Fault-Tolerant Adaptive Model Inversion Control for Vision-Based Autonomous Air Refueling

John Valasek,[*] Douglas Famularo,[†] and Monika Marwaha[‡]
Texas A & M University, College Station, Texas 77843-3141

DOI: 10.2514/1.G001888

The practical autonomous air refueling of unmanned air system tanker aircraft to unmanned air system receiver aircraft will require an integrated relative navigation system and controller that is tolerant to faults. This paper develops and demonstrates a fault-tolerant structured-adaptive-model-inversion controller integrated with a reliable relative-position sensor for this autonomous air-refueling scenario using the probe-and-drogue method. The structured-adaptive-model-inversion controller does not depend on fault-detection information, yet reconfigures and provides smooth trajectory tracking and probe docking in the presence of control-effector failure. The controller also handles parameter uncertainty in the receiver-aircraft model. In this paper, the controller is integrated with a vision-based relative-position sensor, which tracks the relative position of the drogue, and a reference-trajectory generator. The feasibility and performance of the controller and integrated system are demonstrated with simulated docking maneuvers with a nonstationary drogue, in the presence of system uncertainties and control-effector failures. The results presented in the paper demonstrate that the integrated controller/sensor system can provide successful docking in the presence of system uncertainties for a specified class of control-effector failures.

Nomenclature

A	=	unforced dynamics of the system
A_{est}	=	estimated A matrix
A_h	=	Hurwitz gain matrix
A_r	=	unforced dynamics of the reference system
B	=	control effectiveness of the system
B_{est}	=	estimated B matrix
B_r	=	control effectiveness of the reference system
C_a	=	estimate of C_a^* matrix
C_a^*	=	linear map between A and A_{est}
D	=	linear map between calculated and applied control signal
D^*	=	truth value for D matrix
E	=	bias term between calculated and applied control signal
E^*	=	truth value for E matrix
g	=	acceleration due to gravity
h	=	altitude
I	=	moment-of-inertia matrix
J	=	kinematic relationship between position and velocity coordinates
J_r	=	kinematic relationship between position and velocity reference coordinates
M	=	Mach number
m	=	vehicle mass
P	=	positive definite matrix
Q	=	positive definite matrix
s	=	position-level tracking errors
t	=	current time of refueling process
t_0	=	initial time of refueling process
t_1	=	end time for stage 1 of refueling process
t_2	=	end time for stage 2 of refueling process
u_{applied}	=	control signal applied to the system
$u_{\text{calculated}}$	=	control signal calculated by control algorithm
u_r	=	reference input signal
W_1, W_2, W_3	=	positive definite weighting matrices
(X_b, Y_b, Z_b)	=	body-fixed axis system
(X_d, Y_d, Z_d)	=	initial offset between drogue and probe
(X_n, Y_n, Z_n)	=	inertial axis system
x	=	velocity-level tracking errors
$(x_d(t), y_d(t), z_d(t))$	=	coordinates of drogue along inertial axes
$(x_r(t), y_r(t), z_r(t))$	=	coordinates of reference trajectory along inertial axes
y	=	total tracking error
α	=	angle of attack
δa	=	perturbation in aileron deflection from trim value
δe	=	perturbation in elevator deflection from trim value
δN	=	perturbation in vectoring engine nozzle from trim value
$(\delta p, \delta q, \delta r)$	=	perturbations in body-axis roll, pitch, yaw rates
$(\delta\text{pc}1, \delta\text{pc}2, \delta\text{pc}3)$	=	perturbations in pressure control nozzle force from trim value
δr	=	perturbation in rudder deflection from trim value
$\delta T\%$	=	perturbation in percentage of maximum thrust from trim value
$(\delta u, \delta v, \delta w)$	=	perturbations in inertial velocities
$(\delta X, \delta Y, \delta Z)$	=	perturbations in inertial positions
$(\delta\phi, \delta\theta, \delta\psi)$	=	perturbations in bank, pitch, heading attitude angles
λ	=	positive definite matrix
σ	=	position-level coordinates
σ_r	=	position-level reference-system coordinates
ϕ	=	forcing function on velocity-level error dynamics

Presented as Paper 2009-1887 at the AIAA Infotech@Aerospace Conference, Seattle, WA, 6–9 April 2009; received 2 December 2015; revision received 18 November 2016; accepted for publication 26 November 2016; published online 23 February 2017. Copyright © 2016 by John Valasek, Douglas Famularo, and Monika Marwaha. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission. All requests for copying and permission to reprint should be submitted to CCC at www.copyright.com; employ the ISSN 0731-5090 (print) or 1533-3884 (online) to initiate your request. See also AIAA Rights and Permissions www.aiaa.org/randp.

[*] Professor and Director, Vehicle Systems and Control Laboratory, Aerospace Engineering Department; valasek@tamu.edu. Fellow AIAA.

[†] Graduate Research Assistant, Vehicle Systems and Control Laboratory, Aerospace Engineering Department; dougfamularo@gmail.com. Student Member AIAA.

[‡] Graduate Research Assistant, Vehicle Systems and Control Laboratory, Aerospace Engineering Department; monika_marwaha@tamu.edu. Student Member AIAA.

ω = velocity-level coordinates
ω_r = velocity-level reference-system coordinates

I. Introduction

ONE of the important factors that will contribute to the future successful deployment and operation of unmanned aerial vehicles (UAVs) is reliable autonomous in-flight air refueling [1,2]. Two commonly used methods for aerial refueling are the boom-and-receptacle and probe-and-drogue methods. In the boom-and-receptacle method, a refueling boom on the rear of the tanker aircraft is steered into a refueling port on the receiver aircraft. Historically, this procedure requires a special tanker aircraft with a human boom operator onboard. Doebbler et al. successfully demonstrated the boom-and-receptacle method using vision-based sensors for unmanned aerial vehicles [3]. In the probe-and-drogue method, the receiver aircraft has a probe that must be placed or docked into the drogue, an aerodynamically stabilized device consisting of a flexible basket attached to the tanker aircraft by a flexible hose. This is the refueling method considered in this paper. The probe-and-drogue method requires a reliable tracking controller on the receiver aircraft, a function that is provided by the human pilot on manned aircraft, and by guidance laws and a robust controller on unmanned aircraft.

Nalepka and Hinchman have provided an overview of the various challenges involved in aerial refueling for unmanned aerial vehicles [2]. Over the last two decades, research has been conducted using several control approaches in combination with various types of sensors for obtaining accurate measurements of the drogue position. Some of the sensors previously considered for refueling are the Global Positioning System (GPS) [4], sensors with pattern recognition, or vision-based navigation systems [5]. Stepanyan et al. designed an aerial-refueling autopilot using techniques from differential games and adaptive control [6]. It is assumed that perfect measurements are obtained without a true relative navigation sensor. Vortex-induced uncertainties are handled using an adaptive-control methodology to handle longitudinal dynamics. Ochi and Kominami used proportional navigation guidance and line-of-sight angle control for aerial refueling, and included a turbulence model [7]. A quantitative feedback theory controller is used by Pachter et al. to design an automatic station-keeping flight-control system for regulating the position of an aircraft receiving fuel relative to the tanker aircraft, assuming that a relative navigation sensor was available [8]. Pollini et al. have developed an artificial vision system for relative-position estimation based on localization of passive infrared markers, which have a known geometry distribution over the leader airframe or drogue body [9]. A new set of nonlinear equations of motion have been derived using the relative motion between receiver and tanker aircraft, and also considering the aerodynamic coupling due to the trailing vortex shed by the tanker aircraft [10]. More recently, an adaptive-control-based controller with guaranteed transient performance has been designed and demonstrated for aerial refueling [11,12].

Earlier work by the authors developed an optimal nonzero set point controller with control rate weighting (NZSP-CRW) to track and dock with a stationary drogue in Dryden light turbulence using a cooperative vision-based sensor [1], and then extended this work to a moving drogue using the command generator tracker (CGT) control scheme [13,14]. The main drawback to both NZSP-CRW and CGT is that the position or trajectory of the drogue is assumed to be known a priori, and an accurate model of the drogue dynamics is required. To address these limitations, the authors developed a reference-observer-based tracking controller that is able to successfully track and dock with a probe in the presence of moderate turbulence and sensor noise [5].

This paper extends and improves upon the earlier body of work by developing a fault-tolerant structured adaptive model inversion (SAMI) controller, which does not require an accurate model of the receiver aircraft or the drogue dynamics, yet is able to handle control-effector failures. SAMI is based upon dynamic inversion and feedback linearization, and has been used for many spacecraft and aircraft problems [15,16]. A fault-tolerant version of SAMI was developed to handle control-effector failures without using a dedicated fault-identification algorithm, and was subsequently applied to systems with continuous control effectors [17] and systems with discrete control effectors [16]. In this work, a fault-tolerant SAMI controller is developed and integrated with a vision-based relative-position sensor for the purpose of tracking a reference docking trajectory, which is generated onboard the receiver aircraft, in real time. The system is able to provide successful docking in the presence of system uncertainties and control-effector failures. Additionally, the fault-tolerant SAMI controller developed here for autonomous air refueling using the probe-and-drogue method can also be applied to the boom-and-receptacle method.

The paper is organized as follows. Section II describes the visual navigation (VisNav) vision-based sensor and its measurements. Section III describes the equations used for the reference-trajectory generation. Development of fault-tolerant SAMI is shown in Sec. IV along with the stability proof. Section V describes the vehicle model and simulation, and Sec. VI contains a simulation example to investigate the VisNav sensor performance characteristics and a performance evaluation of fault-tolerant SAMI. Conclusions and recommendations for future research are presented in Sec. VII, and the Appendix contains details of the receiver-aircraft model.

