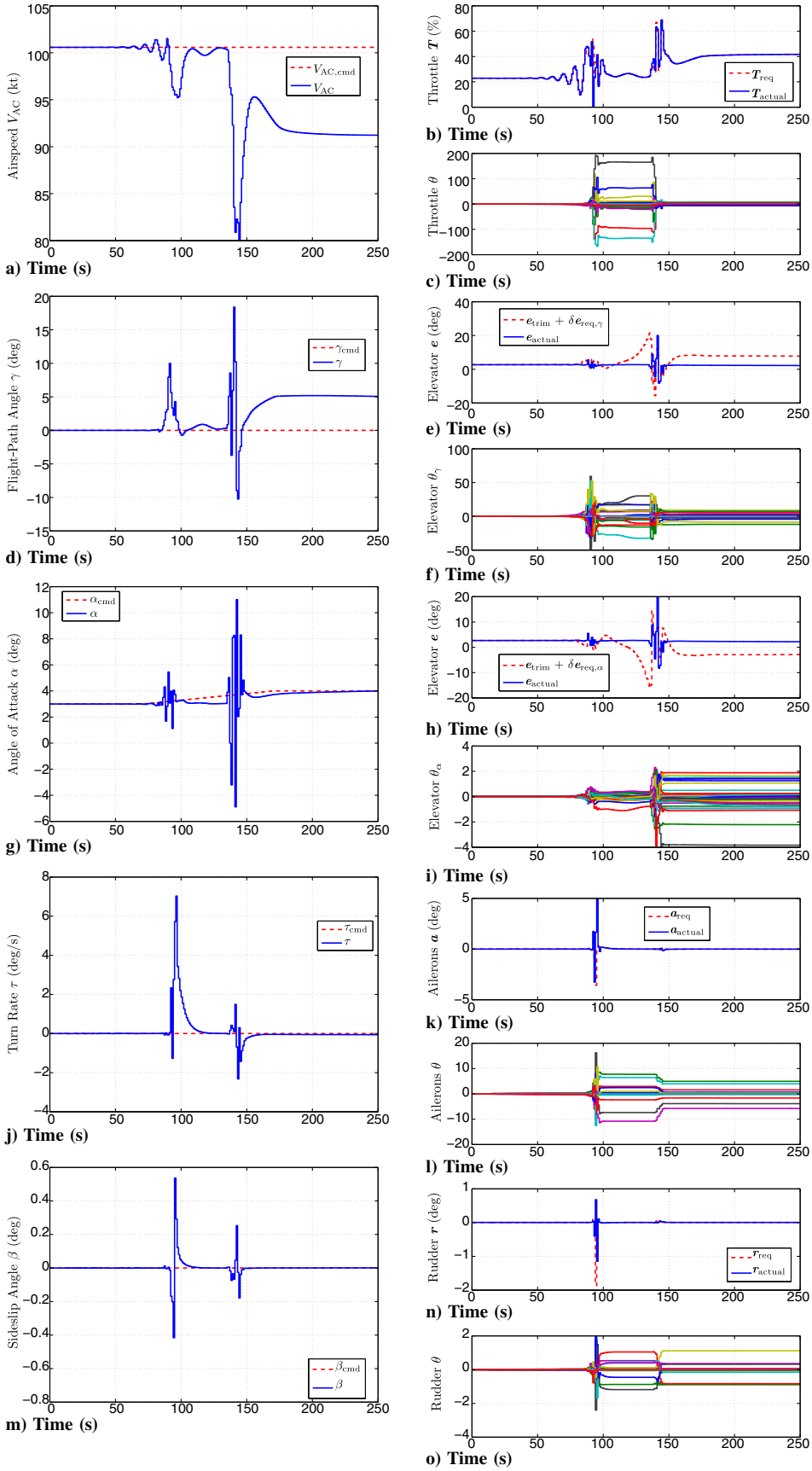

Fig. 23 Example C.1.1: Control architecture A_3 for horizontal straight-line flight with zero flight-path-angle and trapezoidal angle-of-attack commands. After 70 s, the commanded angle of attack increases, whereas V_{AC}, γ, τ, and β are kept unchanged. The angle-of-attack command thus conflicts with the remaining commands.

Figure 24 shows that the adaptive controller trims the aircraft at $V_{AC} = 96.7$ kt, $\gamma = 1.97$ deg, $\alpha = 3.34$ deg, $\tau = -0.02$ deg/s, and $\beta = 0$ deg with the command-following errors $V_{AC} - V_{AC,cmd} = 4.4$ kt, $\gamma - \gamma_{cmd} = 2.7$ deg, $\alpha - \alpha_{cmd} = 0.38$ deg, $\tau - \tau_{cmd} = 0.02$ deg/s, and $\beta - \beta_{cmd} = 0$ deg at $t = 250$ s. The adaptive controller resolves the conflicting commands at the expense of steady-state command-following errors in airspeed, flight-path angle, and angle of attack. Note that all of the requested actuator settings are constant after the conflict is resolved.■

B. Scenario C.2: Trim Flight Stability

We investigate the open-loop stability of the trim flight achieved by the adaptive controller without knowing the flight envelope (see Sec. II.J for details on trim flight) using control architecture A_4. We consider the following two possibilities for stability of the trim flight:

1) For asymptotically stable trim, the aircraft returns to the trim if the control surfaces are slightly perturbed and then set back to their trim angles.

2) For not asymptotically stable trim, the aircraft does not return to the trim if the control surfaces are slightly perturbed and then set back to their trim angles.

The GTM is initialized with the trim

$$\begin{aligned} V_{AC}(0) = V_{AC,trim} = 67.36 \text{ kt}, \gamma(0) = \gamma_{trim} = 0 \text{ deg}, \tau(0) = \tau_{trim} = 0 \text{ deg/s}, \\ \beta(0) = \beta_{trim} = 0 \text{ deg}, \alpha(0) = \alpha_{trim} = 8 \text{ deg}, (0) = 8000 \text{ ft} \end{aligned} \tag{67}$$

for all of the examples in Scenario C.2.

Example C.2.1: For horizontal circular flight with trapezoidal airspeed command and trapezoidal turn-rate command, the incremental commands are given by

$$\begin{aligned} \delta V_{AC,cmd}(k) &= \begin{cases} 0, & k < 700, \\ \min\{5, 0.005(k-700)\} \text{ kt}, & k \geq 700, \end{cases} \\ \delta\tau_{cmd}(k) &= \begin{cases} 0, & k < 2500, \\ \min\{4, 0.004(k-2500)\} \text{ deg/s}, & 2500 \leq k < 4000, \\ \min\{6, 4 + 0.002(k-4000)\} \text{ deg/s}, & k \geq 4000, \end{cases} \\ \delta\gamma_{cmd}(k) &= 0 \text{ deg} \end{aligned} \tag{68}$$

where the airspeed and turn-rate commands start at $t = 70$ and 250 s, respectively.

At $t = 900$ s, the adaptation is disabled and the control surfaces are fixed at their trim angles. At $t = 1000$ s, the elevator and aileron are perturbed by 0.5 deg for 0.2 s, respectively, from their trim angles. Figure 25 shows that the aircraft is in trim flight from $t = 900$ to 1000 s. After the perturbation, the aircraft returns to the same trim, indicating that the trim achieved by the adaptive controller is asymptotically stable. ■

Example C.2.2: For horizontal circular flight with trapezoidal airspeed command and trapezoidal turn-rate command, the incremental commands are given by

$$\begin{aligned} \delta V_{AC,cmd}(k) &= \begin{cases} 0, & k < 700, \\ \max\{-5, -0.005(k-700)\} \text{ kt}, & k \geq 700, \end{cases} \\ \delta\tau_{cmd}(k) &= \begin{cases} 0, & k < 2500, \\ \min\{4, 0.004(k-2500)\} \text{ deg/s}, & 2500 \leq k < 4000, \\ \min\{6, 4 + 0.002(k-4000)\} \text{ deg/s}, & k \geq 4000, \end{cases} \\ \delta\gamma_{cmd}(k) &= 0 \text{ deg} \end{aligned} \tag{69}$$

where the airspeed and turn-rate commands start at $t = 70$ and 250 s, respectively.

At $t = 900$ s, the adaptation is disabled and the control surfaces are fixed at their trim angles. At $t = 1000$ s, the elevator and aileron are perturbed by 0.5 deg for 0.2 s, respectively, from their trim angles. Figure 26 shows that the aircraft is in trim flight from $t = 900$ to 1000 s. After the perturbation, the aircraft does not return to the same trim but converges to a new trim, indicating that the trim achieved by the adaptive controller is not asymptotically stable. This shows the ability of RCAC to actively stabilize trim flight that is not asymptotically stable. ■

C. Scenario C.3: Flight Envelope Exploration

We now explore and map the flight envelope using architecture A_4. The exploration scheme consists of the following steps, which maneuver the aircraft around the flight envelope and determine the stability of the attained trim states:

1) Start the aircraft at an initial trim condition.

2) Specify a trim command.

3) Increase or decrease the airspeed to the desired value by a sequence of intermediate trapezoidal commands with a slope of 0.1 kt/s and sufficient dwell time for the aircraft to reach the command; $\delta\tau_{cmd}$ and $\delta\gamma_{cmd}$ are zero during this step.

4) Increase or decrease the turn rate to the desired value by a sequence of intermediate trapezoidal commands with a slope of 0.1 deg/s^2 and sufficient dwell time for the aircraft to reach the command; $\delta\gamma_{cmd}$ is zero during this step.

5) After reaching the desired trim, return to open-loop flight by disabling the adaptation and freezing all the actuators and throttle settings.

6) Impulse the elevator and aileron to observe the stability of the aircraft at the new trim state.

Figure 27 shows the mapping of the flight envelope after the exploration scheme is completed. Starting from the initial trim (67), we search through a slice of the flight envelope by keeping flight-path-angle zero while varying airspeed and turn rate, as indicated by the arrows. Note that, during this search, the tuning settings of RCAC remain unchanged and the aircraft transitions between different trims without gain scheduling, as is traditionally done in practice for transitioning between different trims in the flight envelope. The region of attained trim states consists mainly of asymptotically stable trims, with trims that are not asymptotically stable at the boundary.

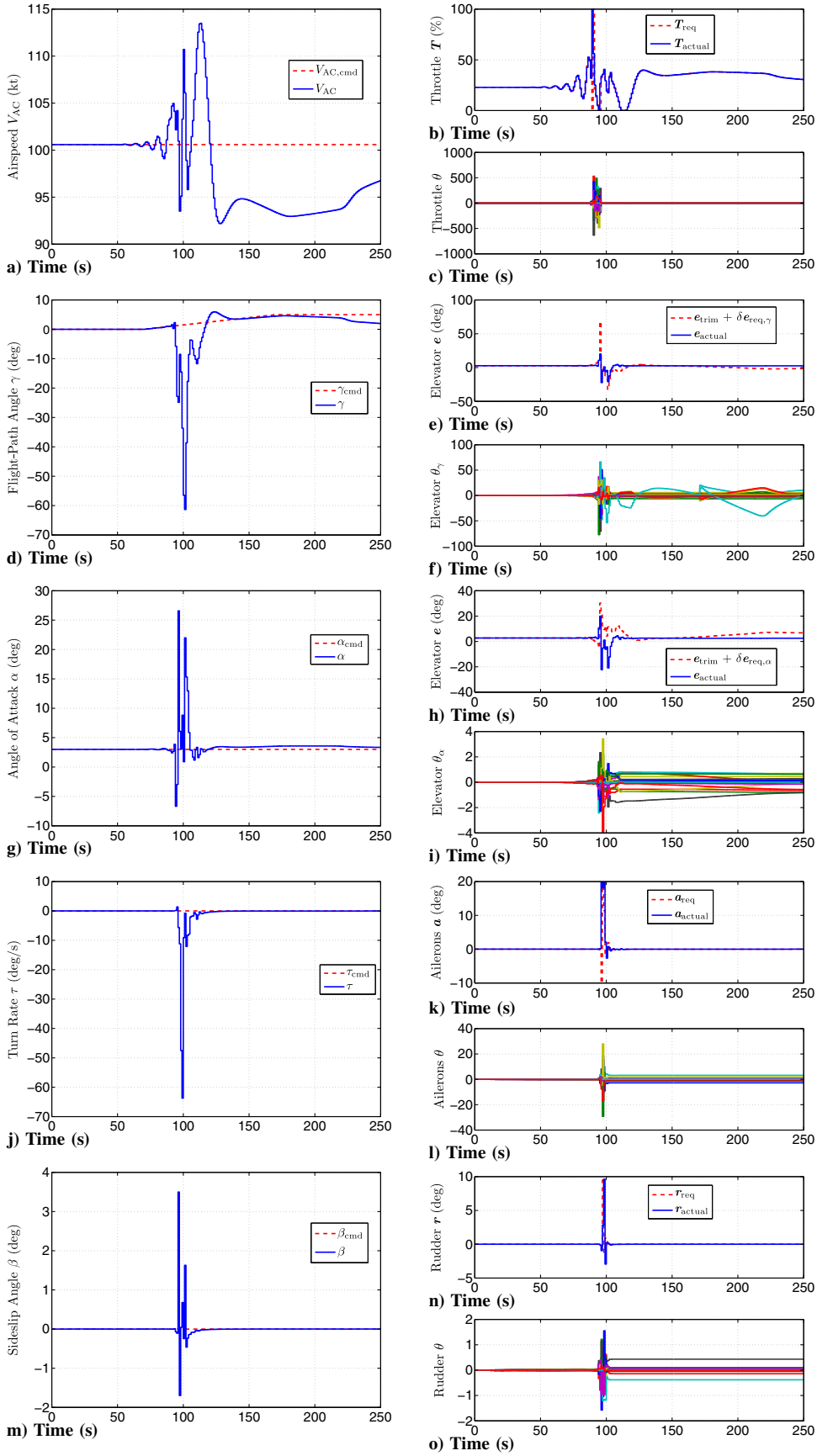

Fig. 24 Example C.1.2: Control architecture A_3 for ascending straight-line flight with trapezoidal flight-path-angle and constant angle-of-attack commands. After 70 s, the commanded flight-path angle increases, whereas V_{AC}, α, τ, and β are kept unchanged. The flight-path-angle command thus conflicts with the remaining commands.

