JOURNAL OF AEROSPACE COMPUTING, INFORMATION, AND COMMUNICATION
Vol. 6, October 2009

Architecture for Harmonizing Manual and Automatic Flight Controls

Eri Itoh*
Electronic Navigation Research Institute, Tokyo, Japan

and

Shinji Suzuki†
University of Tokyo, Tokyo, Japan

DOI: 10.2514/1.39829

Automatic flight control systems occasionally confuse pilots. A pilot should shift to manual control if necessary in an emergency situation. However, it is often difficult for a pilot to make a decision to disengage the autopilot. Against this background, we propose the "human as a control module" architecture for harmonizing pilot and autopilot control. The human as a control module architecture adjusts pilot and autopilot control authorities automatically when simultaneous inputs to the aircraft are given. Avoiding the overlap of pilot and autopilot inputs, the proposed architecture helps to circumvent the effect of conflicting actions. This paper culminates in the ultimate purpose of the human as a control module architecture and demonstrates how it could improve aircraft safety by applying it to a past aircraft incident.

Nomenclature

E	error index
i	number of modules
j	number of outputs
m	number of time steps of past tracking errors
n	number of time steps at the present time
r	target value
t, t_n	present time
u	input vector to aircraft from the ARBITER module
V	airspeed
V_{mo}	maximum limitation of airspeed
x	input vector from each module
y	simulated output vector corresponding to each module
z	vector of modification signal

Received 16 July 2008; accepted for publication 16 June 2009. Copyright © 2009 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved. Copies of this paper may be made for personal or internal use, on condition that the copier pay the $10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 1542-9423/09 $10.00 in correspondence with the CCC.

* PhD, Research Scientist, Air Traffic Management Department, Electronic Navigation Research Institute ENRI, 7-42-23, Jindaiji-higashi, Chofu, Tokyo 182-0012, Japan, eri@enri.go.jp, AIAA Member.

† Professor, Department of Aeronautics and Astronautics, the University of Tokyo, 7-3-1, Hongo, Bunkyo-ku, Tokyo 113-8565, Japan, tshinji@mail.ecc.u-tokyo.ac.jp, AIAA Senior Member.

Δ	modeling error
ε	error value between simulated output and target value of output
θ	pitch angle
λ	contribution ratio
$\dot{\omega}$	vertical acceleration
$\ddot{\omega}$	change in vertical acceleration

I. Introduction

PILOTS have reported that aircraft under autopilot control can behave unpredictably [1,2]. Under automatic flight control, pilot uncertainty in aircraft response is increased when there is not enough time to monitor the cockpit displays. This uncertainty has the potential to trigger disasters [1–3]. Furthermore, it may compel pilots to disengage the autopilot and switch to manual control. The decision to disengage the autopilot and when to do it can be difficult. The sudden transfer to manual control triggers pilot-induced oscillation (PIO), which is well known as a control feedback problem.

Against this background, this research focuses on conflicts between pilot and autopilot in the following two situations: (1) pilot and autopilot have different goals and (2) pilot or autopilot control abilities are exceeded. As an example of the former situation, in 1994 the pilot of an A300 attempted to land at Nagoya airport without knowing that the autopilot was in "take off, go around" mode [4–6]. The interference between the pilot and autopilot control inputs caused the aircraft to crash. After the crash, flight control systems were improved to allow pilots to override the autopilot. Several examples of the latter situation have been documented for MD11 accidents caused by PIO [4,5,7]. Specifically, in 1997 an MD11 flying on autopilot met with atmospheric turbulence [7]. The autopilot was disengaged when the pilot believed that manual control would be more effective at countering the effects of turbulence. The same type of incident also arose on a B747-400 in 2002 [8].

There is a significant body of literature on the above "automation surprises" problems with mode confusion and how to prevent them [9–11]. Rushby provided a tool to examine a pilot's mental model using "mechanized calculation" [9]. Joshi et al. [10] analyzed potential sources of mode confusion. Vakil and Hansman [11] presented an overview of problems in modern cockpits. While issues found during operational experience are conventionally resolved through training and procedure modifications, they claimed that the design process of automation systems should be changed.