II. Vision-Based Navigation Sensor

One of the important requirements for autonomous aerial refueling (AAR) is a sensor to measure the relative position and orientation between the receiver aircraft and the tanker aircraft. Some of the methods considered in the past for aerial refueling are GPS and visual servoing with pattern-recognition software [4]. Although GPS measurements have been made with 1–2 cm accuracy for formation flying, problems associated with lock on, integer ambiguity, and low bandwidth present challenges for application to in-flight refueling. Pattern-recognition codes are not reliable in all lighting conditions, and with adequate fault tolerance, may require large amounts of computational power to converge with sufficient confidence to a solution.

The vision-based navigation system called VisNav [1] provides high precision, six-degree-of-freedom (DOF) information for real-time navigation applications. The sensors are small in size and need low power, and hence are suitable for UAV platforms. They also provide the desired accuracy in measurements, which is required for in-flight refueling. When VisNav is operating, the digital signal processor commands a beacon controller to signal each light-emitting-diode (LED) beacon to activate in turn. As each beacon turns on, light energy comes through the wide angle lens and is focused onto the position-sensing diode (PSD). The focused light creates a centroid, or spot, on the photo diode, which causes a current imbalance in the four terminals on each side of the PSD, as shown in Fig. 1. The closer the light centroid is to one side of the photo diode, the higher the current in the nearest terminal. By measuring the

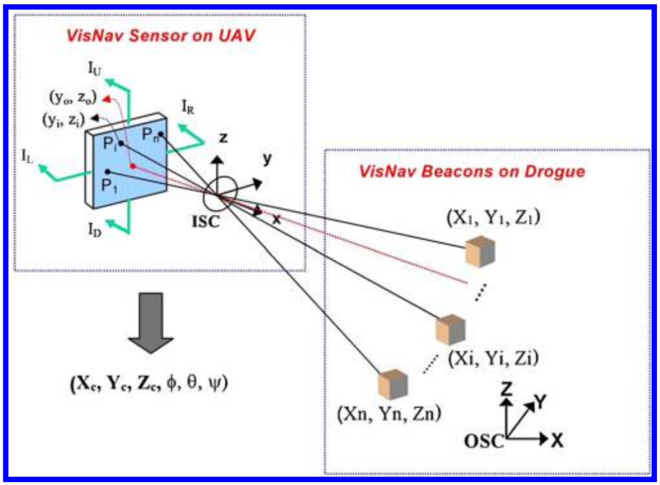

Fig. 1 VisNav sensor model.

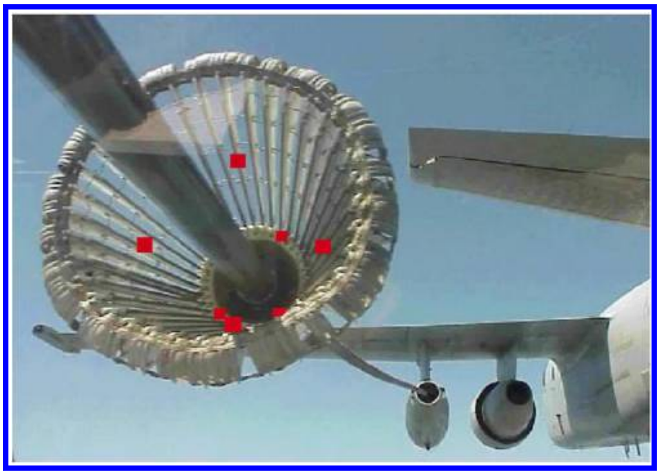

Fig. 2 Candidate beacon configuration for AAR, as seen from VisNav sensor.

Downloaded by TEXAS A & M UNIVERSITY on March 7, 2017 | http://arc.aiaa.org | DOI: 10.2514/1.G001888

voltage at each terminal, the two-dimensional (2-D) position of the light centroid on the PSD can be found with a nonlinear calibration function. From these measurements, unit line-of-sight vectors from the sensor to each beacon can be determined. Once measurements from four or more beacons are collected, they are passed to a Gaussian least-squares-differential-correction (GLSDC) algorithm [1]. This routine calculates the minimum-variance estimate of the position and orientation of the sensor relative to the target frame. To address the depth of field problem commonly associated with optical sensors, the power of each LED beacon is adaptively adjusted online to optimize the received energy amplitude. This maximizes the signal-to-noise ratio of each individual measurement and is accomplished by a wireless feedback loop closed at 100 Hz.

Figure 2 shows the location of the active beacon array mounted on the refueling drogue. In the GLSDC measurement model, there are six unknowns: the three relative positions and the three relative orientation coordinates. Each available beacon contributes two measurement equations: one for each 2-D coordinate of its projection on the PSD. Measurements from at least three beacons must be obtained before the system of equations may be solved, but a unique geometry cannot be obtained using only three beacons because there will be more than one configuration of three beacons that project identically onto the 2-D plane. At least four beacons are required to obtain a unique six-DOF navigation solution, and although using more than four beacons provides robustness to measurement noise, it also adds computational expense. A set of eight beacons provides a good balance between these two factors for the AAR application. GLSDC finds the best-fit solution for a given set of measurements and an assumed amount of measurement noise [1]. When measurement noise is accounted for, it is the quality of the geometry [the three-dimensional (3-D) arrangement of the beacons] and not the number of beacons that determines the accuracy of the pose estimate at a given range. This provides redundancy in case a beacon falls outside of the field of view, and the additional measurements improve the convergence performance of the estimation routine. A second set of beacons that are close together may then be used for close-proximity navigation. A desirable configuration ensures that the lateral extent of the beacon array takes up at least 10% of the sensor field of view within the range of interest. Valasek et al. [1] detail how the VisNav sensor system is specifically configured for the aerial-refueling task. The same configuration is used for the system in this paper.

III. Reference-Trajectory Generation

Figure 3 shows the Earth-fixed inertial axis system (X_n, Y_n, Z_n) oriented with the X_n axis pointing along the heading of both the tanker and receiver aircraft, and the Z_n axis pointing in the direction of gravity. The body axis (X_b, Y_b, Z_b) is attached to the receiver aircraft with the origin at its center of gravity.

The initial offset between the mean position of the refueling drogue and the probe attached to the receiver aircraft, as measured along the inertial axis, is given by X_d, Y_d, Z_d. The receiver aircraft is initially lined up in the (Y_n, Z_n) plane behind the drogue, and docking is accomplished by modulating the drogue position forward along X_n. The concept of using a reference trajectory comes from five and is designed using the same idea (i.e., in two stages). In the first stage, the refueling probe on the receiver aircraft tries to line up behind the mean position of the drogue so that the initial offset (Y_d, Z_d) becomes zero. The flight trajectory is designed using a fifth-order polynomial to ensure continuity in the position, velocity, and acceleration states at both end points of the trajectory. The continuity in acceleration enforces continuity in the desired control effort without much computational burden. To illustrate, consider the offset along the Y_n axis. At the beginning of stage 1, the reference trajectory is a steady-level 1 g flight path, and hence perturbations in reference position, velocity, and acceleration are zero at the initial time t_0. At the final time t_1 of stage 1, the reference position is the offset Y_d, and the velocity and acceleration are zero. Let $y_r(t)$ be the coordinate of the reference trajectory along the Y_n axis. The boundary conditions are given as

$$y_r(t) = 0, \qquad \frac{\mathrm{d}y_r}{\mathrm{d}t}(t) = 0, \qquad \frac{\mathrm{d}^2 y_r}{\mathrm{d}t^2}(t) = 0 \text{ at } t = 0 \tag{1}$$

$$y_r(t) = Y_d, \qquad \frac{\mathrm{d}y_r}{\mathrm{d}t}(t) = 0, \qquad \frac{\mathrm{d}^2 y_r}{\mathrm{d}t^2}(t) = 0 \text{ at } t = t1 \tag{2}$$

To enforce the boundary conditions, a fifth-order polynomial is selected, so that $y_r(t)$ can be written as

$$y_r(t) = y_0 + y_1 t + y_2 t^2 + y_3 t^3 + y_4 t^4 + y_5 t^5, \qquad \text{if} \quad t \leq t_1 \tag{3}$$

The parameters $y_0, \ldots, y_5$ can be calculated by imposing the boundary conditions introduced previously, and $z_r(t)$ for stage 1 can be

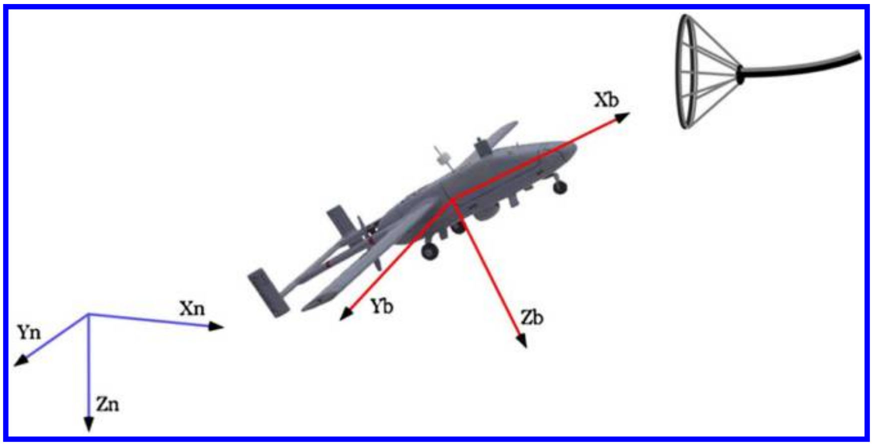

Fig. 3 Axis system.