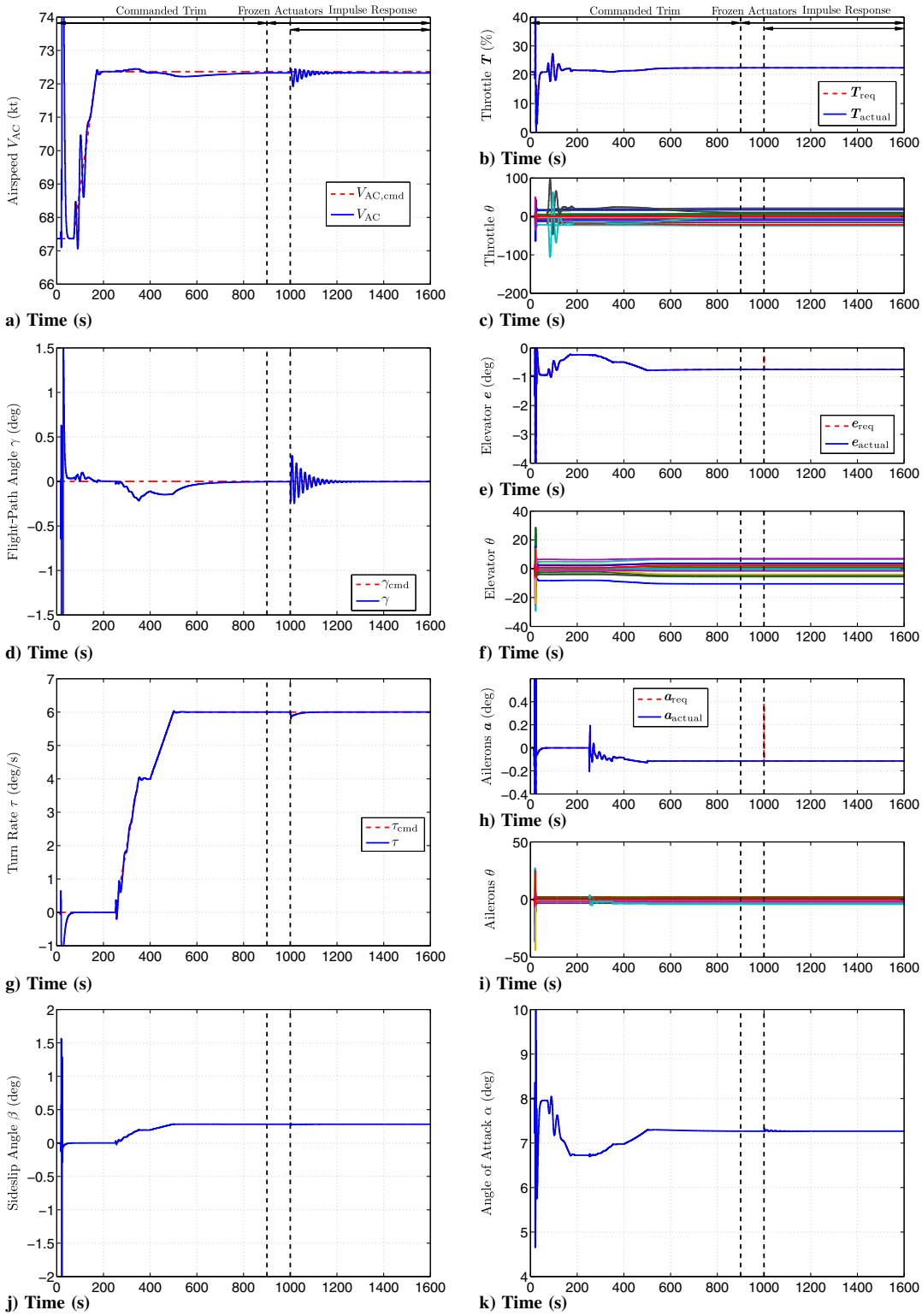

Fig. 25 Example C.2.1: Control architecture A_4 for horizontal circular flight with trapezoidal airspeed command and trapezoidal turn-rate command. At $t = 900$ s, the adaptation is disabled and the control surfaces are fixed at their trim angles. To determine the stability of the achieved trim at $t = 1000$ s, the elevator and aileron are perturbed by 0.5 deg for 0.2 s, respectively, from their trim values.

VIII. Case D: Control with Known Sensing Constraints

We investigate Scenario D.1 where the air data system (ADS) fails during flight, and thus no angle-of-attack, sideslip-angle, or airspeed measurements are available. We thus rely on GPS and inertial measurement unit data to follow a desired ground position trajectory. To do so, we use control architecture A_5, which uses no ADS measurements, as shown in Eq. (A3).

Example D.1.1: For reference flight-trajectory following in the presence of wind shear, the reference trajectory is composed of ascending, right-turn, and straight-line maneuvers. The incremental command δh_{cmd} is

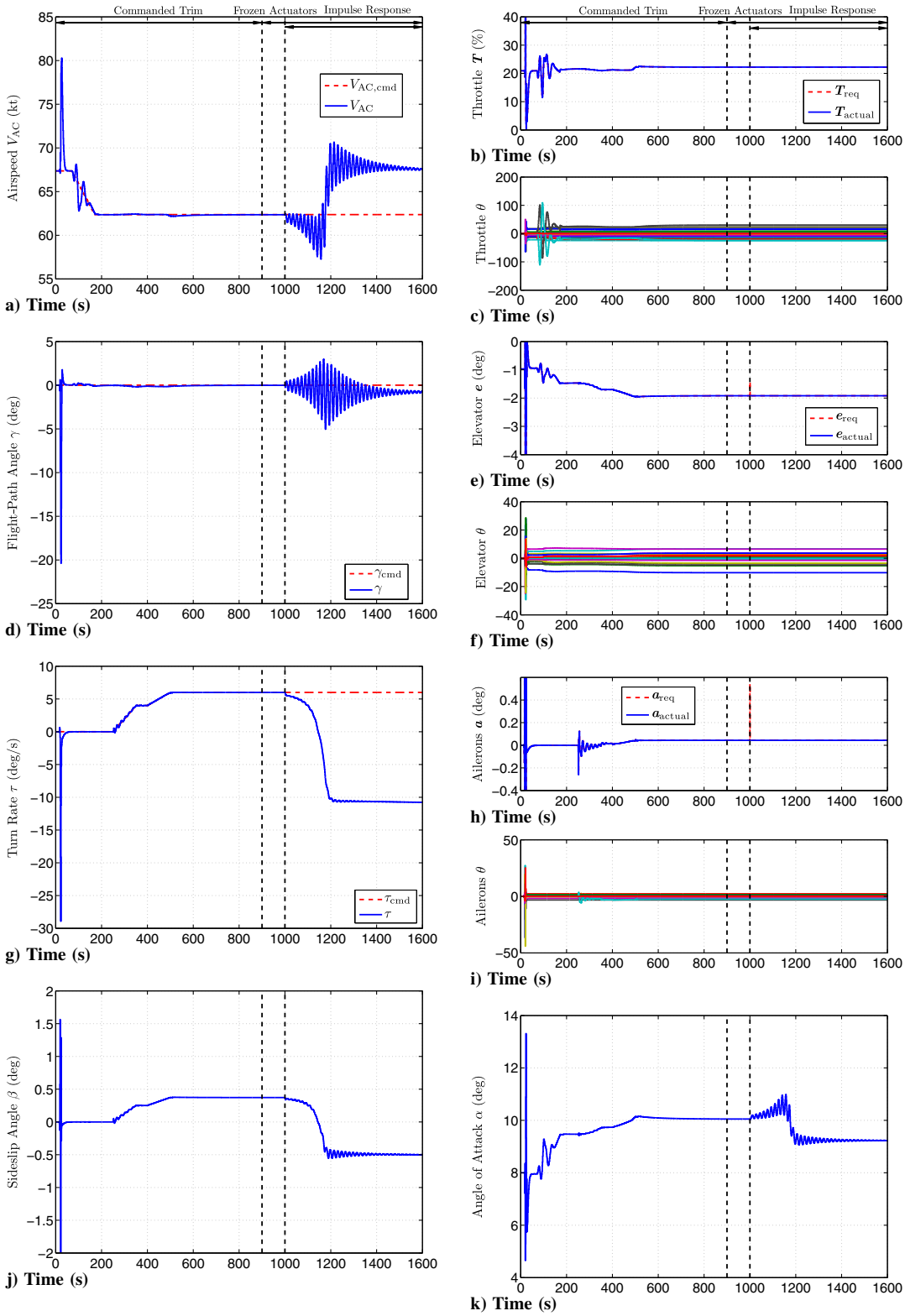

Fig. 26 Example C.2.2: Control architecture A_4 for horizontal circular flight with trapezoidal airspeed command and trapezoidal turn-rate command. At $t = 900$ s, the adaptation is disabled and the control surfaces are fixed at their trim angles. To determine the stability of the achieved trim at $t = 1000$ s, the elevator and aileron are perturbed by 0.5 deg for 0.2 s, respectively, from their trim values.

$$\delta h_{\mathrm{cmd}}(k) = \begin{cases} 0, & k < 700, \\ \min\{500, 0.5(k-700)\}\ \mathrm{ft}, & k \geq 700 \end{cases} \tag{70}$$

where the trapezoidal altitude command starts at $t = 70$ s.

Figure 28 shows that the maximum command-following errors for $e_{\parallel}$, h, $e_{\perp}$, and η are 373, 119, and 195 ft, and 5.3 deg, respectively, whereas the GTM follows the reference flight trajectory. The maximum angle of attack, sideslip angle, and airspeed (not shown in Fig. 28) are 13 deg, 0.2 deg, and 129 kt, respectively. ■

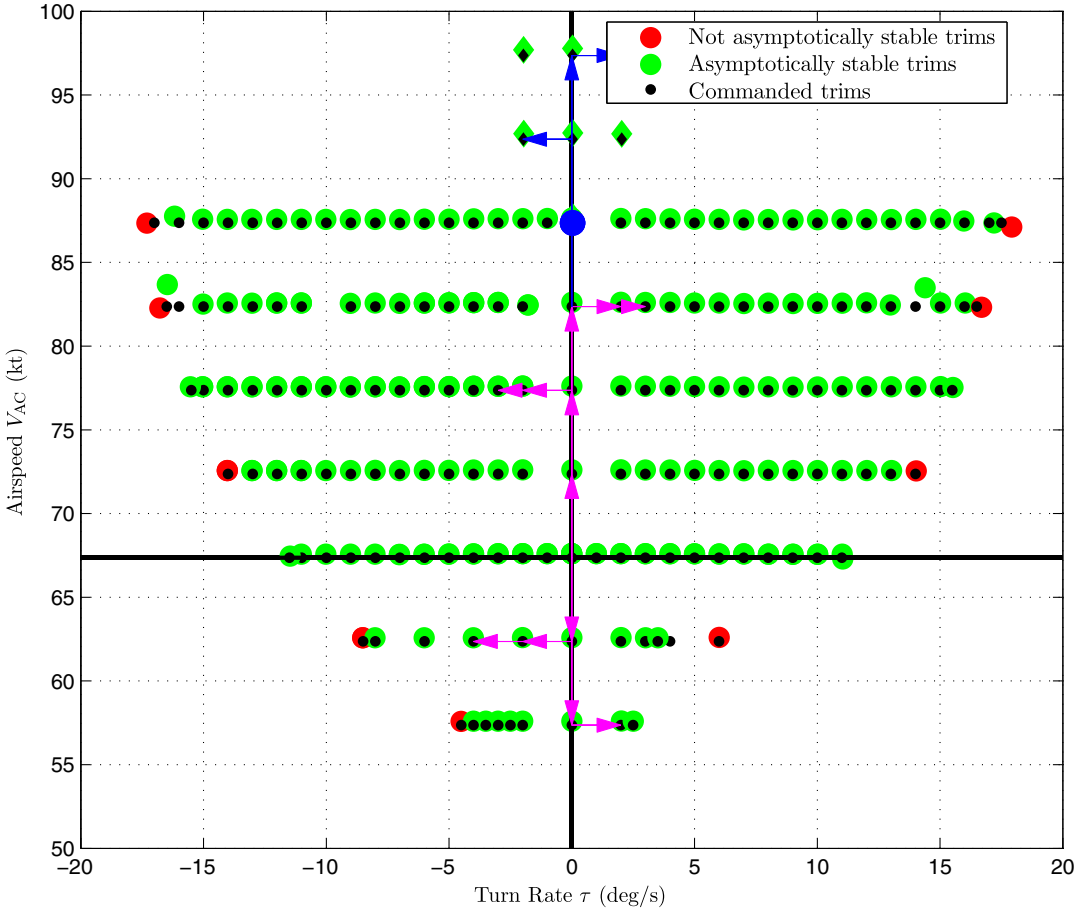

Fig. 27 Region of the attained trim states starting from the initial trim (67) and keeping the flight-path angle zero. Further exploration starting from a different initial condition, indicated by the blue dot, is also shown.

Example D.1.2: For reference flight-trajectory following in the presence of wind shear, the reference trajectory is composed of descending, left-turn, and straight-line maneuvers. The incremental command δh_{cmd} is

$$\delta h_{\text{cmd}}(k) = \begin{cases} 0, & k < 700, \\ \max\{-1000, -1(k-700)\} \text{ ft}, & k \geq 700 \end{cases} \tag{71}$$

where the trapezoidal altitude command starts at $t = 70$ s.

Figure 29 shows that the maximum command-following errors for $e_{\parallel}$, h, $e_{\perp}$, and η are 370, 52, and 147 ft, and 1.9 deg, respectively, whereas the GTM follows the reference flight trajectory. The maximum angle of attack, sideslip angle, and airspeed (not shown in Fig. 29) are 9.4 deg, 0.08 deg, and 115 kt, respectively. ■

IX. Case E: Control with Known Actuation Constraints

We investigate Scenario E.1, where the aircraft is initially flying in the straight-line trim given by Eq. (47) and the ailerons and rudder become stuck at their trim deflections in the initial trim flight beginning at $t = 0$ s. The goal is to maintain lateral flight control using differential thrust with control architecture A_6. The requested actuator setting increments for the left and right engines are $\delta \boldsymbol{T}_{\text{req}}$ and $-\delta \boldsymbol{T}_{\text{req}}$, respectively.

Example E.1.1: For horizontal circular flight with trapezoidal turn-rate command and constant altitude command, the incremental commands are given by

$$\delta\tau_{\text{cmd}}(k) = \begin{cases} 0, & k < 700, \\ \min\{2, 0.002(k-700)\} \text{ deg/s}, & k \geq 700, \end{cases} \quad \delta h_{\text{cmd}}(k) = 0 \text{ ft} \tag{72}$$

where the trapezoidal turn-rate command starts at $t = 70$ s.