Autopilots are designed according to two primary goals. The first goal is to avoid human errors by eliminating the need for pilot-controlled maneuvering. The second goal is to implement human-centered automation and thus provide manual override capability. In this research, we propose the "human as a control module (HACM) architecture" [12–15] that combines the advantages of manual and purely automatic control. The HACM architecture simulates the aircraft motion when pilot and autopilot simultaneously control the aircraft. Both pilot and autopilot performance are evaluated by the past and a one time-step prediction based on the simulation results. If the pilot manual control causes an unsafe aircraft motion, for example by a sudden change of acceleration, the HACM architecture smoothly transfers the control authority to the autopilot by using a soft-max function. Past studies on a modular structure [16–20] work on bumpless transfer to and from motor control in the human brain. Wolpert and Kawato [18] and Haruno et al. [20] proposed a modular structure termed "MOdular Selection And Identification for Control (MOSAIC)" in which a soft-max function is employed to switch modules.

The concepts and algorithms of the HACM architecture have been developed in several past studies [12–15]. The initial ideas have been evaluated through manual tracking experiments [12]. While oscillation was observed when pilot control was transferred to and from the autopilot, the algorithms have been improved to evaluate the past performance of pilot and autopilot. The improved HACM architecture has then been applied to PIO avoidance [13,14]. The robustness was discussed when the architecture worked under an error between real wind and wind data estimated by an extended Kalman filter [15]. It indicated that further modifications of feedback signals protect the autopilot performance. In the initial concepts, pilot and autopilot control were combined. However, the past studies suggested changes of the design concepts in order to put priority on pilot control while it is within safety constraints. This paper reflects these findings and culminates in the ultimate purpose of the HACM architecture.

This paper consists of four parts. First, we summarize the design concepts and structures of the HACM architecture. Second, one of the past aircraft incidents is picked up. Based on the analysis of autopilot design related to the incident factors, the incident situation is reconstructed. Third, the HACM architecture is applied to the incident to confirm its effectiveness. Finally, ideas to avoid the effects of automation surprises are discussed.

II. Human As a Control Module Architecture

A. Concept

Past aircraft accidents/incidents were often induced when the pilot switched from autopilot control to manual control. A pilot has great ability to assess a situation by changing control characteristics according to the environment. On the other hand, a pilot's physiological ability is limited. There are also possibilities that give rise to human error. The autopilot will control an aircraft appropriately within the design conditions. However, the autopilot is not designed to control an aircraft in emergency situations such as when it encounters strong atmospheric turbulence and failure events. Since both autopilot and pilot have advantages and disadvantages in aircraft control, it is difficult to judge in advance of any emergent situation whether control authority should be given to pilot or autopilot. For this reason, this research proposes a new concept for automation design: HACM architecture [12–15], which blends pilot maneuver with autopilot control instead of switching between them.

Figure 1 shows a schematic diagram representing the HACM architecture. The pilot is treated as one of the control elements. The architecture consists of three control elements: the HUMAN module, CONTROLLER module, and ARBITER module. The HUMAN module corresponds to a pilot, the CONTROLLER module corresponds to an autopilot, and the ARBITER module works to harmonize the CONTROLLER module and the HUMAN module. As shown in Fig. 1, the HUMAN and CONTROLLER modules are arranged in parallel to construct a modular structure. Both of them generate control commands at the same time. The ARBITER module takes these control commands as input, which it then adjusts and harmonizes.