calculated in a similar manner. To zero the offset X_d, a similar fifth-order polynomial is used to design the reference trajectory using the initial and final times as the initial time of the first stage (t_0), and the final time of the second stage (t_2), respectively. The probe follows the drogue positions along the Y_n and Z_n axes during the second stage. The drogue exhibits random oscillatory behavior in the plane parallel to the (Y_n, Z_n) plane. The drogue is also perturbed in translation along the direction of flight, but the magnitude of these perturbations is small. The mean position of the drogue may be estimated by taking an average of the drogue position over a period of 10 s prior to initiating the docking maneuver. The objective is to design the reference trajectory as a smooth transition between the mean drogue position (Y_d, Z_d) and the current drogue position $[y_d(t), z_d(t)]$, along the Y_n and Z_n axes, respectively. Let us consider tracking along the Y_n axis. The reference trajectory using the fifth-order interpolation is given by

$$y_r(t) = Y_d + \kappa y_d(t), \qquad \text{if } t > t_1 \quad \text{and} \quad t \le t_2 \tag{4}$$

$$\text{in which } \kappa = 10\tau^3 - 15\tau^4 + 6\tau^5 \tag{5}$$

$$\text{and} \quad \tau = \frac{t - t_1}{t_2 - t_1} \tag{6}$$

The time durations of the first and second stages (t_1 and t_2) in the preceding equations are design parameters selected as functions of the initial offset (X_d, Y_d, Z_d) to ensure that the reference trajectory does not demand excessive control and state rates. During the first stage, the receiver aircraft only needs an approximate location of the drogue mean position, and so a GPS-based approximate measurement is used. During stage 2, an accurate drogue position is needed because the receiver aircraft is closer to the drogue, and so VisNav measurements are used. The VisNav measurements become more accurate as the range between the sensor and beacons decreases [5].

IV. Fault-Tolerant SAMI Control

As mentioned before, fault-tolerant SAMI does not depend on fault-detection information and reconfigures itself if any fault occurs. The mathematical model used for handling a failure is given by the following equation:

$$u_{\text{applied}} = D u_{\text{calculated}} + E \tag{7}$$

in which $D \in \mathbb{R}^{p \times p}$ and $E \in \mathbb{R}^p$ are constant for a given control configuration, but may change and converge to other constant values if there is control failure. The vector $u_{\text{applied}} \in \mathbb{R}^p$ is the control applied to the system, and $u_{\text{calculated}} \in \mathbb{R}^p$ is the control calculated by the control algorithm. To illustrate, consider the following generic 3-D example equation:

$$\begin{bmatrix} u_{a1} \\ u_{a2} \\ u_{a3} \end{bmatrix} = \begin{bmatrix} D_{11} & 0 & 0 \\ 0 & D_{22} & 0 \\ 0 & 0 & D_{33} \end{bmatrix} \begin{bmatrix} u_{c1} \\ u_{c2} \\ u_{c3} \end{bmatrix} + \begin{bmatrix} E_1 \\ E_2 \\ E_3 \end{bmatrix} \tag{8}$$

In the absence of any failure, the matrix D is the identity matrix and the vector E is equal to the zero vector. In this situation, u_{applied} is equal to $u_{\text{calculated}}$. If there is a control failure, the corresponding element along the diagonal of D will go to zero and the corresponding element of E will go to a constant value at which that control is frozen. Therefore, u_{applied} will not equal $u_{\text{calculated}}$.

A. Development of Dynamic Equations and Control Law

The motion of the vehicle is given by the following set of equations:

$$\dot{\sigma} = J(\sigma)\omega \tag{9}$$

$$\dot{\omega} = A(\sigma, \omega) + B(\sigma, \omega) u_{\text{applied}} \tag{10}$$

in which $\sigma \in \mathbb{R}^n$ are translational and rotational position-level coordinates, and $\omega \in \mathbb{R}^n$ are translational and rotational velocity-level coordinates. $J(\sigma) \in \mathbb{R}^{n \times n}$ represents the nonlinear relationship between $\dot{\sigma}$ and ω, $A(\sigma, \omega) \in \mathbb{R}^n$ represents the unforced behavior of the system, and $B(\sigma, \omega) \in \mathbb{R}^{n \times p}$ represents the control-effectiveness matrix of the system. The $A(\sigma, \omega)$ and $B(\sigma, \omega)$ matrices of the original plant contain uncertainties or are not known very well. A reference model with the same structure is selected and can be expressed as

$$\dot{\sigma}_r = J_r(\sigma_r)\omega_r \tag{11}$$

$$\dot{\omega}_r = A_r(\sigma_r, \omega_r) + B_r(\sigma_r, \omega_r) u_r \tag{12}$$

in which σ_r, ω_r, u_r, J_r, A_r, and B_r are known and have the same dimension as σ, ω, u_{applied}, J, A, and B, respectively. The tracking errors between the states and reference states are once again divided as position-level and velocity-level errors. They are defined as

$$s = \sigma - \sigma_r \tag{13}$$

$$x = \omega - \omega_r \tag{14}$$

$$\dot{x} = A(\sigma, \omega) + B(\sigma, \omega) u_{\text{applied}} - \dot{\omega}_r \tag{15}$$

On substituting the expression for u_{applied} from Eq. (7) into Eq. (15), the following equation is obtained:

$$\dot{x} = A(\sigma, \omega) + B(\sigma, \omega) D u_{\text{calculated}} + B(\sigma, \omega) E - \dot{\omega}_r \tag{16}$$

The control objective is to make the error between the reference model and the actual plant states to go to zero. Hence, the following dynamics are prescribed for the error x:

$$\dot{x} = A_h x + \phi \tag{17}$$

in which $A_h \in \mathbb{R}^{n \times n}$ is a Hurwitz matrix. A_h is selected by the designer, and by placing its eigenvalues properly, one can specify how fast the velocity error goes to zero. The vector $\phi \in \mathbb{R}^n$ is a forcing function on the velocity error dynamics, which helps in achieving the tracking objective. Upon adding and subtracting Eq. (17) to Eq. (16), one is left with

$$\begin{aligned} \dot{x} = {} & A_h x + \phi + A(\sigma, \omega) + B(\sigma, \omega) D u_{\text{calculated}} + B(\sigma, \omega) E \\ & - (\dot{w}_r + A_h x + \phi) \end{aligned} \tag{18}$$

It is assumed that all of the state variables are measurable. Because the rightmost expression in parentheses in Eq. (18) is known, let

$$\psi = \dot{w}_r + A_h x + \phi \tag{19}$$

As system matrices A and B are not known exactly, best guesses A_{est} and B_{est} will be used in real time for these matrices. The following relationship between the actual A and B matrices and their estimates is established:

$$C_a^* A_{\text{est}} = A \tag{20}$$

$$B_{\text{est}} D^* = BD \tag{21}$$

$$B_{\text{est}} E^* = BE \tag{22}$$

Therefore, Eq. (18) can now be rewritten as

$$\begin{aligned} \dot{x} = {} & A_h x + \phi + C_a^* A_{\text{est}}(\sigma, \omega) + B_{\text{est}}(\sigma, \omega) D^* u_{\text{calculated}} \\ & + B_{\text{est}}(\sigma, \omega) E^* - \psi \end{aligned} \tag{23}$$

The actual matrices $C_a^* \in \mathbb{R}^{n \times n}$, $D^* \in \mathbb{R}^{p \times p}$, and $E^* \in \mathbb{R}^p$ are unknown. If they were known, the actual A and B matrices could be calculated exactly, and adaptive control would not be necessary. Therefore, the adapted learning matrix $C_a \in \mathbb{R}^{n \times n}$ will be updated online and will account for uncertainty in the A matrix, whereas the already-defined matrices D and E will also be updated online and will account for uncertainties in the B matrix. Using dynamic inversion, we can solve for the control vector so that the velocity error x has the desired dynamics:

$$u_{\text{calculated}} = (B_{\text{est}} D)^{-1}(\psi - C_a A_{\text{est}} - B_{\text{est}} E) \tag{24}$$

From Eq. (24), one can solve for ψ:

$$\psi = B_{\text{est}} D u_{\text{calculated}} + C_a A_{\text{est}} + B_{\text{est}} E \tag{25}$$

and update Eq. (23) by plugging in this value of ψ

$$\dot{x} = A_h x + \phi + \tilde{C}_a A_{\text{est}} + B_{\text{est}} \tilde{D} u_{\text{calculated}} + B_{\text{est}} \tilde{E} \tag{26}$$

in which

$$\tilde{C}_a = C_a^* - C_a \tag{27}$$

$$\tilde{D} = D^* - D \tag{28}$$

$$\tilde{E} = E^* - E \tag{29}$$

B. Development of Update Laws Through Lyapunov Stability Analysis

To show perfect tracking of both the velocity- and position-level coordinates, the tracking error should have a contribution from both levels of the state vector. The control law is derived from the dynamic part, and it is assumed that the adaptive mechanism provides perfect velocity tracking. This does not ensure that the position reference will be tracked correctly. If the actual initial position deviates from that of the reference model or if errors in the velocity-level states develop during the transient stage, before perfect velocity tracking is achieved, the position will stray from the reference, and no attempts at correcting this error will be made unless this deviation is identified. Therefore, define the total tracking error to include error terms for both the kinematic and dynamic states as

$$y = \dot{s} + \lambda s = J\omega - J_r \omega_r + \lambda s \tag{30}$$

in which $\lambda \in \mathbb{R}^{n \times n}$ is a positive definite matrix. Adding and subtracting $J\omega_r$ to the right-hand side of Eq. (30) leave

$$y = Jx + J\omega_r - J_r \omega_r + \lambda s \tag{31}$$

The following error dynamics are chosen to make sure that error converges to zero:

$$\dot{y} = A_h y \tag{32}$$

Just as mentioned, choosing all of the eigenvalues of A_h to lie in the left-half complex plane will guarantee that y is both stable and asymptotically approaching zero. Another expression for $\dot{y}$ can be calculated by differentiating Eq. (31) with respect to time.