Figure 30 shows that the maximum command-following errors for turn rate and altitude are 2.1 deg/s and 8.2 ft, respectively, and the aircraft maintains constant turn rate and altitude in horizontal flight. At $t = 400$ s, the sideslip angle is 0.03 deg, the roll angle (not shown) is 10.2 deg, and the command-following errors for turn rate and altitude are 0.05 deg/s and 0.2 ft, respectively. ■

Example E.1.2: For helical flight around a vertical axis with trapezoidal turn-rate command and trapezoidal altitude command, the incremental commands are given by

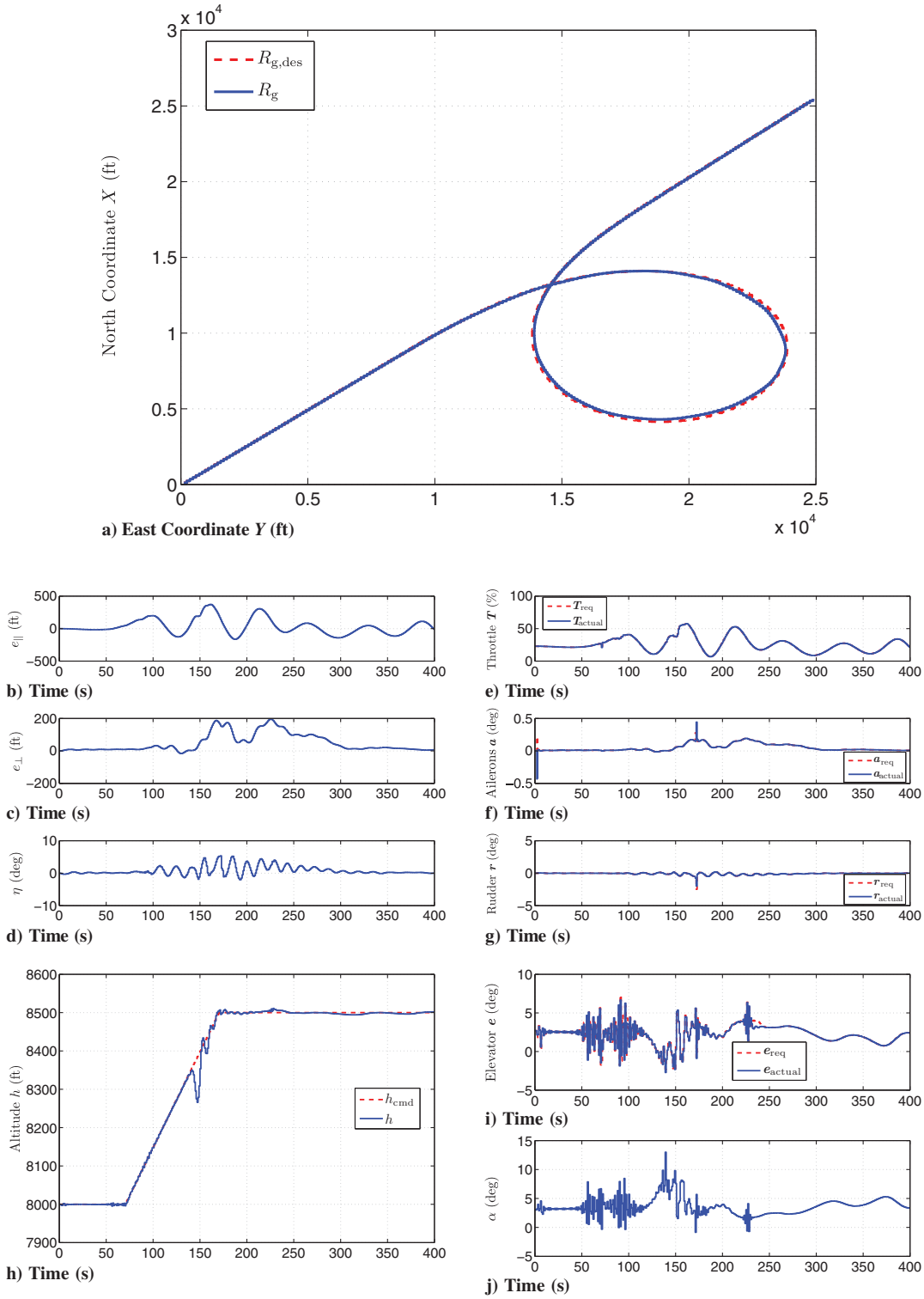

Fig. 28 Example D.1.1: Control architecture A_5 for ascending flight-trajectory following in the presence of wind shear. The reference trajectory is composed of ascending, right-turn, and straight-line maneuvers. The GTM follows the reference flight trajectory.

$$\delta\tau_{\text{cmd}}(k) = \begin{cases} 0, & k < 700, \\ \min\{2, 0.002(k-700)\} \ \text{deg/s}, & k \geq 700, \end{cases}$$
$$\delta h_{\text{cmd}}(k) = \begin{cases} 0, & k < 700, \\ \max\{-1000, -1(k-700)\} \ \text{ft}, & k \geq 700 \end{cases} \tag{73}$$

where the trapezoidal turn-rate and altitude commands start at $t = 70$ s.

Figure 31 shows that the maximum command-following errors for turn rate and altitude are 2.4 deg/s and 46.4 ft, respectively, and the aircraft maintains constant turn rate and altitude in horizontal flight. At $t = 400$ s, the sideslip angle is 0.03 deg, the roll angle (not shown) is 10.2 deg, and the command-following errors for turn rate and altitude are 0.05 deg/s and 0.08 ft, respectively. ■

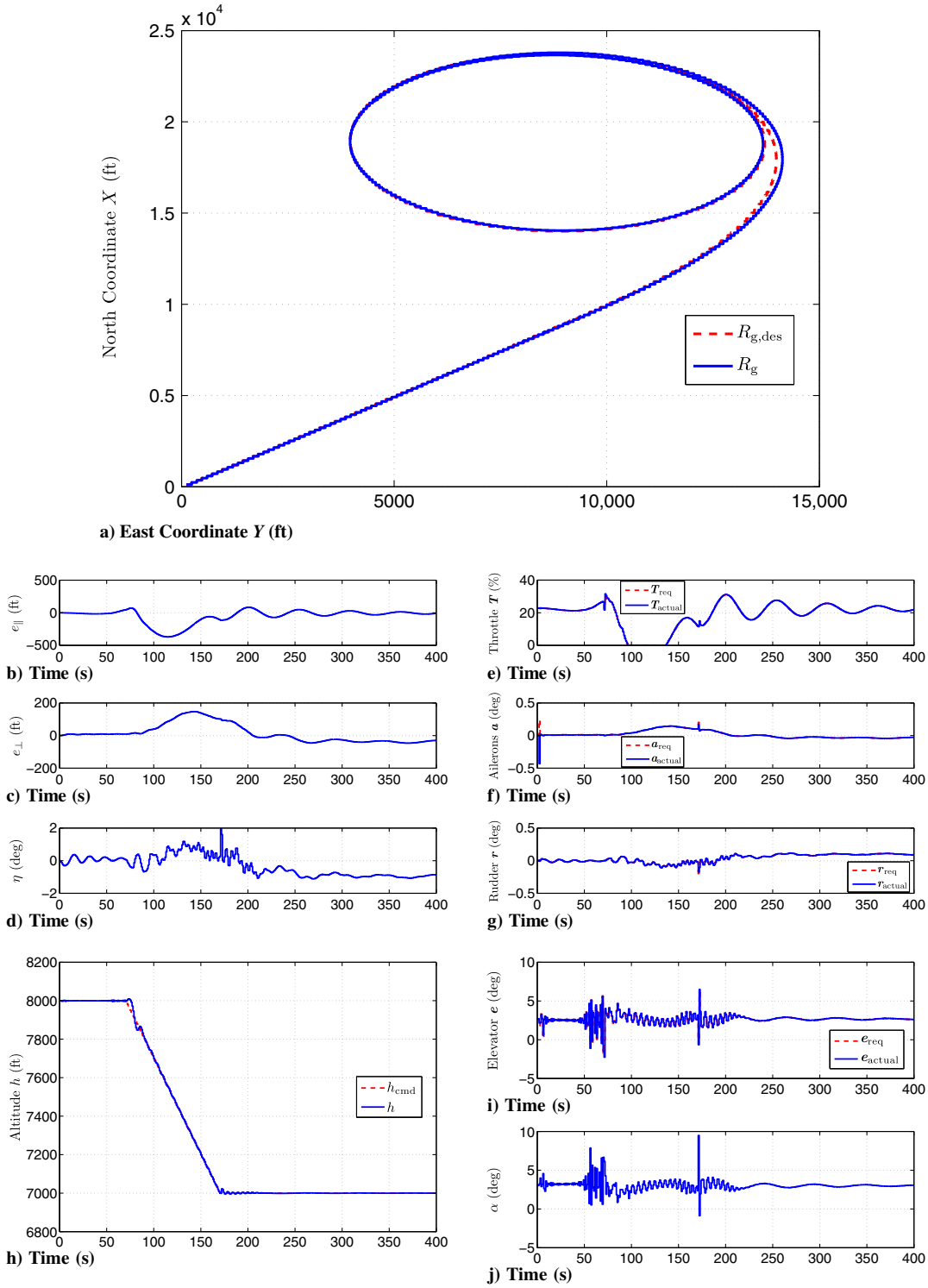

Fig. 29 Example D.1.2: Control architecture A_5 for descending flight-trajectory following in the presence of wind shear. The reference trajectory is composed of descending, left-turn, and straight-line maneuvers. The GTM follows the reference flight trajectory.

X. Case F: Control with Unknown Sensor Failure

A. Scenario F.1: Stuck Sensor

Example F.1.1: For horizontal straight-line flight with stuck pitot-tube measurement after $t = 50$ s using control architecture A_2, the incremental commands are given by

$$\delta V_{\mathrm{AC,cmd}}(k) = \begin{cases} 0, & k < 700, \\ \min\{5, 0.005(k-700)\}\ \mathrm{kt}, & k \geq 700, \end{cases}$$
$$\delta h_{\mathrm{AC,cmd}}(k) = 0\ \mathrm{ft}, \delta\tau_{\mathrm{cmd}}(k) = 0\ \ \mathrm{deg/s}, \delta\beta_{\mathrm{cmd}}(k) = 0\ \ \mathrm{deg} \tag{74}$$

where the trapezoidal airspeed command starts at $t = 70$ s.

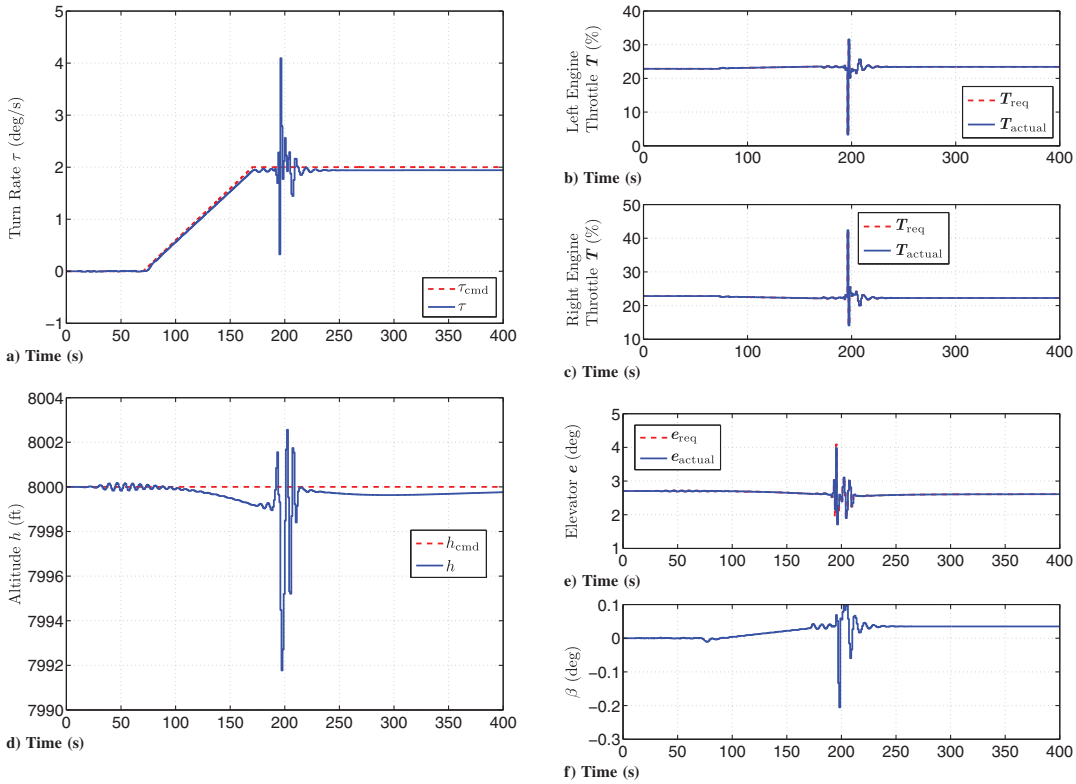

Fig. 30 Example E.1.1: Control architecture A_6 for horizontal circular flight with trapezoidal turn-rate command and constant altitude command. Beginning at $t = 0$ s, the ailerons and rudder become stuck at their trim deflections. The turn-rate command following is recovered using differential thrust.

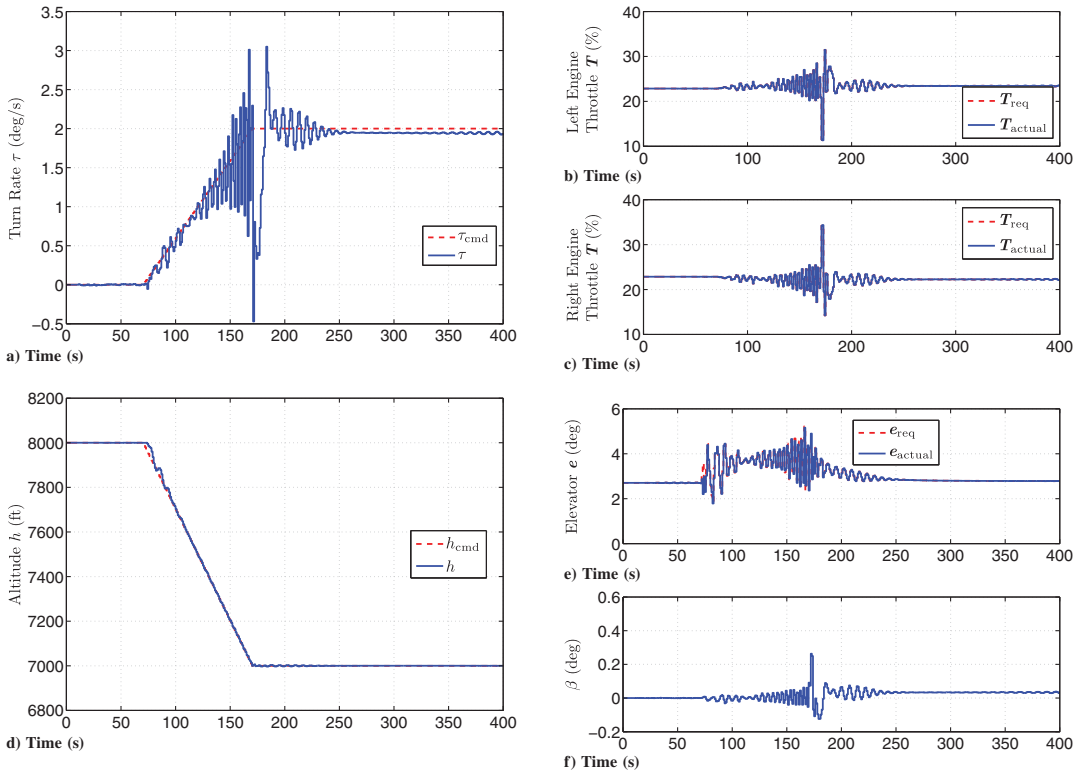

Fig. 31 Example E.1.2: Control architecture A_6 for helical flight around a vertical axis with trapezoidal turn-rate command and trapezoidal altitude command. This example is similar to Example E.1.1, except that the incremental altitude command is a trapezoid.