B. ARBITER Mechanism

The ARBITER module simulates aircraft movements corresponding to each control of the HUMAN and CONTROLLER modules. Then it generates control commands based on the simulated results. Through the ARBITER module, the HACM architecture avoids the overlapping of pilot and autopilot full control by adaptively adjusting weights given to their controls. When a pilot manually controls the aircraft safely, the ARBITER module works

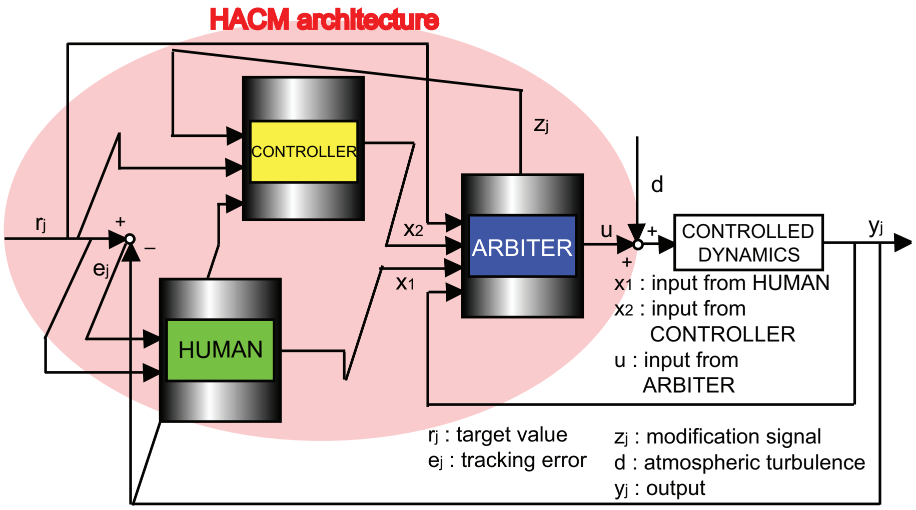

Fig. 1 HACM architecture.

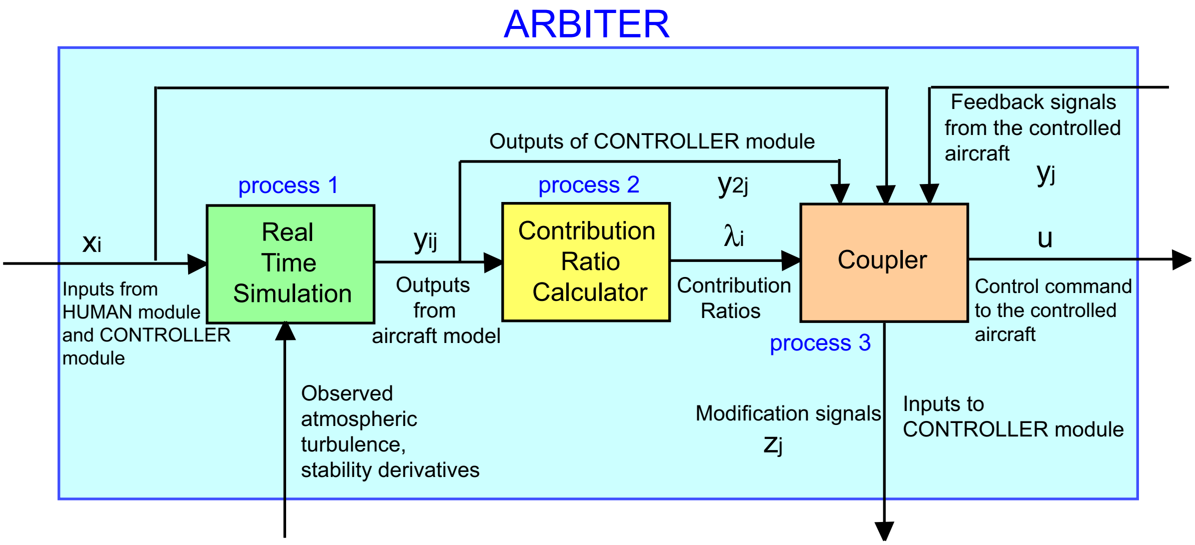
ARBITER
process 1
Real Time Simulation
process 2
Contribution Ratio Calculator
process 3
Coupler
xi
Inputs from HUMAN module and CONTROLLER module
yij
Outputs from aircraft model
Outputs of CONTROLLER module
y2j
λi
Contribution Ratios
Feedback signals from the controlled aircraft
yj
u
Control command to the controlled aircraft
Observed atmospheric turbulence, stability derivatives
Modification signals
zj
Inputs to CONTROLLER module

Fig. 2 The ARBITER module.

to prevent autopilot interference and protects pilot control. On the other hand, if the pilot induces unsafe aircraft movement, the ARBITER blends autopilot control.