$$\begin{aligned} \dot{y} &= J\dot{x} + \dot{J}x + (\dot{J} - \dot{J}_r)\omega_r + (J - J_r)\dot{\omega}_r + \lambda \dot{s} \\ &= J(A_h x + \phi + \tilde{C}_a A_{\text{est}} + B_{\text{est}} \tilde{D} u_{\text{calculated}} + B_{\text{est}} \tilde{E}) \\ &\quad + \dot{J}x + (\dot{J} - \dot{J}_r)\omega_r + (J - J_r)\dot{\omega}_r + \lambda \dot{s} \end{aligned} \tag{33}$$

An expression for ϕ can be extracted by letting the sum of all known terms in Eq. (33) be equal to the right-hand side of Eq. (32) with the intention of having them cancel:

$$JA_h x + J\phi + \dot{J}x + (\dot{J} - \dot{J}_r)\omega_r + (J - J_r)\dot{\omega}_r + \lambda \dot{s} = A_h y \tag{34}$$

$$\phi = J^{-1}(A_h y - \dot{J}\omega + \dot{J}_r \omega_r + J_r \dot{\omega}_r - \lambda \dot{s}) - \dot{\omega}_r - A_h x \tag{35}$$

Using this new expression for ϕ in Eq. (33) results in the final form of the tracking-error dynamics:

$$\dot{y} = A_h y + J(\tilde{C}_a A_{\text{est}} + B_{\text{est}} \tilde{D} u_{\text{calculated}} + B_{\text{est}} \tilde{E}) \tag{36}$$

Next, Lyapunov stability analysis is used to find the update laws for matrices C_a, D, and E that will lead to a controller that is both stable and able to achieve the control objective. The following candidate Lyapunov function is selected:

$$V = y^T P y + Tr(\tilde{C}_a^T W_1 \tilde{C}_a + \tilde{D}^T W_2 \tilde{D}) + \tilde{E}^T W_3 \tilde{E} \tag{37}$$

in which $P \in \mathbb{R}^{n \times n}$, $W_1 \in \mathbb{R}^{n \times n}$, $W_2 \in \mathbb{R}^{p \times p}$, and $W_3 \in \mathbb{R}^{p \times p}$ are positive definite weighting matrices, and y, $\tilde{C}_a$, $\tilde{D}$, and $\tilde{E}$ hold the same meaning as mentioned. The derivative of V is taken with respect to time, and a substitution is made for $\dot{y}$ according to the tracking-error dynamics, Eq. (36):

$$\dot{V} = \dot{y}^T P y + y^T P \dot{y} + 2Tr(\tilde{C}_a^T W_1 \dot{\tilde{C}}_a + \tilde{D}^T W_2 \dot{\tilde{D}}) + 2\tilde{E}^T W_3 \dot{\tilde{E}} \tag{38}$$

$$\begin{aligned} \dot{V} &= y^T A_h^T P y + y^T P A_h y \\ &\quad + 2Tr(A_{\text{est}}^T \tilde{C}_a^T J^T P y + u_{\text{calculated}}^T \tilde{D}^T B_{\text{est}}^T J^T P y) + \tilde{E}^T B_{\text{est}}^T J^T P y \\ &\quad + 2Tr(\tilde{C}_a^T W_1 \dot{\tilde{C}}_a + \tilde{D}^T W_2 \dot{\tilde{D}}) + 2\tilde{E}^T W_3 \dot{\tilde{E}} \end{aligned} \tag{39}$$

Using the Lyapunov equation, let $Q \in \mathbb{R}^{n \times n}$ be a positive definite matrix described by

$$PA_h + A_h^T P = -Q \tag{40}$$

Substituting Q into Eq. (39) leaves

$$\begin{aligned} \dot{V} &= -y^T Q y + 2Tr[(\tilde{C}_a^T (J^T P y A_{\text{est}}^T + W_1 \tilde{C}_a)] \\ &\quad + 2Tr[\tilde{D}^T (B_{\text{est}}^T J^T P y u_{\text{calculated}}^T + W_2 \dot{\tilde{D}})] \\ &\quad + 2\tilde{E}^T (B_{\text{est}}^T J^T P y + W_3 \dot{\tilde{E}}) \end{aligned} \tag{41}$$

With $\dot{V}$ in this form, the proper update laws to create a stable controller can be identified:

$$\begin{aligned} \dot{C}_a &= -W_1^{-1}(J^T P y A_{\text{est}}^T) \\ \dot{D} &= -W_2^{-1}(B_{\text{est}}^T J^T P y u_{\text{calculated}}^T) \\ \dot{E} &= -W_3^{-1}(B_{\text{est}}^T J^T P y) \end{aligned} \tag{42}$$

Substitution of the update laws into the expression for the time derivative of the Lyapunov function leaves

$$\dot{V} = -y^T Q y \tag{43}$$

which is negative semidefinite.

C. Analysis

The candidate Lyapunov function V, a function of $\tilde{C}_a, \tilde{D}, \tilde{E}$, and error y, is zero if and only if all of these independent variables are equal to zero and it is otherwise positive. From Eq. (43), it can be seen that the time derivative of the Lyapunov function $\dot{V}$ is zero if and only if $y = 0$ and that at all other times $\dot{V}$ is negative. This proves that $\dot{V}$ is negative semidefinite, which means $\tilde{C}, \tilde{D}, \tilde{E} \in L_\infty$ and $y \in (L_2 \cap L_\infty)$. From the expression of y in Eq. (30), it is concluded that error $s + \lambda \dot{s} \in (L_2 \cap L_\infty)$. Reference trajectories σ_r and $\dot{\sigma}_r$ are bounded, and $\in L_\infty$. As s and $\dot{s} \in L_\infty$ and reference trajectories are bounded, it is concluded from the definition of s and $\dot{s}$ that $(\sigma, \dot{\sigma}) \in L_\infty$. Using the kinematic relation between σ and ω, it is concluded that $\omega \in L_\infty$, and hence $A(\sigma, \omega)$ and $\phi \in L_\infty$. All of the terms in the expression for $\dot{y}$ are bounded, and therefore $\dot{y}$ itself is also bounded. From the discussed properties of V and $\dot{V}$ and using Barbalat's lemma [18], it is concluded that $\sigma \to \sigma_r$ and $\omega \to \omega_r$ as $t \to \infty$, that is, perfect tracking can be achieved for both the kinematic and dynamic level states of the system.

V. Air-Vehicle Model

A six-DOF nonlinear time-invariant model of the UCAV6 is considered as an unmanned receiver aircraft. It is a 60% scale AV-8B Harrier aircraft with the pilot and all pilot support devices removed, and with the mass properties and aerodynamics adjusted accordingly. The state vectors for this model, divided as position-level and velocity-level coordinates, are

$$\sigma = [\delta\phi \quad \delta\theta \quad \delta\psi \quad \delta X \quad \delta Y \quad \delta Z]^T \tag{44}$$

$$\omega = [\delta p \quad \delta q \quad \delta r \quad \delta u \quad \delta v \quad \delta w]^T \tag{45}$$

in which $\delta(\cdot)$ are the perturbations from the steady-state values, and the steady state is assumed as steady-level 1 g flight. Here, $\delta X, \delta Y, \delta Z$ are perturbations in the inertial positions; $\delta u, \delta v, \delta w$ are perturbations in the body-axis velocities; δp, δq, δr are perturbations in the body-axis angular velocities; and $\delta\phi$, $\delta\theta$, $\delta\psi$ are perturbations in the Euler attitude angles. Both the tanker and receiver aircraft are considered to be flying at 1 g steady-level flight with the same velocity and heading.

Elevon, thrust, and vectoring engine nozzles are used as longitudinal controls, and aileron and rudder are used as lateral/directional controls. To use the fault-tolerant algorithm, it is assumed that three smaller pressure-valve control nozzles are attached to the UCAV6, just as on the AV-8B Harrier. The inclusion of these extra control variables ensures that the plant remains controllable in the presence of a control failure. These nozzles affect both the lateral and longitudinal states; are labeled as $\delta pc1$, $\delta pc2$, and $\delta pc3$; and have units of force. The pressure control nozzles are redundant in the sense that, if there were no control failures, they would not be required for the plant to be controllable. However, because they are included in the model, the control algorithm does not treat them as redundant and render them unused, even in the nominal condition. The control variables δe (elevon), $\delta T\%$ (percentage of maximum thrust), δN (synchronized vectoring engine nozzles), δa (aileron), and δr (rudder) are perturbations in the control effectors from the trim values. The total control vector can be expressed as

$$u_{\text{calculated}} = [\delta a \quad \delta r \quad \delta e \quad \delta T\% \quad \delta N \quad \delta \text{pc1} \quad \delta \text{pc2} \quad \delta \text{pc3}]^T \tag{46}$$

With the state and control vectors defined, the kinematics and dynamics for the simulation can be expressed in the form of Eqs. (9) and (10).