Figure 32a shows that, after $t = 50$ s, the pitot-tube measurement $V_{AC,meas}$ is fixed at 100.6 kt. After $t = 70$ s, Fig. 32 shows that the adaptive controller initially uses full throttle when $V_{AC,meas} - V_{AC,cmd}$ is increasing and then converges to a constant throttle setting at $t = 160$ due to the lack of effect on the error signal $V_{AC,meas} - V_{AC,cmd}$. After $t = 170$ s, $\delta V_{AC,cmd}$ is constant, and the adaptive controller trims the aircraft at

$V_{AC} = 122.98$ kt, $h = 8010$ ft, $\tau = 0.004$ deg/s, and $\beta = 0.006$ deg with the command-following errors $V_{AC} - V_{AC,cmd} = 17.39$ kt, $h - h_{cmd} = 10$ ft, $\tau - \tau_{cmd} = 0.004$ deg/s, and $\beta - \beta_{cmd} = 0.006$ deg at $t = 250$ s. ■

B. Scenario F.2: Biased Sensor

Example F.2.1: For horizontal straight-line flight with biased pitot-tube measurement using control architecture A_2, the incremental commands are given by

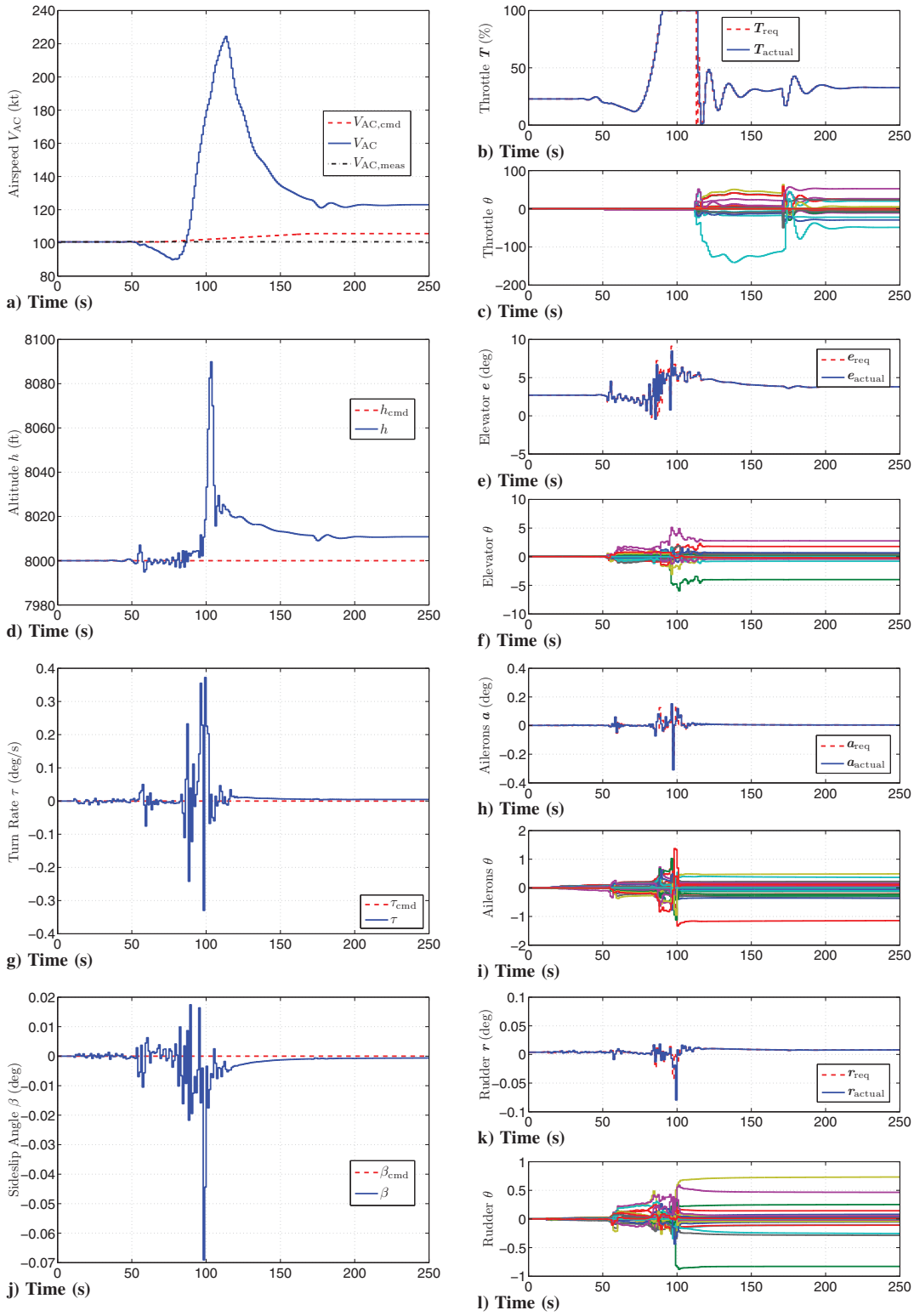

Fig. 32 Example F.1.1: Control architecture A_2 for horizontal straight-line flight with stuck pitot-tube measurement after $t = 50$ s. After $t = 70$ s, the adaptive controller initially uses full throttle when $V_{AC,meas} - V_{AC,cmd}$ is increasing and then converges to a constant throttle setting at $t = 160$ due to the lack of effect on the error signal $V_{AC,meas} - V_{AC,cmd}$.

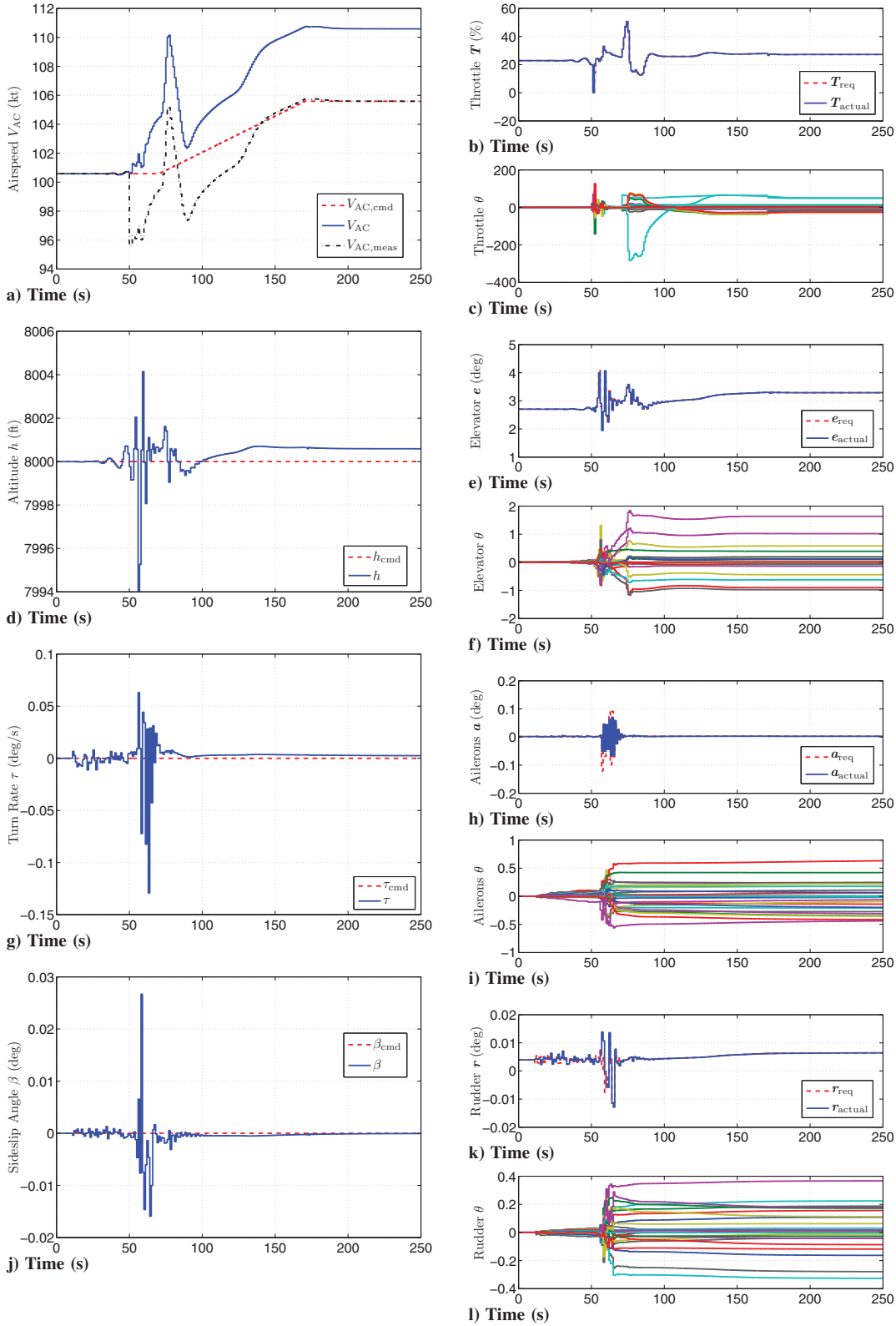

Fig. 33 Example F.2.1: Control architecture A_2 for horizontal straight-line flight with biased pitot-tube measurement. After $t = 50$ s, the pitot-tube measurement $V_{AC,meas}$ has a bias of 5 kt and the aircraft follows the commanded airspeed with 5 kt bias.

$$\delta V_{AC,cmd}(k) = \begin{cases} 0, & k < 700, \\ \min\{5, 0.005(k-700)\}\ \text{kt}, & k \geq 700, \end{cases}$$
$$\delta h_{AC,cmd}(k) = 0\ \text{ft}, \delta\tau_{cmd}(k) = 0\ \text{deg/s}, \delta\beta_{cmd}(k) = 0\ \text{deg} \tag{75}$$

where the trapezoidal airspeed command starts at $t = 70$ s.

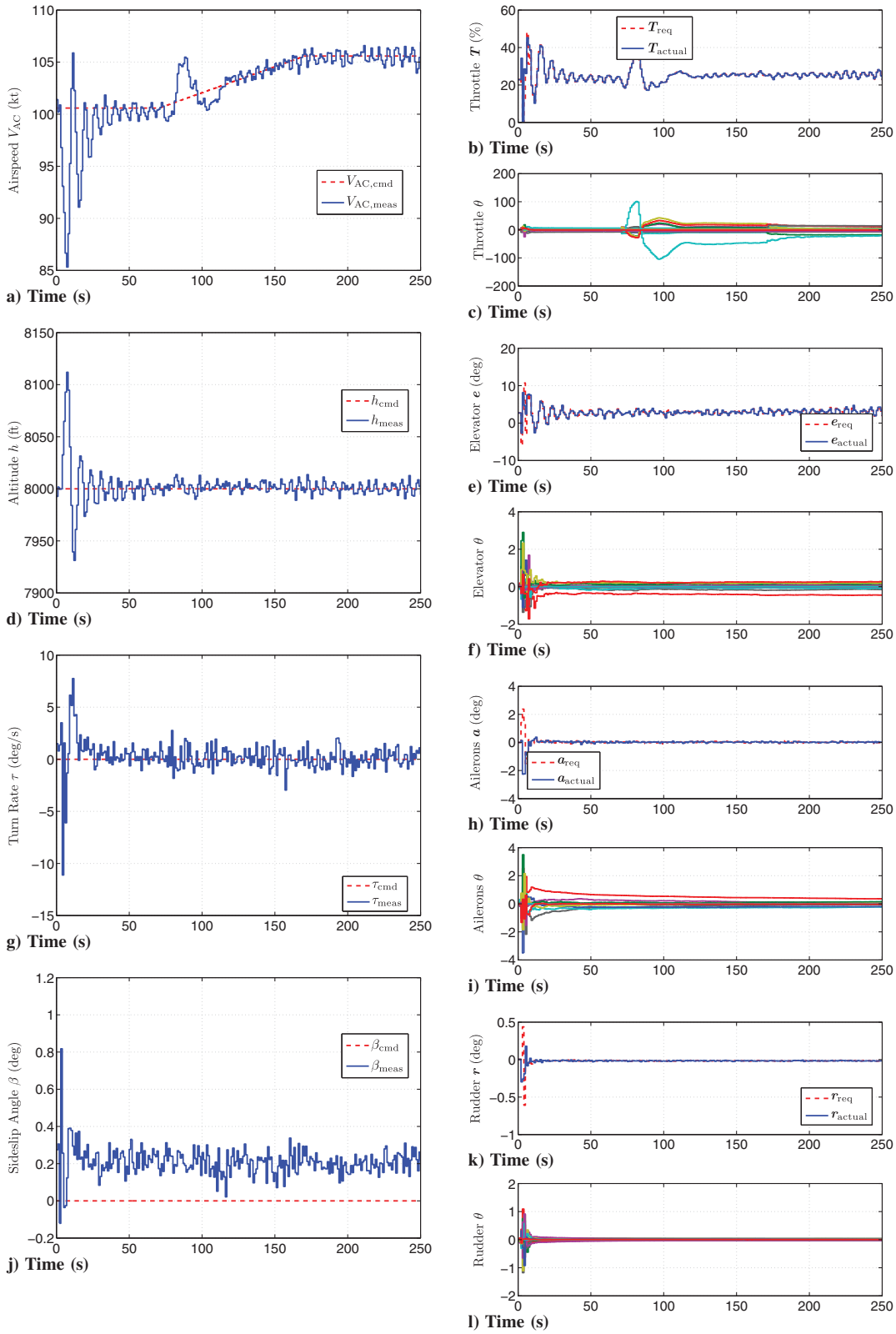

Fig. 34 Example F.3.1: Control architecture A_2 for horizontal straight-line flight with broadband sensor noise. The adaptive controller is able to follow the commands in the presence of noisy measurements.