Figure 2 shows the mechanism of the ARBITER module. Control commands from the HUMAN module $x_1(t)$ and the CONTROLLER module $x_2(t)$ are input to the ARBITER module. The general mechanism in the ARBITER module is comprised of the following three processes: a real-time simulation, contribution ratio calculator, and coupler. Figure 3 shows calculation flow in the ARBITER module.

1. *Real-time Simulation*

In the first step, the ARBITER module simulates aircraft motion corresponding to both controls of the HUMAN and CONTROLLER modules in real time. The ARBITER possesses a dynamic model of the controlled aircraft within its framework. By using this dynamic model, the outputs $y_{ij}(t)$ $(i = 1, 2;\ j = 1, 2, \ldots, l)$ due to inputs $x_i(t)$ are numerically simulated. It is noted that $y_{1j}(t)$ and $y_{2j}(t)$ are the corresponding outputs of the HUMAN and CONTROLLER modules. Atmospheric turbulence should be considered in the simulation. The output $y_{ij}(t)$ is calculated by Euler integration, which is initialized by the previous simulation step.

2. *Contribution Ratio Calculator*

In the second step, contribution ratios $\lambda_i(t)$ are calculated by using the outputs $y_{ij}(t)$. The terms $\lambda_1(t)$ and $\lambda_2(t)$ are contribution ratios for the HUMAN and CONTROLLER modules, respectively. A contribution ratio represents the weighting factor to combine each input.

In order to calculate the contribution ratio, the performance of the HUMAN and CONTROLLER modules are evaluated individually. The performance of each control is measured following the index

$$E_{ij}(t) = \frac{\sum_{k=n-m}^{n} (\varepsilon_{ij}(t_k))^2 e^{(k-n+m)/m}}{\sum_{k=n-m}^{n} e^{(k-n+m)/m}} \tag{1}$$

where $\varepsilon_{ij}(t)$ is the error between $y_{ij}(t)$ and the target values, which are the desired outputs of the aircraft at time t; n is the number of time steps at present time t; and m is the number of past time steps in which past tracking errors are considered.

In order to calculate $\varepsilon_{ij}(t)$ in Eq. (1), a concept similar to flight envelope protection is applied. The ARBITER module adaptively adjusts the control authority when the HUMAN and CONTROLLER modules simultaneously control the aircraft and at least the HUMAN module violates the defined flight envelope. In this paper, the HACM architecture is applied for longitudinal control of the aircraft. The flight envelope is defined as follows

$$\begin{aligned}\theta_{\min} \leqslant \theta_i(t) \leqslant \theta_{\max}\\ \dot{\omega}_{\min} \leqslant \dot{\omega}_i(t) \leqslant \dot{\omega}_{\max}\\ \ddot{\omega}_{\min} \leqslant \ddot{\omega}_i(t) \leqslant \ddot{\omega}_{\max}\end{aligned} \tag{2}$$

where $\theta_i(t)$, $\dot{\omega}_i(t)$, and $\ddot{\omega}_i(t)$ are the outputs of the aircraft model corresponding to input from the HUMAN and CONTROLLER modules calculated in the ARBITER module. As shown in Eq. (2), the maximum and minimum limitations are introduced for pitch angle, vertical acceleration, and rate of vertical acceleration. The limitation of $\ddot{\omega}$ is the rate limitation of vertical acceleration in the speed control mode of the autopilot in the distressed aircraft.