$$\dot{\sigma} = J(\sigma)\omega$$

$$J(\sigma) = \begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) & 0 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) & 0 & 0 & 0 \\ 0 & \sin(\phi)\sec(\theta) & \cos(\phi)\sec(\theta) & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos(\alpha) & 0 & \sin(\alpha) \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -\sin(\alpha) & 0 & \cos(\alpha) \end{bmatrix} \tag{47}$$

$$\dot{\omega} = A(\sigma, \omega) + B(\sigma, \omega)u_{\text{applied}}$$

$$A(\sigma, \omega) = \begin{bmatrix} \frac{1}{I_1}(L_v v + L_p p + L_r r + (I_2 - I_3)qr) \\ \frac{1}{I_2}(M_u u + M_w w + M_q q + (I_3 - I_1)pr) \\ \frac{1}{I_3}(N_v v + N_p p + N_r r + (I_1 - I_2)pq) \\ X_u u + X_w w + X_q q - g\cos(\theta)\cos(\phi) + rv - qw \\ Y_v v + Y_p p + Y_r r + g\cos(\theta)\sin(\phi) + pw - ru \\ Z_u u + Z_w w + Z_q q - g\sin(\theta) + qu - pv \end{bmatrix} \tag{48}$$

$$B(\sigma, \omega) = \begin{bmatrix} L_{\delta a} & L_{\delta r} & 0 & 0 & 0 & L_{\delta_{\text{pc1}}} & L_{\delta_{\text{pc2}}} & 0 \\ 0 & 0 & M_{\delta e} & M_{\delta T} & M_{\delta N} & 0 & 0 & M_{\delta_{\text{pc3}}} \\ N_{\delta a} & N_{\delta r} & 0 & 0 & 0 & N_{\delta_{\text{pc1}}} & 0 & N_{\delta_{\text{pc3}}} \\ 0 & 0 & X_{\delta e} & X_{\delta T} & X_{\delta N} & X_{\delta_{\text{pc1}}} & 0 & 0 \\ Y_{\delta a} & Y_{\delta r} & 0 & 0 & 0 & 0 & Y_{\delta_{\text{pc2}}} & 0 \\ 0 & 0 & Z_{\delta e} & Z_{\delta T} & Z_{\delta N} & 0 & 0 & Z_{\delta_{\text{pc3}}} \end{bmatrix} \tag{49}$$

$$I = \begin{bmatrix} I_1 & 0 & 0 \\ 0 & I_2 & 0 \\ 0 & 0 & I_3 \end{bmatrix} = \begin{bmatrix} 692.2 & 0 & 0 \\ 0 & 1095.7 & 0 \\ 0 & 0 & 2287.5 \end{bmatrix} \text{kg} \cdot \text{m}^2 \tag{50}$$

in which $m = 6055$ kg is the vehicle mass, I is the vehicle moment of inertia, g is the acceleration due to gravity, and all of the X_x-type terms represent the various stability and control derivatives of the aircraft. The values used for these derivatives can be found in the Appendix.

VI. Numerical Example

The control objective for the receiver aircraft is to first line up behind the drogue in the (Y_n, Z_n) plane, and then dock the tip position of the refueling probe with a moving drogue receptacle with an accuracy of ±2 cm, and keep it there. The system is simulated for a flight condition of 0.37 Mach number and 10,000 m altitude. The VisNav-equipped UCAV6 receiver aircraft is trimmed at 3.5 deg angle of attack with an elevon deflection of 0.83 deg, with trim engine power of 60%. It is initially at a relative position 30 m behind, 15 m to the right, and 15 m below the drogue suspended from the tanker aircraft. The drogue basket is configured with eight LED VisNav beacons. The four outer beacons are located 20 cm from the nozzle in a 36-cm-diam circle. This location corresponds to a typical location where LED lights are currently installed by drogue manufacturers for nighttime refueling operations. The inner four beacons are located near the nozzle itself in a circle with a diameter of 5 cm. The VisNav PSD sensor is mounted on the UAV, 0.5 m behind the tip of the refueling probe and 13 cm below it. For the examples, the VisNav relative-position estimates are obtained from a simulation of the VisNav system that includes calibrations, range effects, corrections due to optical distortions, and sensor noise. The high-fidelity VisNav sensor system simulation is integrated with the fault-tolerant SAMI controller.

The actual control vector u_{applied} is calculated according to Eq. (7). For all cases presented, the parameters used in the update laws, Eq. (42), were as follows: the adaptive gains were chosen, such that $W_1 = 0.01$ and $W_2 = W_3 = 0.001$. The positive definite matrix P was solved for using the Lyapunov equation (40) with $Q = 10I_6$ and $A_h = -10I_6$. The positive definite matrix λ used in the total tracking-error expression, Eq. (30), was $10I_6$. Finally, the initial conditions used on the update parameters were $C_a(t=0) = I_6$, $D(t=0) = I_8$, and $E(t=0) = 0$. The matrices A_{est} and B_{est} can be found in the Appendix.

Table 1 Receiver-aircraft modeling uncertainty

Derivative	Uncertainty, %	Derivative	Uncertainty, %
C_{L_α}	5	$C_{l_{\delta a}}$	5
C_{m_α}	10	$C_{l_{\delta r}}$	5
C_{l_β}	20	$C_{m_{\delta e}}$	5
C_{n_β}	15	$C_{m_{\delta pwr}}$	5
C_{l_p}	15	$C_{n_{\delta a}}$	5
		$C_{n_{\delta r}}$	5

The reference trajectory is designed according to Sec. III, and a second-order spring–mass–damper model is used to simulate the drogue motion. The uncertainties defined in Table 1 are introduced into the system parameters. The controller designed in Sec. IV was derived under the assumption of full-state information, and hence measurements of all reference states that the receiver aircraft should follow are made available so that it can track the reference trajectory.

For all test cases, fault-tolerant performance is evaluated according to tracking errors for both the kinematic and dynamic level states during the docking maneuver, in the presence of control-effector failures. Two test cases are presented: in test case 1, the rudder is failed at 8 s and locked at 2 deg. In test case 2, the elevon is failed at 10 s and is locked at 1.2 deg.

A. Test Case 1

Once the rudder is failed at 8 s in test case 1, the states of the receiver aircraft drift from the reference trajectory. Fault-tolerant SAMI reconfigures and brings the states back to their desired reference values. Figure 4 shows the rudder failure and the extra control effort demanded of the rest of the control effectors.

Figure 5 shows the effect of the rudder failure on the orientations and angular-velocity states of the receiver aircraft. The maximum errors in bank angle and roll rate are 8 deg and 16 deg/s, respectively. A heading-angle error 10 deg occurs before it returns to

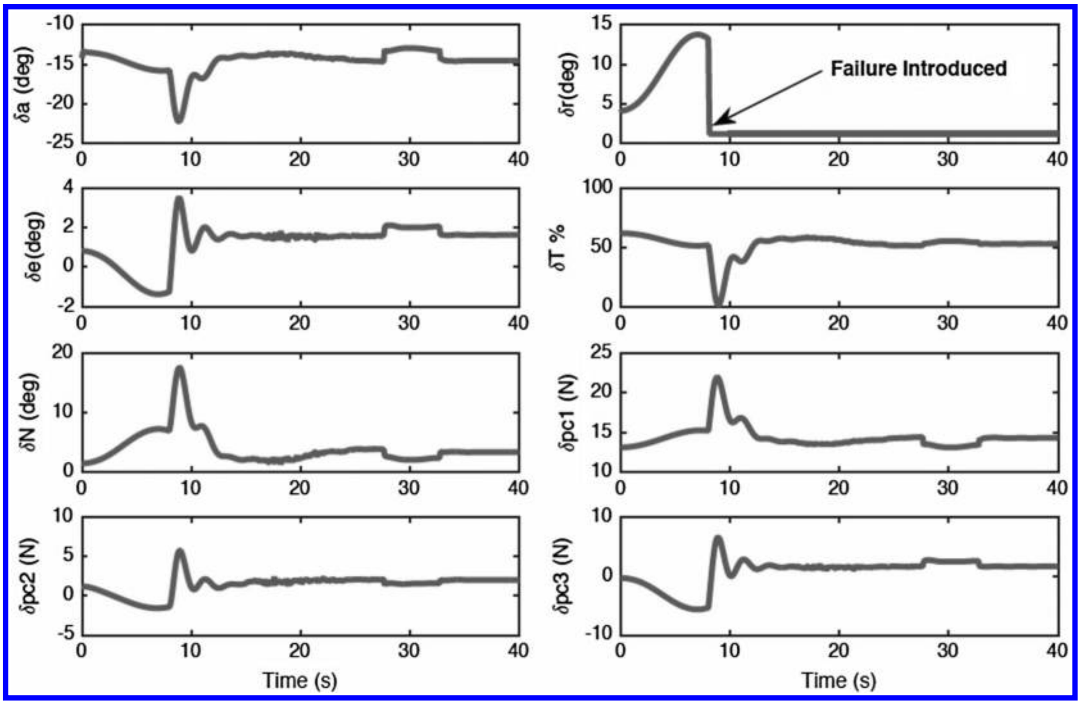

Fig. 4 Test case 1: receiver-aircraft control positions.

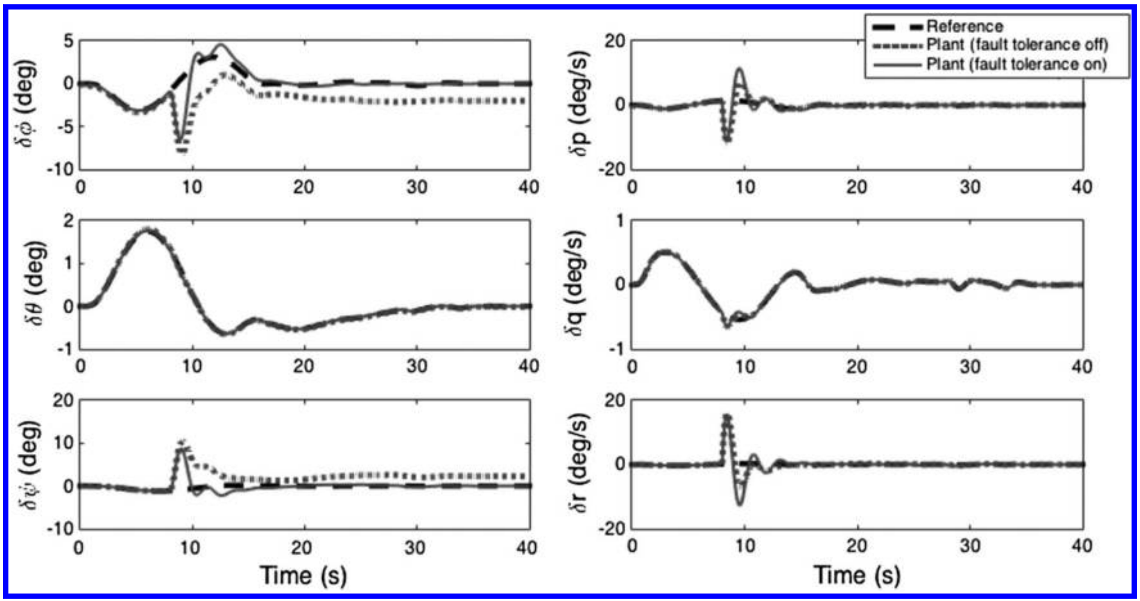

Fig. 5 Test case 1: receiver-aircraft angular states.