Figure 33a shows that, after $t = 50$ s, the pitot-tube measurement $V_{AC,meas}$ has a bias of 5 kt. Figure 33 shows that the aircraft follows the commanded airspeed with a bias of 5 kt. At $t = 250$ s, the command-following errors for airspeed, altitude, turn rate, and sideslip angle are 5 kt, 0.5 ft, 0.002 deg/s, and 0 deg, respectively. ■

C. Scenario F.3: Broadband Sensor Noise

Example F.3.1: For horizontal straight-line flight with broadband sensor noise using control architecture A_2, the incremental commands are given by

Table 3 Stroke and rate limits for severely saturated throttle, elevator, ailerons, and rudder[a]

	Stroke limits	Rate limits
Throttle $\boldsymbol{T}$	[20,40]%	$[-4, 4]$%/s
Elevator $\boldsymbol{e}$	$[-4, 4]$ deg	$[-1, 1]$ deg/s
Ailerons $\boldsymbol{a}$	$[-4, 4]$ deg	$[-1, 1]$ deg/s
Rudder $\boldsymbol{r}$	$[-4, 4]$ deg	$[-1, 1]$ deg/s

[a]These values determine the actual actuator settings.

$$\delta V_{\mathrm{AC,cmd}}(k) = \begin{cases} 0, & k < 700, \\ \min\{5, 0.005(k-700)\}\ \mathrm{kt}, & k \geq 700, \end{cases}$$
$$\delta h_{\mathrm{AC,cmd}}(k) = 0\ \mathrm{ft}, \delta\tau_{\mathrm{cmd}}(k) = 0\ \ \mathrm{deg/s}, \delta\beta_{\mathrm{cmd}}(k) = 0\ \ \mathrm{deg} \tag{76}$$

where the trapezoidal airspeed command starts at $t = 70$ s.

To emulate sensor noise, we use a subroutine of the GTM that models the sensor noise using 1) white Gaussian random variable, 2) sensor bias, and 3) scaling factor [23,29]. Figure 34 shows that the adaptive controller is able to follow the commands in the presence of noisy measurements. From $t = 170$ to 250 s, the root mean square of the errors in command following for airspeed, altitude, turn rate, and sideslip angle are 0.58 kt, 5.7 ft, 0.81 deg/s, and 0.209 deg, respectively. ■

XI. Case G: Control with Unknown Actuator Failure

We investigate several scenarios involving actuator failure, where the actuator failure is unknown to RCAC. Because the actuator failure is unknown, we use the requested actuator setting increment in place of the actual actuator setting increment in the feedback vector ϕ for all control architectures in this section.

A. Scenario G.1: Severe Stroke and Rate Limits

We investigate a scenario where the actuator stroke and rate limits become severely saturated during flight due to a failure, for which the onset and limits are unknown to the adaptive controller. Table 3 gives the stroke and rate limits of the saturated actuators, which are severe compared with the nominal limits in Table 2.

Example G.1.1: For ascending helical flight around a vertical axis, followed by horizontal circular flight with severe throttle, elevator, aileron, and rudder stroke and rate limits using control architecture A_1, the incremental commands are given by

$$\delta\tau_{\mathrm{cmd}}(k) = \begin{cases} 0, & k < 700, \\ \min\{5, 0.005(k-700)\}\ \ \mathrm{deg/s}, & k \geq 700, \end{cases}$$
$$\delta\gamma_{\mathrm{cmd}}(k) = \begin{cases} 0, & k < 700, \\ \min\{4, 0.004(k-700)\}\ \ \mathrm{deg}, & 700 \leq k < 4000, \\ \max\{0, 4 - 0.004(k-4000)\}\ \ \mathrm{deg}, & k \geq 4000, \end{cases}$$
$$\delta V_{\mathrm{AC,cmd}}(k) = 0\ \mathrm{kt}, \delta\beta_{\mathrm{cmd}}(k) = 0\ \ \mathrm{deg} \tag{77}$$

where the trapezoidal flight-path-angle and turn-rate commands start at $t = 70$ s.

Figure 35b shows that the throttle reaches the upper stroke limit of 40% during helical flight. Because of the stroke-saturated throttle, the airspeed drops to a maximum of 69 kt at $t = 400$ s, as shown in Fig. 35a. Figure 35 shows that the maximum command-following errors for airspeed, flight-path angle, turn rate, and sideslip angle are 31 kt, 11 deg, 4.4 deg/s, and 0.5 deg, respectively, and the aircraft maintains constant turn rate and airspeed in horizontal flight after 500 s. ■

Example G.1.2: For horizontal straight-line flight with severe throttle and elevator stroke and rate limits using control architecture A_2, the incremental commands are given by

$$\delta V_{\mathrm{AC,cmd}}(k) = \begin{cases} 0, & k < 700, \\ \min\{5, 0.005(k-700)\}\ \mathrm{kt}, & k \geq 700, \end{cases}$$
$$\delta\tau_{\mathrm{AC,cmd}}(k) = 0\ \ \mathrm{deg/s}, \delta h_{\mathrm{cmd}}(k) = 0\ \mathrm{ft}, \delta\beta_{\mathrm{cmd}}(k) = 0\ \ \mathrm{deg} \tag{78}$$

where the trapezoidal airspeed command starts at $t = 70$ s.

Figures 36b and 36e show that the throttle and elevator reach the stroke limits. Figure 36 shows that the maximum command-following errors for airspeed, altitude, turn rate, and sideslip angle are 20 kt, 250 ft, 0.3 deg/s, and 0.02 deg, respectively, and the aircraft maintains constant airspeed and altitude in horizontal straight-line flight after 300 s. ■

Example G.1.3: For reference flight-trajectory following with severe throttle, aileron, and rudder stroke and rate limits in the presence of wind shear using control architecture A_5 (note that no ADS measurements are used in A_5), the reference trajectory is composed of ascending, right-turn, and straight-line maneuvers. The incremental command δh_{cmd} is

$$\delta h_{\mathrm{cmd}}(k) = \begin{cases} 0, & k < 700, \\ \min\{500, 0.5(k-700)\}\ \mathrm{ft}, & k \geq 700 \end{cases} \tag{79}$$

where the trapezoidal altitude command starts at $t = 70$ s.

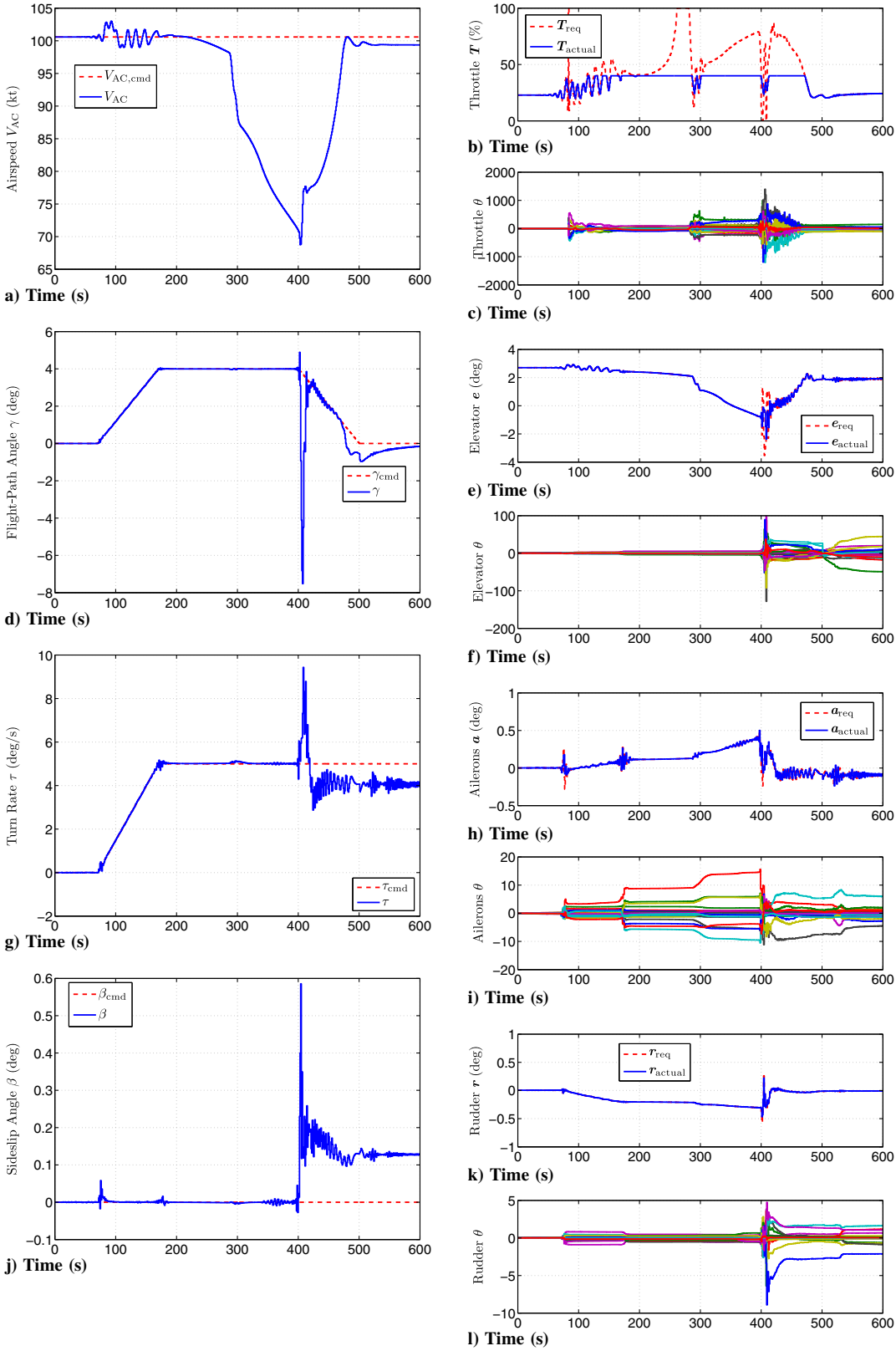

Fig. 35 Example G.1.1: Control architecture A_1 for ascending helical flight around a vertical axis, followed by horizontal circular flight with severe throttle, elevator, aileron, and rudder stroke and rate limits. The throttle reaches the upper stroke limit of 40% during helical flight and thus the airspeed drops to a maximum of 69 kt at $t = 400$ s.

Figure 37 shows that the maximum command-following errors for $e_{\parallel}$, h, $e_{\perp}$, and η are 1082, 61, and 287 ft, and 4.5 deg, respectively, whereas the GTM follows the reference flight trajectory. The maximum angle of attack, sideslip angle, and airspeed (not shown in Fig. 37) are 10 deg, 0.18 deg, and 123 kt, respectively. ■

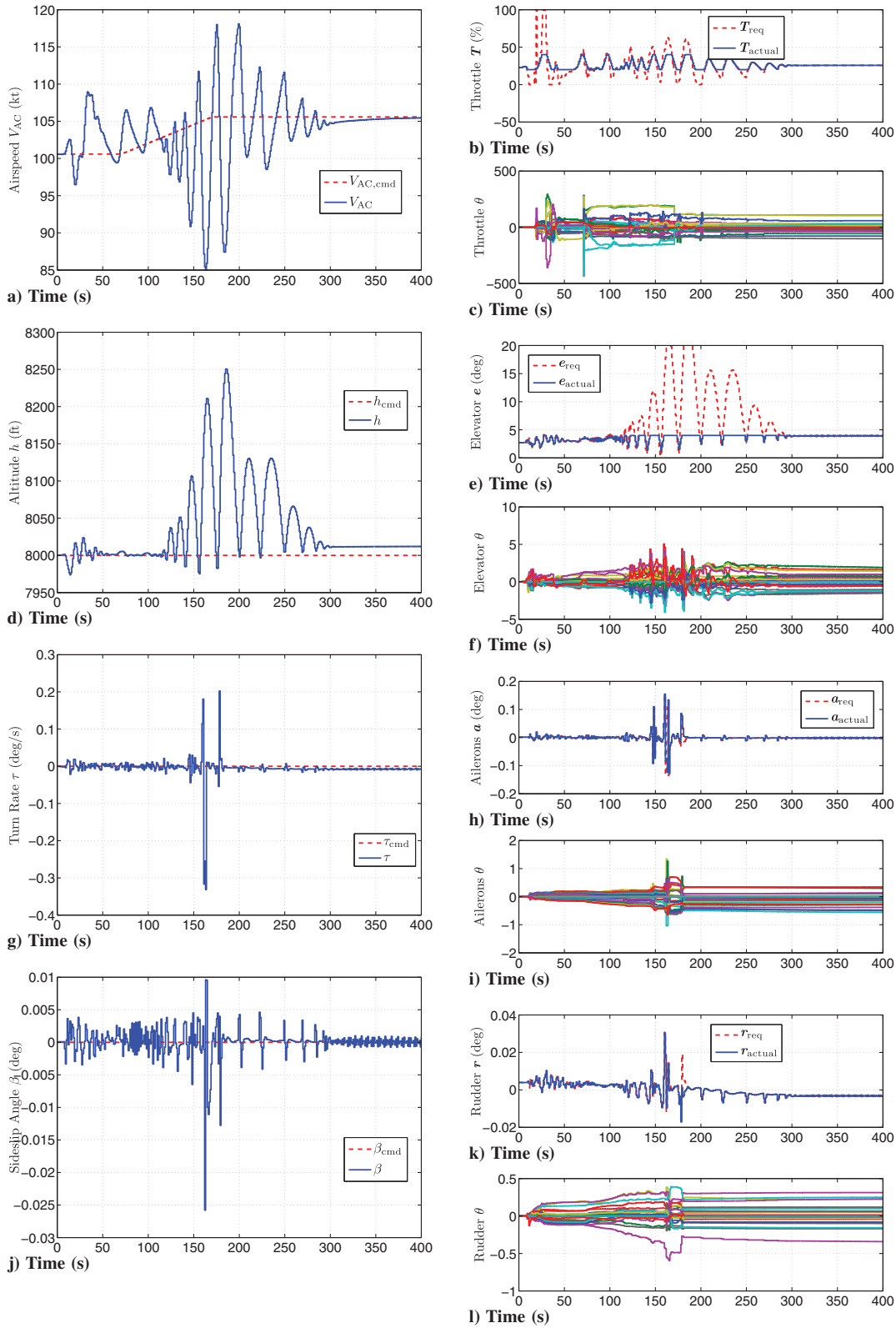

Fig. 36 Example G.1.2: Control architecture A_2 for horizontal straight-line flight with severe throttle and elevator stroke and rate limits. The aircraft maintains constant airspeed and altitude in horizontal straight-line flight after 300 s.