The output errors $\varepsilon_{ij}(t)$ are defined as follows

$$\text{If } y_{ij}(t) < y_{j\,\min},\ \text{then } \varepsilon_{ij}(t) = \frac{y_{ij}(t) - y_{j\,\min}}{y_{j\,\min}} \tag{3}$$

$$\text{If } y_{j\,\max} < y_{ij}(t),\ \text{then } \varepsilon_{ij}(t) = \frac{y_{j\,\max} - y_{ij}(t)}{y_{j\,\max}} \tag{4}$$

where

$$\begin{aligned}[y_i(t)]^{\mathrm{T}}_{(j)}\ (i = 1, 2;\ j = 1, 2, 3) &= [y_{i1}(t)\ \ y_{i2}(t)\ \ y_{i3}(t)]^{\mathrm{T}}\\ &= [\theta_i(t)\ \ \dot{\omega}_i(t)\ \ \ddot{\omega}_i(t)]^{\mathrm{T}}\end{aligned} \tag{5}$$

If $y_{j\,\min} \leqslant y_{ij}(t) \leqslant y_{j\,\max}$, then $\varepsilon_{ij}(t) = 0$.

The contribution ratios of each module $\lambda_i(t)$ are calculated by using the soft-max function. The contribution ratios are given as follows

$$\lambda_i(t) = \frac{\sum_{j=1}^{l} e^{-(E_{ij}(t)/\sigma)}}{\sum_{i=1}^{2}\sum_{j=1}^{l} e^{-(E_{ij}(t)/\sigma)}} \tag{6}$$

where σ is a scaling constant and e stands for the exponential function. The soft-max function normalizes the tracking errors across the modules so that the contribution ratios lie between 0 and 1. The sum of the contributions of each module is 1. Figure 3 shows how the contribution ratios are calculated. When $y_{1j}(t)$ is within the flight envelope, $E_{ij}(t)$ is not calculated, and $\lambda_1(t) = 1.0$ and $\lambda_2(t) = 0.0$ are given. This means the HUMAN module has full authority to control the aircraft. When $y_{1j}(t)$ is outside the flight envelope and $y_{2j}(t)$ is within the flight envelope, $\varepsilon_{2j}(t) = 0.0$ is given in Eq. (1). In the case that both $y_{1j}(t)$ and $y_{2j}(t)$ break the flight envelope, $\varepsilon_{ij}(t)$ is calculated in Eqs. (3) and (4) to adjust the value of $\lambda_i(t)$. The ARBITER works to reduce deviation from the defined minimum/maximum values in the flight envelope.

3. *Coupler*

In the final step, the control commands of the HUMAN and CONTROLLER modules are adjusted. The control command to the controlled aircraft $u(t)$ is given as follows

$$u(t) = \sum_{i=1}^{2} \lambda_i(t)x_i(t) \tag{7}$$

This implies that the HUMAN and CONTROLLER modules control are combined according to the contribution ratios.

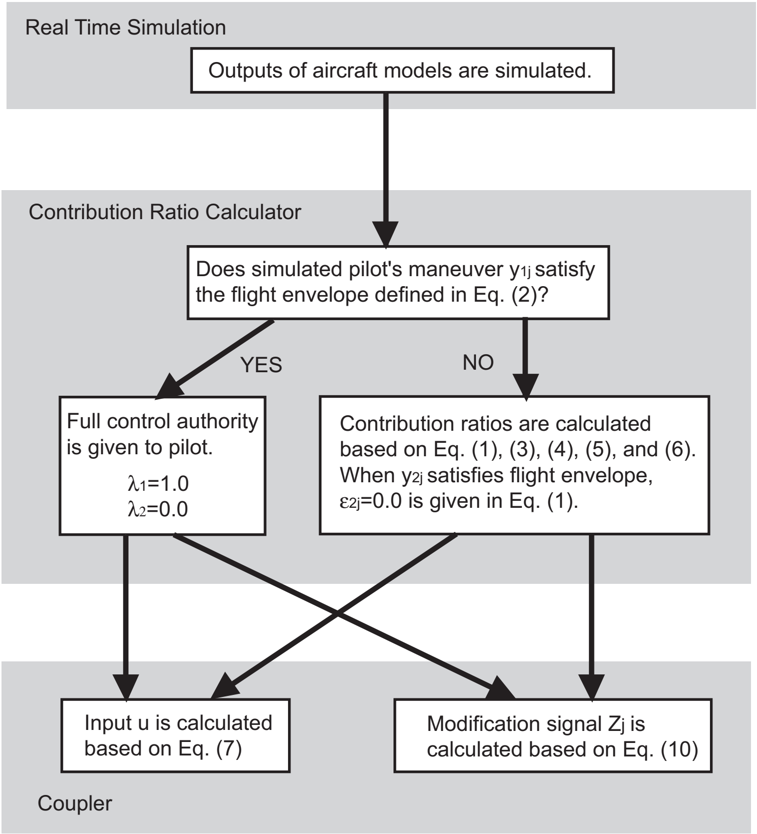
Real Time Simulation
Outputs of aircraft models are simulated.
Contribution Ratio Calculator
Does simulated pilot's maneuver y1j satisfy the flight envelope defined in Eq. (2)?
YES
NO
Full control authority is given to pilot.
λ1=1.0
λ2=0.0
Contribution ratios are calculated based on Eq. (1), (3), (4), (5), and (6). When y2j satisfies flight envelope, ε2j=0.0 is given in Eq. (1).
Input u is calculated based on Eq. (7)
Modification signal Zj is calculated based on Eq. (10)
Coupler