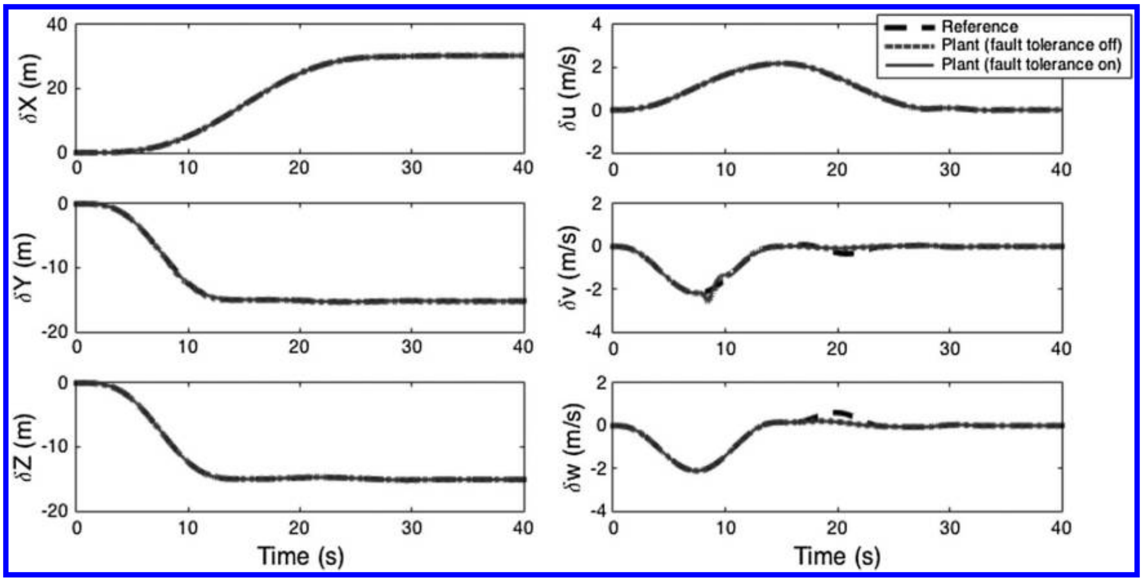

Fig. 6 Test case 1: receiver-aircraft translational states.

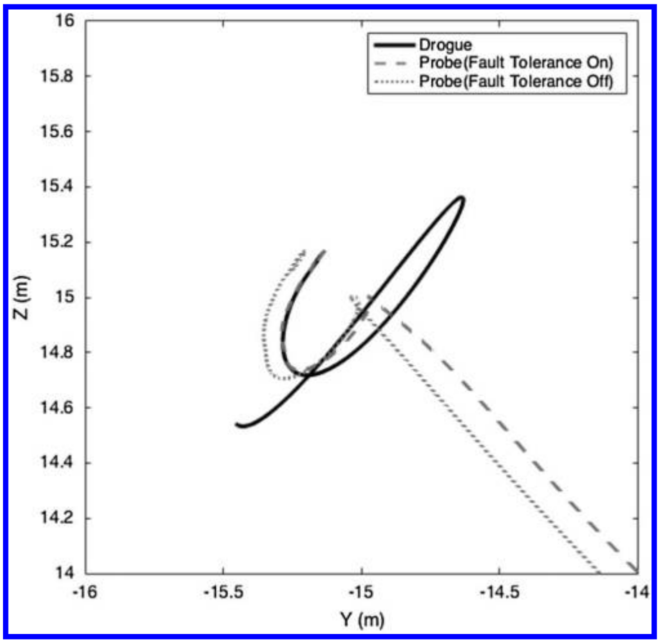

Fig. 7 Test case 1: projection of probe-and-drogue trajectories in Y_n–Z_n plane.

the reference trajectory. In this figure, a comparison is made between the fault-tolerant SAMI controller and the same controller without a fault-tolerance capability, that is, the adaptation is turned off. It is seen that, without a fault-tolerant capability, there is a steady-state error in bank angle and heading angle, which is undesirable.

Figure 6 shows the receiver-aircraft translational positions and velocities. A small error of 1 m/s can be seen in the lateral component of velocity when the failure is introduced at 8 s. Because the reference trajectories for the position-level translational states are very smooth, the error in these states is very small with the adaptive mechanism either on or off, even immediately after the failure is introduced.

Figure 7 shows the probe-and-drogue trajectories in the Y_n–Z_n plane. A docking attempt is considered successful if the probe tip can get within a 0.1 m radius of the refueling port with zero offset angle. It can be seen that, even in the case of a rudder failure, successful docking can be achieved using fault-tolerant SAMI, but if the fault-tolerance mechanism is turned off, the probe is unable to dock.

The adaptive matrices C, D, and E are also observed at the introduction of the failure. Figure 8 shows the elements of the E matrix, Fig. 9 shows the adaptation of the C matrix, and Fig. 10 shows the adaptation of the D matrix. Elements of these matrices adapt according to the adaptive laws derived in Sec. IV, and this helps in handling uncertainties and control failures in the system. It should be noted here that the fault-tolerant SAMI algorithm does not require fault identification at any stage of the refueling process.

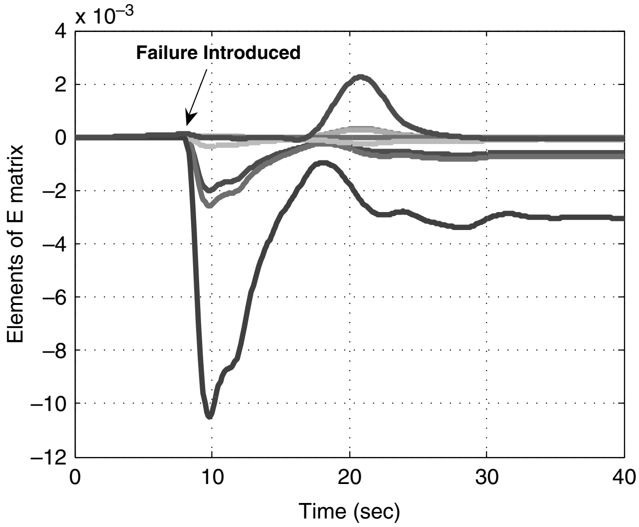

Fig. 8 Test case 1: elements of E matrix.

It is concluded from the results presented in test case 1 that, for the control-effector failure considered, the fault-tolerant SAMI controller can track and successfully dock to a nonstationary drogue with negligible error. All control deflections and states are within tolerable limits for the test case considered. Additionally, the controller did not require or use any knowledge of a control-effector failure.

B. Test Case 2

In test case 2, the elevon fails at 10 s and is locked at 1.2 deg. This failure results in errors, primarily in the longitudinal states, at the introduction of the failure. Relatively quickly, however, the elements of the adaptive matrices reconfigure themselves, and the error between the reference states and the receiver-aircraft states is reduced to zero. Figure 11 shows the elevon locked after 10 s, resulting in a sudden high deflection in the rest of the control surfaces immediately afterward. This figure demonstrates how these types of failures can potentially result in high control-surface deflections in the other working control

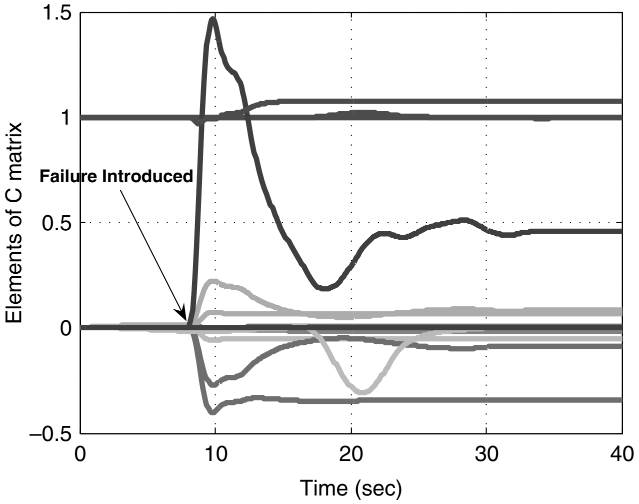

Fig. 9 Test case 1: elements of *C* matrix.