B. Scenario G.2: Dead Zone

Example G.2.1: For horizontal circular flight with aileron-deflection dead zone of ± 0.1 deg using control architecture A_2, the aileron deflection is zero in the dead zone and follows the requested actuator settings outside the dead zone. The incremental commands are given by

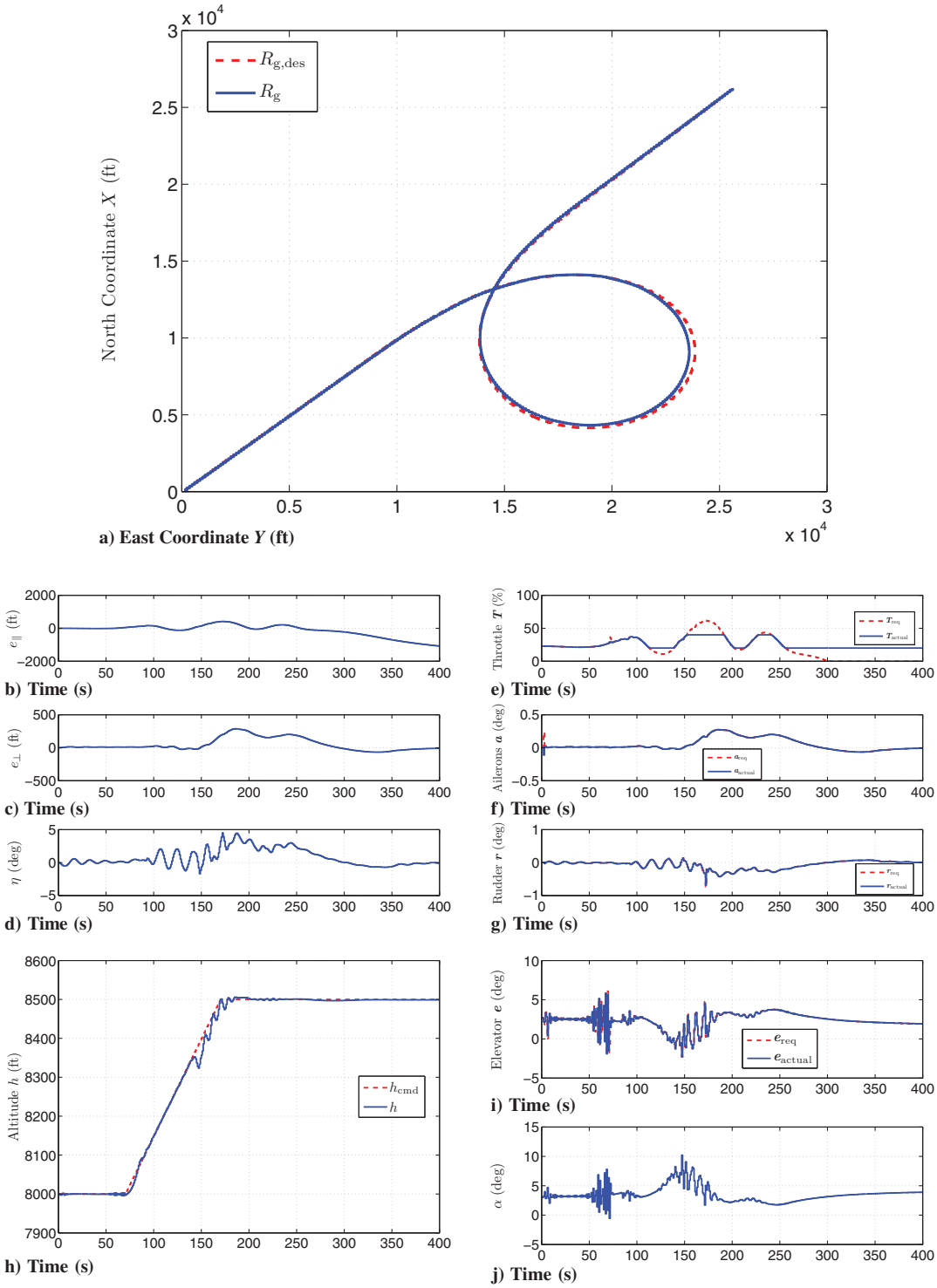

Fig. 37 Example G.1.3: Control architecture A_5 for reference flight-trajectory following with severe throttle, aileron, and rudder stroke and rate stroke and rate limits in the presence of wind shear. The GTM follows the reference flight trajectory.

$$\delta\tau_{\mathrm{AC,cmd}}(k) = \begin{cases} 0, & k < 700, \\ \min\{3, 0.003(k-700)\} \ \mathrm{deg/s}, & k \geq 700, \end{cases}$$
$$\delta V_{\mathrm{AC,cmd}}(k) = 0 \ \mathrm{kt}, \delta h_{\mathrm{cmd}}(k) = 0 \ \mathrm{ft}, \delta\beta_{\mathrm{cmd}}(k) = 0 \ \mathrm{deg} \tag{80}$$

where the trapezoidal turn-rate command starts at $t = 70$ s.

Figure 38h shows that the aileron deflection is zero in the dead zone. Figure 38 shows that the maximum command-following errors for airspeed, altitude, turn rate, and sideslip angle are 3.7 kt, 26.9 ft, 0.5 deg/s, and 0.1 deg, respectively, and the aircraft maintains constant turn rate and altitude in horizontal straight-line flight after 170 s. ■

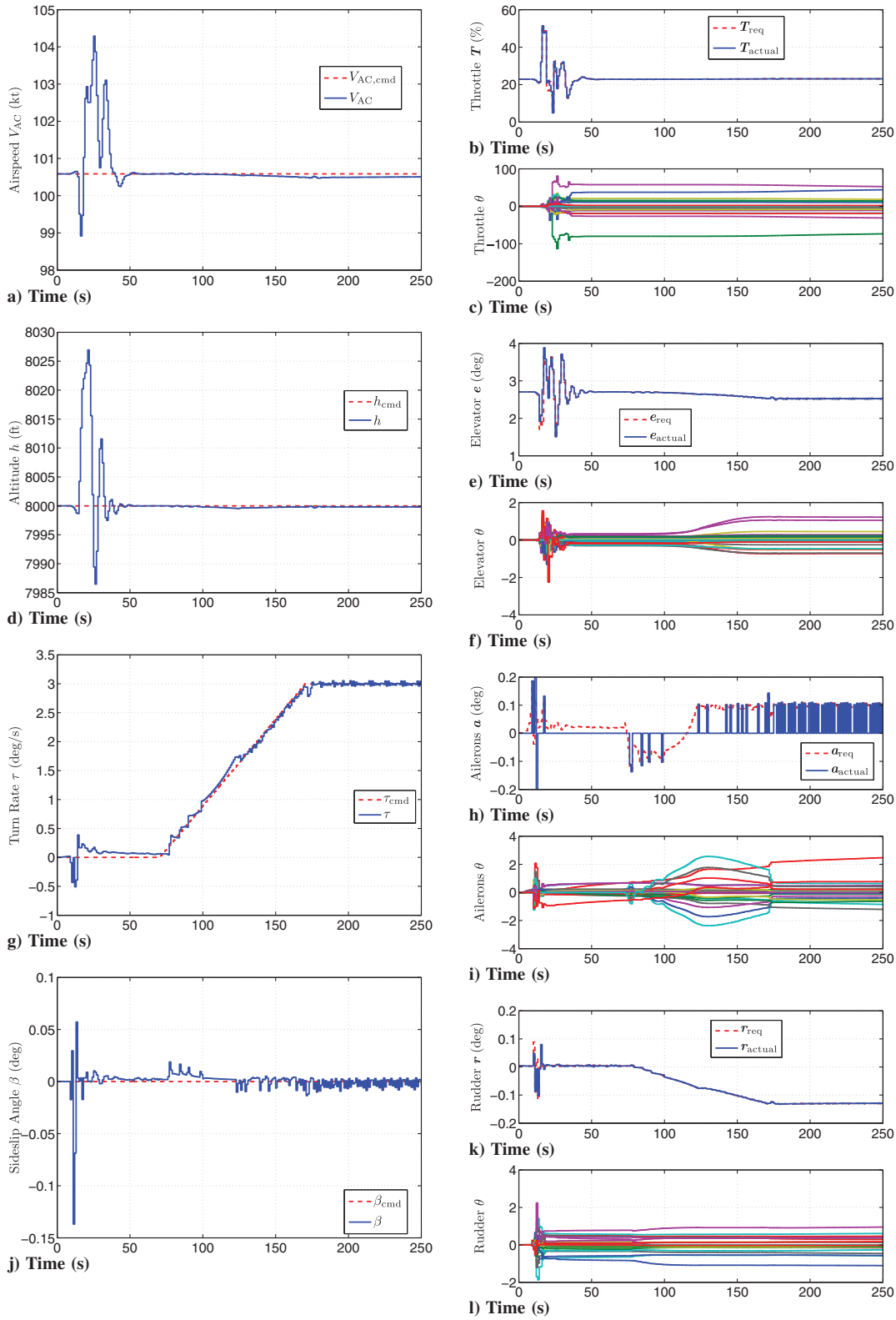

Fig. 38 Example G.2.1: Control architecture A_2 for horizontal circular flight with aileron-deflection dead zone of ± 0.1 deg. The aircraft maintains constant turn-rate and altitude in horizontal circular flight after 170 s.

C. Scenario G.3: Jam

Example G.3.1: For horizontal circular flight, followed by straight-line flight with a rudder jam at $t = 200$ s using control architecture A_1, the incremental commands are given by

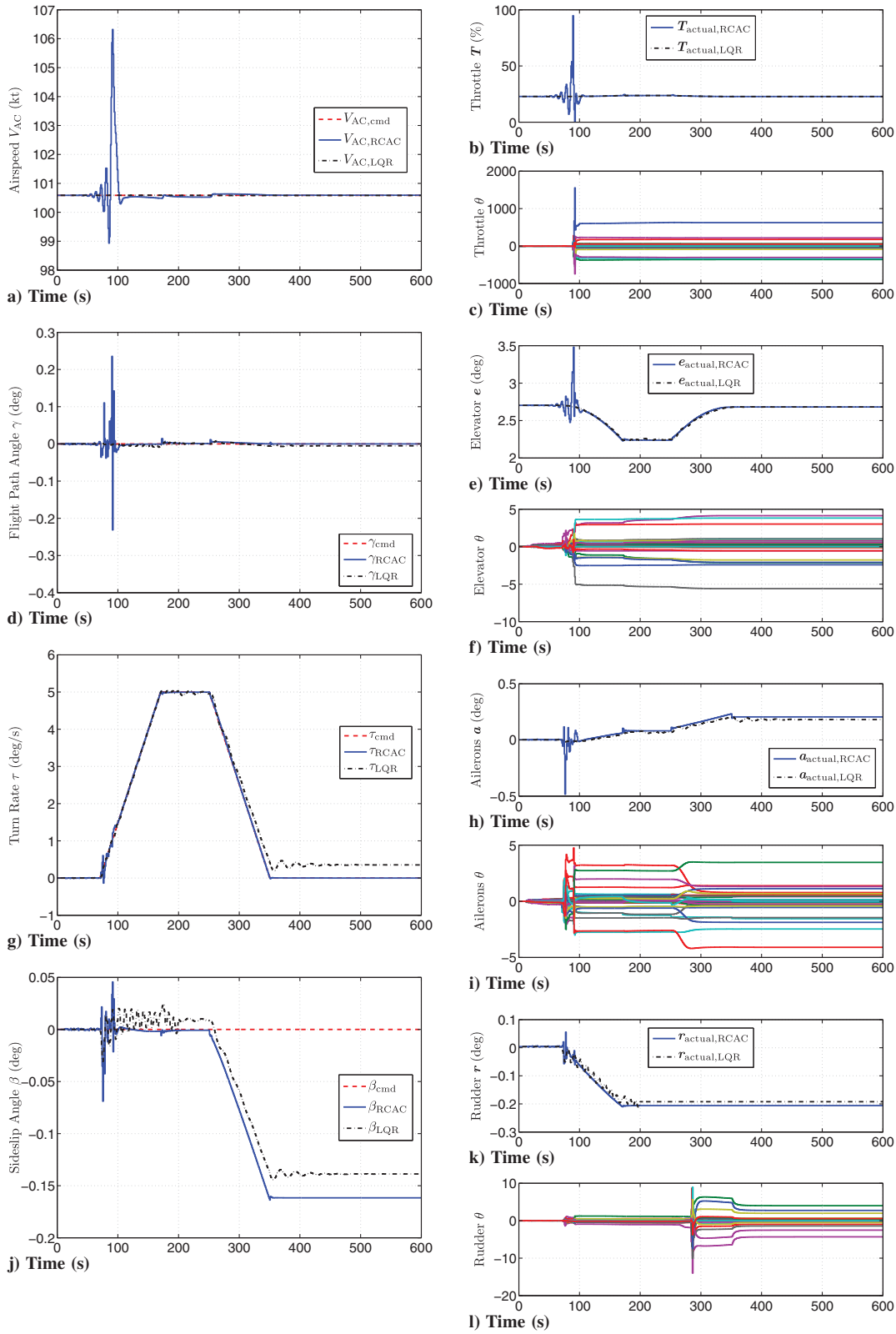

Fig. 39 Example G.3.1: Control architecture A_1 for horizontal circular flight, followed by straight-line flight with a rudder jam at $t = 200$ s. Comparing τ_{RCAC} and τ_{LQR}, RCAC trims the aircraft in the commanded straight-line flight after $t = 350$ s, whereas LQR trims the aircraft in horizontal circular flight after $t = 350$ s due to the 0.35 deg/s command-following error in the turn rate.

$$\delta\tau_{cmd}(k) = \begin{cases} 0, & k < 700, \\ \min\{5, 0.005(k-700)\}\ \text{deg/s}, & 700 \le k < 4000, \\ \max\{0, 5-0.005(k-2500)\}\ \text{deg/s}, & k \ge 2500, \end{cases}$$

$$\delta\gamma_{cmd}(k) = 0\ \text{deg}, \delta V_{AC,cmd}(k) = 0\ \text{kt}, \delta\beta_{cmd}(k) = 0\ \text{deg} \tag{81}$$

where the trapezoidal turn-rate command starts at $t = 70$ s and ends at $t = 350$ s.