Fig. 3 Computational flow in the ARBITER module.

Another function of this process is to modify feedback signals given to the CONTROLLER module. The output of the controlled aircraft$y_j(t)$ is described as follows

$$\begin{aligned} y_j(t) &= \lambda_1(t)y_{1j}(t) + \lambda_2(t)y_{2j}(t) + \Delta_j(t) \\ &= \{\lambda_1(t) + \lambda_2(t)\}y_{2j}(t) - \lambda_1(t)y_{2j}(t) + \lambda_1(t)y_{1j}(t) + \Delta_j(t) \\ &= y_{2j}(t) + \lambda_1(t)\{y_{1j}(t) - y_{2j}(t)\} + \Delta_j(t) \end{aligned} \tag{8}$$

where $\Delta_j(t)$ shows the modeling error. Here we consider $\Delta_j(t) \cong 0$. When the CONTROLLER module uses $y_j(t)$ as a feedback signal, the error value between the target value $r_j(t)$ and $y_j(t)$, which the controller minimizes, is given by using Eq. (8)

$$\begin{aligned} r_j(t) - y_j(t) &= r_j(t) - [y_{2j}(t) + \lambda_1(t)\{y_{1j}(t) - y_{2j}(t)\}] \\ &= [r_j(t) - \lambda_1(t)\{y_{1j}(t) - y_{2j}(t)\}] - y_{2j}(t) \\ &= r'_j(t) - y_{2j}(t) \end{aligned} \tag{9}$$

Equation (9) shows that the CONTROLLER module works to approximate $y_{2j}(t)$ to $r'_j(t)$, not to $r_j(t)$. In order to avoid this problem, the coupler generates a modification signal $z_j(t)$, which is added to the error values between the target signals and the feedback signals.

$$z_j(t) = \lambda_1(t)\{y_{1j}(t) - y_{2j}(t)\} \tag{10}$$

Since the modification signal described in Eq. (10) is added to Eq. (9), the CONTROLLER works to approximate the output $y_{2j}(t)$ to the target value $r_j(t)$.

III. Application to a Past Aircraft Incident

A. Past Aircraft Incident

In 2002, a B747-400 flying at around 40,000 ft with autopilot in Japanese airspace met with atmospheric turbulence [8]. The airspeed suddenly increased due to a strong head wind. The autopilot changed the flight mode from altitude control to speed control, which controls airspeed by pitch angle. While the pitch angle was increased, airspeed approached V_{mo} (the maximum limitation of the airspeed). Then, either the pilot shifted to manual control consciously or the autopilot was disengaged automatically (it is not clear from the report). After disengaging the autopilot, pitch angle oscillation was caused by the pilot's elevator control. Because of the substantial change in vertical acceleration, four people were seriously injured and 29 people were slightly injured. A part of the aircraft was also damaged as a result.

The accident analysis report published in 2006 [8] concluded that the autopilot and the aircraft had normally worked without failure. The pilot had enough experience of controlling B747-400 aircraft with more than 7250 flight hours. Atmospheric turbulence was not observed before the flight.