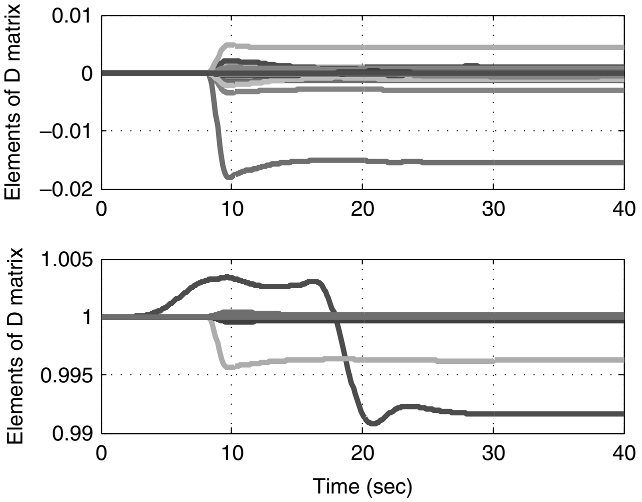

Fig. 10 Test case 1: elements of *D* matrix.

effectors. This may lead to other problems, like saturation or high control rates; however, in this case, all of the control-surface deflections remain within acceptable ranges. This allows for the healthy surfaces to compensate for the failure and provide the desired moments without going out of bounds. Although not demonstrated in this test case, if a large external disturbance, such as wind shear, were to occur while the elevator was locked, this adaptive-control scheme would similarly be able to compensate with the healthy control surfaces so long as the plant remained controllable. If there was a known upper bound on the magnitude of such a disturbance, the response time of the adaptive laws (42) could be adjusted to safely account for any disturbance smaller than that bound, and additional measures could be taken to make sure the control signals remained within saturation limits.

Figure 12 shows the effect of elevator failure on the receiver-aircraft angular states. There is an initial steady-state error in bank angle, which is not due to the introduced failure at 10 s, but instead is due to uncertainty in the system. With the adaptation turned off, this results in a steady-state error. A simple dynamic-inversion-based controller is not able to handle the uncertainties in the system. This emphasizes the importance and need for adaptivity in an aerial-refueling system, even without the presence of control failures. A second steady-state error also develops when the adaptive mechanism turned off — this one in pitch-attitude angle. This error is indeed triggered by the control-surface failure. Figure 13 displays the translational states for test case 2. Once again, the reference trajectories for the linear states are relatively smooth, and therefore any deviation from the reference trajectory is negligible.

Figure 14 shows the probe-and-drogue trajectories in the Y_n–Z_n plane. Although closer than in case 1, it is shown that, with fault-tolerance adaptation turned off, the probe cannot successfully track and dock with the drogue in the presence of the elevon failure. As in case 1, with the fault-tolerance adaptation turned on, a successful docking occurs.

Test case 2 demonstrates that, even in the case of uncertainties and control-effector failures, the controller is able to achieve successful docking. The transient response at the introduction of the failure initially produces large angular body rates, but they reduce to zero within a few seconds.

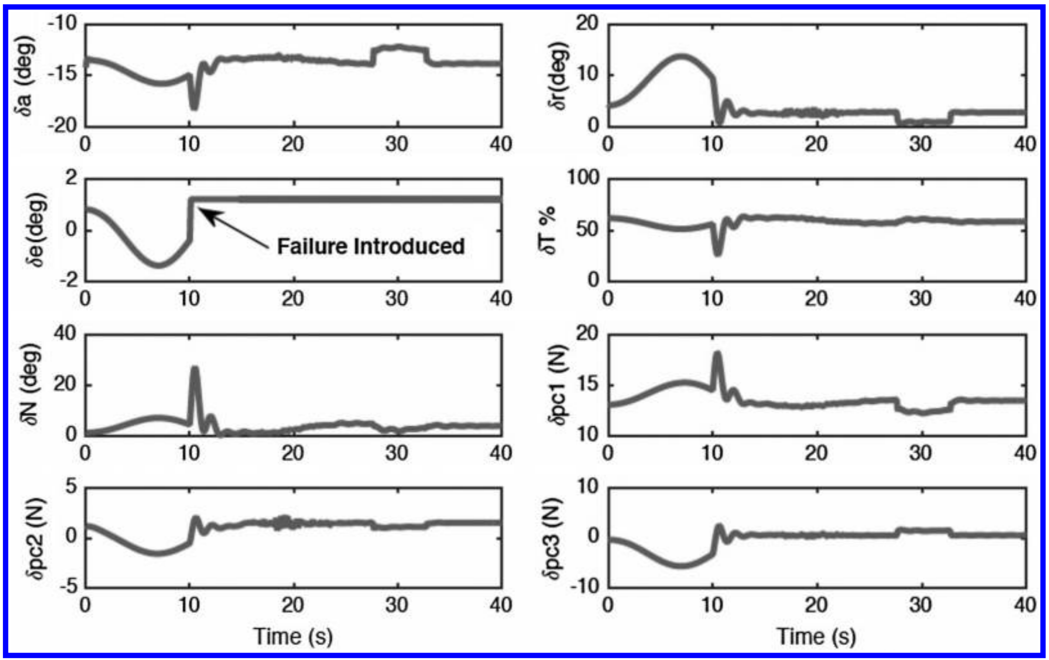

Fig. 11 Test case 2: receiver-aircraft control positions.

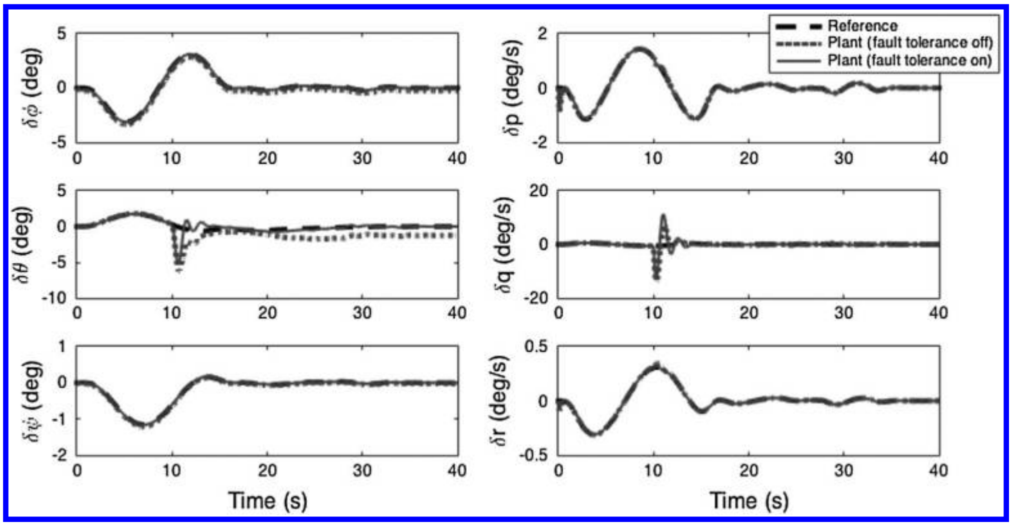

Fig. 12 Test case 2: receiver-aircraft angular states.

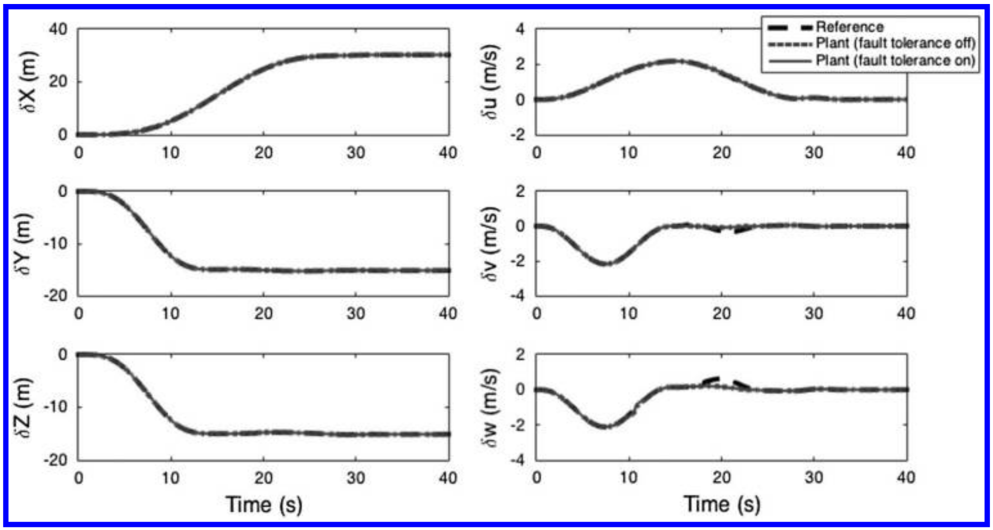

Fig. 13 Test case 2: receiver-aircraft translational states.

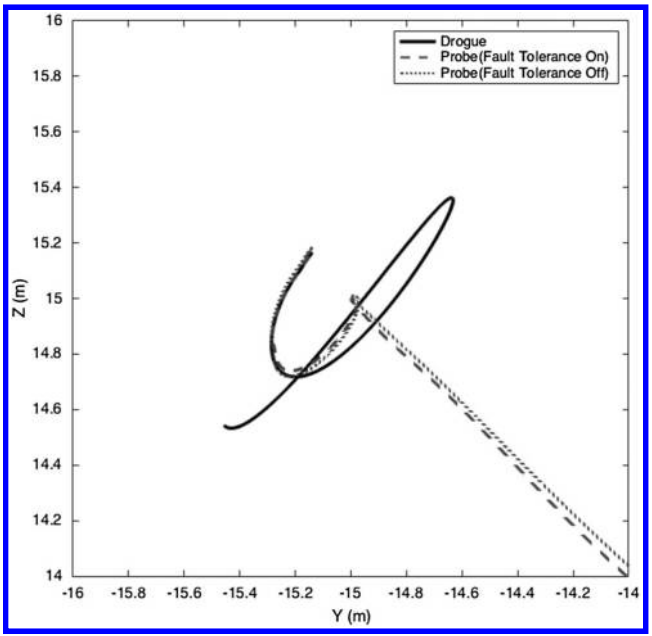

Fig. 14 Test case 2: projection of probe-and-drogue trajectories in Y_n–Z_n plane.