Figure 39k shows that the rudder jam occurs at $t = 200$ s. Because of the jammed rudder, the aileron controller coefficients adapt in response to the failure to follow the turn-rate command, as shown in Fig. 39i. The RCAC command-following errors at $t = 600$ s for airspeed, flight-path angle, turn rate, and sideslip angle are 0.003 kt, 10^{-5} deg, 4×10^{-5} deg/s, and 0.16 deg, respectively, whereas the LQR command-following errors at $t = 600$ s for airspeed, flight-path angle, turn rate, and sideslip angle are 0.005 kt, 0.005 deg, 0.35 deg/s, and 0.013 deg, respectively. Comparing τ_{RCAC} and τ_{LQR} in Fig. 39g, RCAC trims the aircraft in the commanded straight-line flight after $t = 350$ s, whereas LQR trims the aircraft in horizontal circular flight after $t = 350$ s due to the 0.35 deg/s command-following error in the turn rate. ■

XII. Conclusions

In this paper, the performance and robustness of retrospective cost adaptive control for an aircraft under conditions of uncertainty and failure were investigated. In this regard, the ability of RCAC to control the NASA GTM with minimal modeling information and limited measurements were explored. In addition, RCAC does not require an inner-/outer-loop controller structure and it is used to transition between flight regimes without the benefit of gain scheduling.

The GTM is a realistic high-fidelity aircraft model that includes an aerodynamic database and interface to facilitate feedback control from realistic sensors to thrust and control surfaces, including stroke and rate limits. To apply RCAC to the GTM, a collection of architectures were defined that assign specified performance (error) variables to control inputs. Each architecture thus has a decentralized controller structure with respect to adaptation. In addition, measurement signals may be shared among the channels to account for channel coupling.

To determine trim states, the ability of RCAC to reach trim flight from an initial nontrim flight condition was demonstrated and then the attained trim was compared with the trim computed from the trimgtm subroutine of the GTM. Starting the adaptive controller from a trim flight, warmup strategies to improve the transient response of the aircraft during command following were then presented. Examples showed that the transient error was smallest with actuator noise warmup. Then, the ability of RCAC to follow trim commands involving straight-line, circular, and helical flight was shown. The robustness of RCAC tuning to the initial trim flight was also investigated.

For the cases of uncertainty and failure, a series of scenarios that examine the performance of RCAC in various flight regimes was considered. Cases of unknown time-varying aircraft dynamics (such as mass variation and icing) and unknown flight envelope (including the possibility of conflicting commands) were presented, and it was shown that RCAC was able to trim the aircraft under these scenarios. The ability of RCAC to actively stabilize a trim flight that is not asymptotically stable was also demonstrated. RCAC was then used to explore the flight envelope by employing a sequence of trim commands.

For the cases of known sensing and actuation constraints, scenarios of controlling the GTM with total air data system failure were presented, as well as maneuvers with differential thrust in the presence of stuck aileron and rudder. In both cases, RCAC was able to follow the commands.

For the cases of unknown sensor failure, scenarios were presented involving a stuck and biased pitot tube. For the case of a stuck pitot-tube measurement, RCAC converged to a constant throttle setting corresponding to trim flight. For the case of biased pitot-tube measurement, the GTM followed the commanded airspeed with respect to the biased measurement.

For the case of unknown actuator failure, scenarios were presented involving severe stroke and rate limits, dead zone, and jam. RCAC was able to follow commands under these scenarios at the expense of degradation in the transient response and steady-state command-following errors.

RCAC was originally developed for linear plants and was applied to the GTM as a case study of a realistic nonlinear model of a flight vehicle. The examples considered in this paper show that RCAC is able to handle the nonlinearities arising in the GTM, such as in its aerodynamic database, actuator dynamics, and kinematics. The ability of RCAC to handle input nonlinearities is not unexpected due to the application of RCAC to Hammerstein systems in [35]. The most severe cases in terms of unknown nonlinearities are the cases of unknown sensor and actuator failure. Although unknown actuator failure might be viewed as an unknown disturbance, RCAC has no knowledge that the stuck control surface whose deflection it is requesting is not responding. Likewise, RCAC has no way to ascertain that a sensor signal is stuck. In both cases, however, RCAC was able to bring the aircraft to trim flight.

Each example in this paper was confined to a single architecture. Although these examples involve transitions between different trims, the problem of transitioning between different architectures was not considered. This remains an open research problem. Because RCAC does not depend on full-state measurement, it may find application to aircraft that require output feedback control, such as aircraft with sensor and actuator dynamics as well as flexible modes. Each architecture is based on decentralized adaptation, although the separate controllers can take advantage of coupling signals, and thus the controller is centralized. For the case of a scalar performance variable and scalar input signal, as in all architectures used in this paper, limited numerical testing suffices to determine suitable tuning parameters. A more general formulation would be to use fully centralized adaptation, which may provide improved transient performance. However, techniques for choosing the tuning parameters for this case remain to be developed. Given the ease of application of RCAC, this method may provide a path to fast prototyping of control laws for unconventional aircraft for which a simulation model exists. As a first step, RCAC can be used to find trim conditions without derivation and analysis of the linearized flight equations. Finally, RCAC remains to be flight tested. This is a future objective.

Appendix A: Tuning Parameters

The feedback vector ϕ_i, filter G_f, and tuning parameters n_c, R_z, R_u, and R_θ for control architectures A_2, A_3, A_4, A_5, and A_6 are as follows. These tunings are used consistently in all examples.

A1. Control Architecture A_2

The feedback vectors ϕ_1, ϕ_3, and ϕ_4 for A_2 are the same as in A_1 given by Eq. (42), whereas ϕ_2 is

$$\begin{aligned}\phi_2(k) \triangleq [&\delta \boldsymbol{e}_{\mathrm{actual}}(k-1) \ \cdots \ \delta \boldsymbol{e}_{\mathrm{actual}}(k-n_c) \quad \delta h_{\mathrm{cmd}}(k-1) \ \cdots \ \delta h_{\mathrm{cmd}}(k-n_c) \\ &z_2(k-1) \ \cdots \ z_2(k-n_c) \qquad \delta V_{\mathrm{AC}}(k-1) \ \cdots \ \delta V_{\mathrm{AC}}(k-n_c)]^{\mathrm{T}}\end{aligned} \tag{A1}$$

where $z_2(k) \triangleq h(k) - h_{\mathrm{cmd}}(k)$. We use

$$G_{\mathrm{f}}(z) = \mathrm{diag}(1/z^4, -1/z^4, -1/z^4, 1/z^4)$$

and the tuning parameters

$$n_c = 8, R_z = I_4, R_u = \mathrm{diag}(0.5, 10, 10, 10), R_\theta = \mathrm{diag}(10^{-4} I_{4n_c}, I_{4n_c}, 10^{-2} I_{4n_c}, 10^{-3} I_{4n_c})$$

A2. Control Architecture A_3

The feedback vectors ϕ_1, ϕ_2, ϕ_3, and ϕ_4 for A_3 are the same as in A_1 given by Eq. (42), whereas ϕ_5 is

$$\begin{aligned}\phi_5(k) \triangleq [&\delta \boldsymbol{e}_{\text{actual}}(k-1) \ \cdots \ \delta \boldsymbol{e}_{\text{actual}}(k-n_c) \quad \delta\alpha_{\text{cmd}}(k-1) \ \cdots \ \delta\alpha_{\text{cmd}}(k-n_c) \\ &z_5(k-1) \ \cdots \ z_5(k-n_c) \quad \delta V_{\text{AC}}(k-1) \ \cdots \ \delta V_{\text{AC}}(k-n_c)]^{\text{T}}\end{aligned} \tag{A2}$$

where the additional performance variable is $z_5 \triangleq \alpha(k) - \alpha_{\text{cmd}}(k)$. We use

$$G_{\text{f}}(z) = \text{diag}(1/z^4, -1/z^4, -1/z^4, 1/z^4, -1/z^4)$$

and the tuning parameters

$$n_c = 8, R_z = I, R_u = \text{diag}(0.5, 1, 10, 10, 1), R_\theta = \text{diag}(10^{-4} I_{4n_c}, 10^{-3} I_{4n_c}, I_{4n_c}, 10^{-3} I_{4n_c}, I_{4n_c})$$

A3. Control Architecture A_4

The feedback vectors ϕ_1, ϕ_2, ϕ_3 for A_4 are the same as in A_1 given by Eq. (42). We use

$$G_{\text{f}}(z) = \text{diag}(-1/z^4, -1/z^4, -1/z^4)$$

and the tuning parameters

$$n_c = 8, R_z = I_3 R_u = \text{diag}(0.5, 1, 10), R_\theta = \text{diag}(10^{-6} I_{4n_c}, 10^{-4} I_{4n_c}, 10^{-4} I_{4n_c})$$

A4. Control Architecture A_5

The feedback vectors ϕ_1, ϕ_2, ϕ_3, and ϕ_4 are given by

$$\begin{aligned}\phi_1(k) \triangleq [&\delta \boldsymbol{T}_{\text{actual}}(k-1) \ \cdots \ \delta \boldsymbol{T}_{\text{actual}}(k-n_c) \quad \delta V_g(k-1) \ \cdots \ \delta V_g(k-n_c) \\ &z_1(k-1) \ \cdots \ z_1(k-n_c) \quad \delta\gamma(k-1) \ \cdots \ \delta\gamma(k-n_c)]^{\text{T}}, \\ \phi_2(k) \triangleq [&\delta \boldsymbol{e}_{\text{actual}}(k-1) \ \cdots \ \delta \boldsymbol{e}_{\text{actual}}(k-n_c) \quad \delta h_{\text{cmd}}(k-1) \ \cdots \ \delta h_{\text{cmd}}(k-n_c) \\ &z_2(k-1) \ \cdots \ z_2(k-n_c) \quad \delta V_g(k-1) \ \cdots \ \delta V_g(k-n_c)]^{\text{T}}, \\ \phi_3(k) \triangleq [&\delta \boldsymbol{a}_{\text{actual}}(k-1) \ \cdots \ \delta \boldsymbol{a}_{\text{actual}}(k-n_c) \quad \delta\tau_{\text{cmd}}(k-1) \ \cdots \ \delta\tau_{\text{cmd}}(k-n_c) \\ &z_3(k-1) \ \cdots \ z_3(k-n_c) \quad \delta\tau(k-1) \ \cdots \ \delta\tau(k-n_c)]^{\text{T}}, \\ \phi_4(k) \triangleq [&\delta \boldsymbol{r}_{\text{actual}}(k-1) \ \cdots \ \delta \boldsymbol{r}_{\text{actual}}(k-n_c) \quad \delta\tau_{\text{cmd}}(k-1) \ \cdots \ \delta\tau_{\text{cmd}}(k-n_c) \\ &z_4(k-1) \ \cdots \ z_4(k-n_c) \quad \delta\tau(k-1) \ \cdots \ \delta\tau(k-n_c)]^{\text{T}}\end{aligned} \tag{A3}$$

where the performance variable z is given by the error signals

$$z(k) \triangleq \begin{bmatrix} e_{\parallel}(k) \\ h(k) - h_{\text{cmd}}(k) \\ e_{\perp}(k) \\ \eta(k) \end{bmatrix} \tag{A4}$$

We use

$$G_{\text{f}}(z) = \text{diag}(-1/z^4, -1/z^4, -1/z^4, 1/z^4)$$

and the tuning parameters

$$n_c = 8, R_z = I_4, R_u = \text{diag}(10, 10, 1000, 10), R_\theta = \text{diag}(10^{-4} I_{4n_c}, I_{4n_c}, 10^{-2} I_{4n_c}, 10^{-4} I_{4n_c})$$

A5. Control Architecture A_6

The feedback vectors ϕ_1 and ϕ_2 are given by

$$\begin{aligned}\phi_1(k) \triangleq [&\delta \boldsymbol{T}_{\text{actual}}(k-1) \ \cdots \ \delta \boldsymbol{T}_{\text{actual}}(k-n_c) \quad \delta\tau_{\text{cmd}}(k-1) \ \cdots \ \delta\tau_{\text{cmd}}(k-n_c) \\ &z_1(k-1) \ \cdots \ z_1(k-n_c) \quad \delta\beta(k-1) \ \cdots \ \delta\beta(k-n_c)]^{\text{T}}, \\ \phi_2(k) \triangleq [&\delta \boldsymbol{e}_{\text{actual}}(k-1) \ \cdots \ \delta \boldsymbol{e}_{\text{actual}}(k-n_c) \quad \delta h_{\text{cmd}}(k-1) \ \cdots \ \delta h_{\text{cmd}}(k-n_c) \\ &z_2(k-1) \ \cdots \ z_2(k-n_c) \quad \delta V_{\text{AC}}(k-1) \ \cdots \ \delta V_{\text{AC}}(k-n_c)]^{\text{T}}\end{aligned} \tag{A5}$$

where the performance variable z is given by the error signals

$$z(k) \triangleq \begin{bmatrix} \tau(k) - \tau_{\text{cmd}}(k) \\ h(k) - h_{\text{cmd}}(k) \end{bmatrix} \tag{A6}$$

We use

$$G_{\mathrm{f}}(z) = \mathrm{diag}(1/z^4, -1/z^4)$$

and the tuning parameters

$$n_c = 8, R_z = I_2, R_u = \mathrm{diag}(0.1, 10), R_\theta = \mathrm{diag}(I_{4n_c}, 10^{-2} I_{4n_c})$$

Appendix B: RCAC Algorithm

B1. Plant Model

Consider the MIMO discrete-time system

$$x(k+1) = Ax(k) + Bu(k) + D_1 w(k) \tag{B1}$$

$$y(k) = Cx(k) + D_2 w(k) \tag{B2}$$

$$z(k) = E_1 x(k) + E_0 w(k) \tag{B3}$$

where $x(k) \in \mathbb{R}^{l_x}$ is the state, $y(k) \in \mathbb{R}^{l_y}$ is the measurement, $u(k) \in \mathbb{R}^{l_u}$ is the input, $w(k) \in \mathbb{R}^{l_w}$ is the exogenous signal, and $z(k) \in \mathbb{R}^{l_z}$ is the performance variable. The goal is to develop an adaptive output feedback controller that minimizes z in the presence of the exogenous signal w with limited modeling information about Eqs. (B1–B3). The components of w can represent either command signals to be followed, external disturbances to be rejected, or both, depending on the choice of D_1 and E_0. This formulation defines the signals that play a role in RCAC. However, no assumptions are made concerning the state-space realization because RCAC requires input–output model information rather than details of the state-space realization.

B2. Controller

Define the dynamic compensator

$$u(k) = \sum_{i=1}^{n_c} P_i(k) u(k-i) + \sum_{i=k_0}^{n_c} Q_i(k) \xi(k-i) \tag{B4}$$

where $P_i(k) \in \mathbb{R}^{l_u \times l_u}$, $Q_i(k) \in \mathbb{R}^{l_u \times l_\xi}$ are the controller-coefficient matrices, $k_0 \geq 0$, and $\xi(k) \in \mathbb{R}^{l_\xi}$ consists of components of y, z, and w. We rewrite Eq. (B4) as

$$u(k) = \phi(k)\theta(k) \tag{B5}$$

where the regressor matrix $\phi(k)$ is defined by

$$\phi(k) \triangleq \begin{bmatrix} u(k-1) \\ \vdots \\ u(k-n_c) \\ \xi(k-k_0) \\ \vdots \\ \xi(k-n_c) \end{bmatrix}^{\mathrm{T}} \otimes I_{l_u} \in \mathbb{R}^{l_u \times l_\theta} \tag{B6}$$

and

$$\theta(k) \triangleq \mathrm{vec}[\, P_1(k) \ \cdots \ P_{n_c}(k) Q_{k_0}(k) \ \cdots \ Q_{n_c}(k) \,] \in \mathbb{R}^{l_\theta} \tag{B7}$$

where $l_\theta \triangleq l_u^2 n_c + l_u l_\xi (n_c + 1 - k_0)$, "$\otimes$" is the Kronecker product, and "vec" is the column-stacking operator. Note that $k_0 = 0$ yields an exactly proper controller, whereas $k_0 \geq 1$ yields a strictly proper controller.

B3. Retrospective Performance Variable

We define the retrospective control as

$$\hat{u}(k) = \phi(k)\hat{\theta} \tag{B8}$$

and the corresponding retrospective performance variable as

$$\hat{z}(k) \triangleq z(k) + \phi_{\mathrm{f}}(k)\hat{\theta} - u_f(k) \tag{B9}$$

where $\hat{\theta} \in \mathbb{R}^{l_\theta}$ is determined by optimization next, and $\phi_{\mathrm{f}}(k) \in \mathbb{R}^{l_z \times l_\theta}$ and $u_f(k) \in \mathbb{R}^{l_z}$ are filtered versions of $\phi(k)$ and $u(k)$, respectively, defined by

$$\phi_{\mathrm{f}}(k) \triangleq G_{\mathrm{f}}(\boldsymbol{q})\phi(k) \tag{B10}$$

$$u_{\mathrm{f}}(k) \triangleq G_{\mathrm{f}}(\boldsymbol{q})u(k) \tag{B11}$$

The filter G_{f} has the form

$$G_{\mathrm{f}}(\boldsymbol{q}) \triangleq D_{\mathrm{f}}^{-1}(\boldsymbol{q})N_{\mathrm{f}}(\boldsymbol{q}) \tag{B12}$$

where D_{f} and N_{f} are polynomial matrices and D_{f} is monic. The choice of the filter is discussed in the next section.

B4. Markov Parameters

In [15,16,25,27,36], the filter is based on the Markov parameters of the control-to-performance transfer matrix G_{zu}. In particular, for all complex numbers z for which the absolute value is greater than the spectral radius of A, it follows that

$$G_{zu}(z) = E_1(zI - A)^{-1}B = \sum_{i=0}^{\infty} \frac{H_i}{z^i} \tag{B13}$$

where, for all, $i \geq 1$, the ith Markov parameter of G_{zu} is defined by

$$H_i \triangleq E_1 A^{i-1} B \tag{B14}$$

B5. Retrospective Cost Function

Using the retrospective performance variable $\hat{z}(k)$, we define the retrospective cost function

$$J(k, \hat{\theta}) \triangleq \sum_{i=1}^{k} k_0[\hat{z}^{\mathrm{T}}(i)R_z\hat{z}(i) + (\phi(i)\hat{\theta})^{\mathrm{T}}R_u\phi(i)\hat{\theta}] + (\hat{\theta} - \theta(0))^{\mathrm{T}}R_\theta(\hat{\theta} - \theta(0)) \tag{B15}$$

where R_z, R_u, and R_θ are positive definite.

Proposition: Let $P(0) = R_\theta^{-1}$. Then, for all $k \geq k_0$, the retrospective cost function (B15) has a unique global minimizer $\theta(k)$, which is given by

$$\theta(k) = \theta(k-1) - P(k-1)\tilde{\phi}(k)^{\mathrm{T}}\Gamma(k)^{-1}[\tilde{\phi}(k)\theta(k-1) + \tilde{z}(k)] \tag{B16}$$

$$P(k) = P(k-1) - P(k-1)\tilde{\phi}(k)^{\mathrm{T}}\Gamma(k)^{-1}\tilde{\phi}(k)P(k-1) \tag{B17}$$

where

$$\tilde{\phi}(k) \triangleq \begin{bmatrix} \phi_f(k) \\ \phi(k) \end{bmatrix} \in \mathbb{R}^{(l_z+l_u)\times l_\theta} \tag{B18}$$

$$\tilde{R}(k) \triangleq \begin{bmatrix} R_z(k) & 0 \\ 0 & R_u(k) \end{bmatrix} \in \mathbb{R}^{(l_z+l_u)\times(l_z+l_u)} \tag{B19}$$

$$\tilde{z}(k) \triangleq \begin{bmatrix} z(k) - u_f(k) \\ 0 \end{bmatrix} \in \mathbb{R}^{(l_z+l_u)} \tag{B20}$$

$$\Gamma(k) \triangleq \tilde{R}(k)^{-1} + \tilde{\phi}(k)P(k-1)\tilde{\phi}(k)^{\mathrm{T}} \tag{B21}$$

■

Acknowledgments

The authors wish to thank Gerardo Cruz, Ming Yu, Shicong Dai, and Frantisek Sobolic for their assistance in this research. This research was supported in part by the Office of Naval Research under grant N00014-14-1-0596, the National Science Foundation under grant CMMI 1160916, and by NASA under contract number NNX12AM54A.

References

[1] Blakelock, J. H., *Automatic Control of Aircraft and Missiles*, Wiley, New York, 1991.
[2] Nelson, R. C., *Flight Stability and Automatic Control*, McGraw–Hill, New York, 1998.

[3] McRuer, D. T., Graham, D., and Ashkenas, I., *Aircraft Dynamics and Automatic Control*, Princeton Univ. Press, Princeton, NJ, 2014.
[4] Stevens, B. L., Lewis, F. L., and Johnson, E. N., *Aircraft Control and Simulation: Dynamics, Controls Design, and Autonomous Systems*, Wiley, Hoboken, NJ, 2015.
[5] Tewari, A., *Advanced Control of Aircraft, Spacecraft and Rockets*, Wiley, Hoboken, NJ, 2011.
[6] Adams, R. J., Buffington, J. M., Sparks, A. G., and Banda, S. S., *Robust Multivariable Flight Control*, Springer–Verlag, New York, 2012.
[7] Lavretsky, E., and Wise, K., *Robust and Adaptive Control: With Aerospace Applications*, Springer–Verlag, New York, 2012.
[8] Nguyen, N., Krishnakumar, K., Kaneshige, J., and Nespeca, P., "Dynamics and Adaptive Control for Stability Recovery of Damaged Asymmetric Aircraft," *AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 2006, pp. 21–24; also AIAA Paper 2006-6049, 2006.
[9] Nguyen, N. T., Krishnakumar, K. S., Kaneshige, J. T., and Nespeca, P. P., "Flight Dynamics and Hybrid Adaptive Control of Damaged Aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 3, 2008, pp. 751–764.
doi:10.2514/1.28142
[10] Liu, Y., Tao, G., and Joshi, S. M., "Modeling and Model Reference Adaptive Control of Aircraft with Asymmetric Damage," *Journal of Guidance, Control, and Dynamics*, Vol. 33, No. 5, 2010, pp. 1500–1517.
doi:10.2514/1.47996
[11] Stepanyan, V., Krishnakumar, K., and Nguyen, N., "Adaptive Control of a Transport Aircraft Using Differential Thrust," *AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2009-5741, 2009.
[12] Dydek, Z. T., Annaswamy, A. M., and Lavretsky, E., "Adaptive Control and the NASA X-15-3 Flight Revisited," *IEEE Control Systems*, Vol. 30, No. 3, 2010, pp. 32–48.
doi:10.1109/MCS.2010.936292
[13] Hovakimyan, N., Cao, C., Kharisov, E., Xargay, E., and Gregory, I. M., "L1 Adaptive Control for Safety-Critical Systems," *IEEE Control Systems*, Vol. 31, No. 5, 2011, pp. 54–104.
doi:10.1109/MCS.2011.941961
[14] Rahman, Y., Xie, A., Hoagg, J. B., and Bernstein, D. S., "A Tutorial and Overview of Retrospective-Cost-Based Adaptive Control," *Proceedings of American Control Conference*, 2016, pp. 3386–3409.
[15] Venugopal, R., and Bernstein, D. S., "Adaptive Disturbance Rejection Using ARMARKOV System Representations," *IEEE Transactions on Control Systems Technology*, Vol. 8, No. 2, 2000, pp. 257–269.
doi:10.1109/87.826797
[16] Santillo, M. A., and Bernstein, D. S., "Adaptive Control Based on Retrospective Cost Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 33, No. 2, 2010, pp. 289–304.
doi:10.2514/1.46741
[17] Hoagg, J. B., and Bernstein, D. S., "Retrospective Cost Model Reference Adaptive Control for Nonminimum-Phase Systems," *Journal of Guidance, Control, and Dynamics*, Vol. 35, No. 6, 2012, pp. 1767–1786.
doi:10.2514/1.57001
[18] Dai, S., Ren, Z., and Bernstein, D. S., "Adaptive Control of Nonminimum-Phase Systems Using Shifted Laurent Series," *International Journal of Control*, Vol. 90, No. 3, 2017, pp. 423–443.
doi:10.1080/00207179.2016.1183173
[19] Ansari, A., and Bernstein, D. S., "Adaptive Control of an Aircraft with Uncertain Nonminimum-Phase Dynamics," *Proceedings of American Control Conference*, 2015, pp. 844–849.
[20] Jordan, T., Langford, W., Belcastro, C., Foster, J., Shah, G., Howland, G., and Kidd, R., "Development of a Dynamically Scaled Generic Transport Model Testbed for Flight Research Experiments," *AUVSI Unmanned Unlimited*, NASA Langley Research Center, Hampton, VA, 2004.
[21] Bailey, R. M., Hostetler, R. W., Barnes, K. N., Belcastro, C. M., and Belcastro, C. M., "Experimental Validation: Subscale Aircraft Ground Facilities and Integrated Test Capability," *Proceedings of AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 2005, p. 110; also AIAA Paper 2005-6433, 2005.
[22] Jordan, T. L., and Bailey, R. M., "NASA Langleys AirSTAR Testbed: A Subscale Flight Test Capability for Flight Dynamics and Control System Experiments," *Proceedings of AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 2008, pp. 18–21; also AIAA Paper 2008-6660, 2008.
[23] Cunningham, K., Cox, D. E., Murri, D. G., and Riddick, S. E., "A Piloted Evaluation of Damage Accommodating Flight Control Using a Remotely Piloted Vehicle," *Proceedings of AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2011-6451, 2011.
[24] Santillo, M. A., Holzel, M. S., Hoagg, J. B., and Bernstein, D. S., "Adaptive Control of the NASA Generic Transport Model Using Retrospective Cost Optimization," *Proceedings of AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2009-5616, 2009.
[25] Yu, M. J., Rahman, Y., Atkins, E. M., Kolmanovsky, I., and Bernstein, D. S., "Minimal Modeling Adaptive Control of the NASA Generic Transport Model with Unknown Control-Surface Faults," *Proceedings of AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2013-4693, 2013.
[26] Yu, M.-J., Zhong, J., Atkins, E. M., Kolmanovsky, I. V., and Bernstein, D. S., "Trim-Commanded Adaptive Control for Waypoint-Defined Trajectory Following," *Proceedings of AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2013-5000, 2013.
[27] Sobolic, F., and Bernstein, D. S., "Aerodynamic-Free Adaptive Control of the NASA Generic Transport Model," *Proceedings of AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2013-4999, 2013.
[28] Ansari, A., Yu, M.-J., and Bernstein, D. S., "Exploration and Mapping of an Unknown Flight Envelope," *Proceedings of Dynamic Systems Control Conference*, 2014, pp. 523–528.
[29] Cox, D., "GTM DesignSim: The Generic Transport Model," 2008, https://github.com/nasa/GTM_DesignSim [accessed 16 Feb. 2017].
[30] Elgersma, M. R., and Morton, B. G., "Nonlinear Six-Degree-of-Freedom Aircraft Trim," *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 2, 2000, pp. 305–311.
doi:10.2514/2.4523
[31] Paranjape, A., Sinha, N. K., and Ananthkrishnan, N., "Use of Bifurcation and Continuation Methods for Aircraft Trim and Stability Analysis—A State-of-the-Art," *Journal of Aerospace Science and Technology*, Vol. 60, No. 2, 2008, p. 85.
[32] Sinha, N. K., and Ananthkrishnan, N., *Elementary Flight Dynamics with an Introduction to Bifurcation and Continuation Methods*, CRC Press, Boca Raton, FL, 2013.
[33] Chang, B., Kwatny, H., Ballouz, E. R., and Hartman, D., "Aircraft Trim Recovery from Highly Nonlinear Upset Conditions," *Proceedings of AIAA Guidance, Navigation, and Control Conference and Exhibit*, AIAA, Reston, VA, 2016, p. 880; also AIAA Paper 2016-0880, 2016.
[34] Rhudy, M., "Analytical Determination of Trim Values for Nonlinear Fixed Wing Aircraft Models," *AIAA Modeling and Simulation Technologies Conference*, AIAA, Reston, VA, 2016, p. 4301; also AIAA Paper 2016-4301, 2016.
[35] Yan, J., and Bernstein, D. S., "Minimum Modelling Retrospective Cost Adaptive Control of Uncertain Hammerstein Systems Using Auxiliary Nonlinearities," *International Journal of Control*, Vol. 87, No. 3, 2014, pp. 483–505.
doi:10.1080/00207179.2013.842264
[36] Sumer, E. D., Hoagg, J. B., and Bernstein, D. S., "Broadband Disturbance Rejection Using Retrospective Cost Adaptive Control," *Proceedings of IEEE Conference on Decision and Control*, Oct. 2012, pp. 1–10.

J. P. How
Associate Editor