VII. Conclusions

This paper developed a fault-tolerant structured adaptive model inversion tracking controller and integrated it with the visual navigation vision-based sensor system to provide fully autonomous aerial refueling for the probe-and-drogue method. Smooth reference trajectories from the initial position of the receiver aircraft to the position of a moving drogue were generated, and a nonlinear simulation of the UCAV6 receiver aircraft was used to test the fault-tolerance capability of the controller. Failures in the form of control surfaces locked at nonneutral positions were investigated. Based upon the results presented in the paper, it is concluded that the controller is able to successfully steer the probe into docking position with the drogue in the presence of control-effector failures and system parametric uncertainties. The maximum angular rate experienced in the presence of control-effector failures is 16 deg/s, and the maximum error in rotational angles is 10 deg. The maximum error in translational velocity is 1 m/s, which shows that translational states are not strongly affected by the control-effector failures investigated. The results also show that, for a simple dynamic-inversion-based controller with adaptation turned off, uncertainties in the system produce steady-state errors. This demonstrates the utility of adaptive-control approaches for problems that use model-based controllers.

Appendix: Receiver-Aircraft Model

The stability and control derivatives are given in the following Tables A1 and A2.

Table A1 Stability derivatives

Derivative	Value	Derivative	Value
X_u	−0.04599	Y_v	−0.40773
X_w	0.17588	Y_p	0.00565
X_q	−0.00565	Y_r	3.58484
Z_u	−0.08302	L_v	−0.02178
Z_w	−1.76159	L_p	−4.76282
Z_q	−13.57715	L_r	2.16780
M_u	0.00254	N_v	0.02478
M_w	−0.07165	N_p	−0.22500
M_q	−0.81723	N_r	−1.56983

Table A2 Control derivatives

Derivative	Value	Derivative	Value
$X_{\delta e}$	0.00563	$Y_{\delta a}$	−0.37624
$X_{\delta T}$	0.32974	$Y_{\delta r}$	0.56176
$X_{\delta N}$	0.07386	$Y_{\delta \mathrm{pc2}}$	2.397
$X_{\delta \mathrm{pc1}}$	2.397	$L_{\delta a}$	0.64103
$Z_{\delta e}$	0.35083	$L_{\delta r}$	0.08855
$Z_{\delta T}$	0.33081	$L_{\delta \mathrm{pc1}}$	0.6088
$Z_{\delta N}$	−0.00533	$L_{\delta \mathrm{pc2}}$	0.2435
$Z_{\delta \mathrm{pc3}}$	2.397	$N_{\delta a}$	0.03034
$M_{\delta e}$	0.14704	$N_{\delta r}$	−0.10738
$M_{\delta T}$	0.01350	$N_{\delta \mathrm{pc1}}$	0.0603
$M_{\delta N}$	0.02735	$N_{\delta \mathrm{pc3}}$	−0.1842
$M_{\delta \mathrm{pc3}}$	0.3846		

The estimated A and B matrices, A_{est} and B_{est}, used in the update laws are

$$A_{\mathrm{est}} = \begin{bmatrix} \frac{1}{I_1}(0.8L_v v + 0.85L_p p + L_r r + (I_2 - I_3)qr) \\ \frac{1}{I_2}(M_u u + 0.9M_w w + M_q q + (I_3 - I_1)pr) \\ \frac{1}{I_3}(0.85N_v v + N_p p + N_r r + (I_1 - I_2)pq) \\ X_u u + X_w w + X_q q - g\cos(\theta)\cos(\phi) + rv - qw \\ Y_v v + Y_p p + Y_r r + g\cos(\theta)\sin(\phi) + pw - ru \\ Z_u u + 0.95Z_w w + Z_q q - g\sin(\theta) + qu - pv \end{bmatrix}$$

$$B_{\mathrm{est}} = \begin{bmatrix} 0.95L_{\delta a} & 0.95L_{\delta r} & 0 & 0 & 0 & L_{\delta_{pc1}} & L_{\delta_{pc2}} & 0 \\ 0 & 0 & 0.95M_{\delta e} & 0.95M_{\delta T} & 0.95M_{\delta N} & 0 & 0 & M_{\delta_{pc3}} \\ 0.95N_{\delta a} & 0.95N_{\delta r} & 0 & 0 & 0 & N_{\delta_{pc1}} & 0 & N_{\delta_{pc3}} \\ 0 & 0 & 0.95X_{\delta e} & X_{\delta T} & X_{\delta N} & X_{\delta_{pc1}} & 0 & 0 \\ Y_{\delta a} & Y_{\delta r} & 0 & 0 & 0 & 0 & Y_{\delta_{pc2}} & 0 \\ 0 & 0 & Z_{\delta e} & Z_{\delta T} & Z_{\delta N} & 0 & 0 & Z_{\delta_{pc3}} \end{bmatrix}$$

Acknowledgments

The authors wish to acknowledge the support of NASA Johnson Space Center under award number C08-00223. The technical monitor is Steven Fitzgerald. This work was also sponsored (in part) by the U.S. Air Force Office of Scientific Research, U.S. Air Force, under grant/contract number FA9550-08-1-0038. The technical monitor is William M. McEneaney. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the NASA or the U.S. Air Force Office of Scientific Research, or the U.S. Government.

References

[1] Valasek, J., Gunnam, K., Kimmett, J., Tandale, M. D., Junkins, J. L., and Hughes, D., "Vision-Based Sensor and Navigation System for Autonomous Air Refueling," *Journal of Guidance, Control, and Dynamics*, Vol. 28, No. 5, Sept.–Oct. 2005, pp. 979–989. doi:10.2514/1.11934

[2] Nalepka, J. P., and Hinchman, J. L., "Automated Aerial Refueling: Extending the Effectiveness of Unmanned Air Vehicles," *AIAA Modeling and Simulation Technologies Conference and Exhibit*, AIAA Paper 2005-6005, Aug. 2005.

[3] Doebbler, J., Spaeth, T., Valasek, J., Monda, M. J., and Schaub, H., "Boom and Receptacle Autonomous Air Refueling Using a Visual Pressure Snake Optical Sensor," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 6, Nov.–Dec. 2007, pp. 1753–1769. doi:10.2514/1.28305

[4] Lachapelle, G., Sun, H., Cannon, M. E., and Lu, G., "Precise Aircraft-to-Aircraft Positioning Using a Multiple Receiver Configuration," *Proceedings of the National Technical Meeting*, Institute of Navigation, Alexandria, VA, 1994, pp. 793–799.

[5] Tandale, M. D., Bowers, R., and Valasek, J., "Trajectory Tracking Controller for Vision-Based Probe and Drogue Autonomous Aerial Refueling," *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 4, July–Aug. 2006, pp. 846–857. doi:10.2514/1.19694

[6] Stepanyan, V., Lavretsky, E., and Hovakimyan, N., "Aerial Refueling Autopilot Design Methodology: Application to F-16 Aircraft Model," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, AIAA Paper 2004-5321, Aug. 2004.

[7] Ochi, Y., and Kominami, T., "Flight Control for Automatic Aerial Refueling via PNG and LOS Angle Control," *AIAA Guidance Navigation and Control Conference and Exhibit*, AIAA Paper 2005-6268, Aug. 2005.

[8] Pachter, M., Houpis, C. H., and Trosen, D. W., "Design of an Air-to-Air Automatic Refueling Flight Control System Using Quantitative Feedback Theory," *International Journal of Robust and Nonlinear Control*, Vol. 7, No. 6, 1997, pp. 561–580. doi:10.1002/(ISSN)1099-1239

[9] Pollini, L., Innocenti, M., and Mati, R., "Vision Algorithms for Formation Flight and Aerial Refueling with Optimal Marker Labeling," *AIAA Modeling and Simulation Technologies Conference and Exhibit*, AIAA Paper 2005-6005, Aug. 2005.

[10] Dogan, A., Sato, S., and Blake, W., "Flight Control and Simulation for Aerial Refueling," *AIAA Guidance Navigation and Control Conference and Exhibit*, AIAA Paper 2005-6264, Aug. 2005.

[11] Wang, J., Patel, V. V., Cao, C., Hovakimyan, N., and Lavretsky, E., "Novel L1 Adaptive Control Methodology for Aerial Refueling with

Guaranteed Transient Performance," *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 1, Jan.–Feb. 2008, pp. 182–193. doi:10.2514/1.31199

[12] Wang, J., Patel, V. V., Cao, C., Hovakimyan, N., and Lavretsky, E., "Verifiable L1 Adaptive Controller for Aerial Refueling," *AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2007-6313, Aug. 2007.

[13] Kimmett, J., Valasek, J., and Junkins, J. L., "Vision Based Controller for Autonomous Aerial Refueling," *Proceedings of the IEEE Control Systems Society Conference on Control Applications*, CCA02-CCAREG-1126, Sept. 2002.

[14] Valasek, J., and Junkins, J. L., "Intelligent Control Systems and Vision Based Navigation to Enable Autonomous Aerial Refueling of UAVs," *27th Annual AAS Guidance and Control Conference*, AAS Paper 04-012, Feb. 2004.

[15] Subbarao, K., and Junkins, J. L., "Structured Model Reference Adaptive Control of a Class of Nonlinear Systems," *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 4, July–Aug. 2003, pp. 551–557. doi:10.2514/2.5104

[16] Marwaha, M., and Valasek, J., "Fault Tolerant Control Allocation for Mars Entry Vehicle Using Adaptive Control," *International Journal of Adaptive Control and Signal Processing*, Vol. 25, No. 2, Feb. 2011, pp. 95–113. doi:10.1002/acs.v25.2

[17] Tandale, M., and Valasek, J., "Fault Tolerant Structured Adaptive Model Inversion Control," *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 3, May–June 2006, pp. 635–642. doi:10.2514/1.15244

[18] Ioannou, P. A., and Sun, J., *Robust Adaptive Control*, Prentice–Hall, Upper Saddle River, NJ, 1996, Chaps. 3, 6.