B. Reconstruction of the Incident Situation

Based on the official data in the accident analysis report, flight data such as wind speed and direction, elevator angle, yaw angle, and roll angle have been extracted to reconstruct the incident situation. Since the dynamics of the B747-400 have not officially been published, our simulations are based on the nonlinear dynamics of a B747-100 flying at 40,000 ft with a cruising speed of 871 ft/s. The nonlinear equations for the aircraft and the parameters utilized in the simulation are summarized in Appendix.

Since the details of autopilot design are not disclosed, we design an autopilot that captures the characteristics of the autopilot covered in the report. The report points out the following two design characteristics of the autopilot to explain the reasons why the autopilot could not reduce the airspeed by pitch angle control.

1. *Design Characteristic 1*

The autopilot was designed to control elevator angle within the limitation of the change in vertical acceleration caused by airspeed, horizontal and vertical acceleration, and change in pitch angle. The limitation of vertical acceleration change was set within $\pm 0.15\,G$. This value was not changed when the airspeed was increased to around V_{mo}. Consequently, rapid change in airspeed was not achieved.

2. *Design Characteristic 2*

Because the autopilot controlled elevator angle by using filtered airspeed in order to eliminate high frequency noise, there was a time delay in changing controlled airspeed.

A block diagram of the designed autopilot is shown in Fig. 4. The autopilot generates an elevator command by using the error value of the target airspeed V_c and feedback of filtered airspeed V. It consists of a coupler, filters, a PID controller, and feedback gains. Design parameters in the control system are adjusted to satisfy the above design characteristics. Details of the autopilot design are summarized in Appendix.

Forty seconds of flight including the sudden change in vertical acceleration were simulated. The autopilot worked to hold the altitude until 18 s. Then, it changed mode from altitude hold to speed control. Altitude hold mode was realized by giving the target pitch angle θ_c as 0 as in Fig. 4. After 26 s, the autopilot was disengaged in the simulation and then the pilot controlled the aircraft manually. The accident analysis report does not give an accurate time for when the autopilot was disengaged. It is also uncertain whether the pilot himself disengaged the autopilot or not. This paper makes the assumption that the pilot disengaged the autopilot and switched to manual control at 26 s. The elevator angle, as controlled by the pilot from 26 s, was extracted from the reported data. Additionally the elevator angle was adjusted to realize the vertical acceleration and pitch angle described in the aircraft accident report. Since the aircraft model used in the simulation was different from the actual aircraft, adjustment of the elevator angle was necessary to reconstruct the incident situation. The wind data following the x, y directions of the earth axis are shown in Figs. 5 and 6.

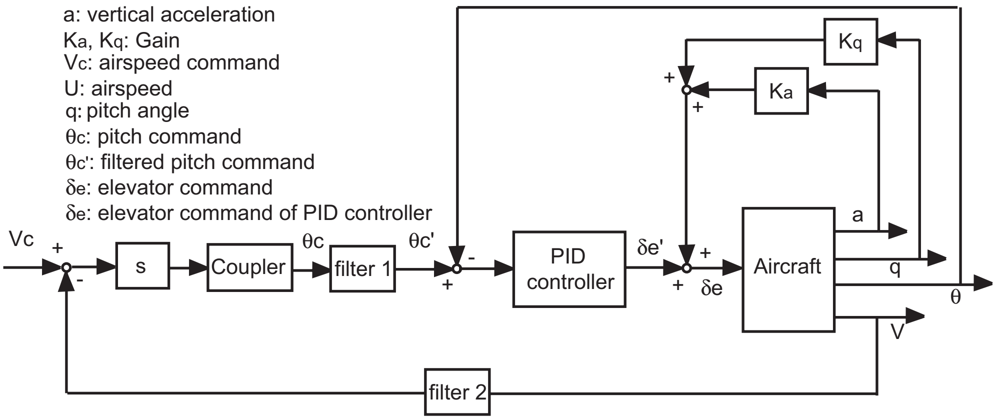

Fig. 4 Autopilot design.

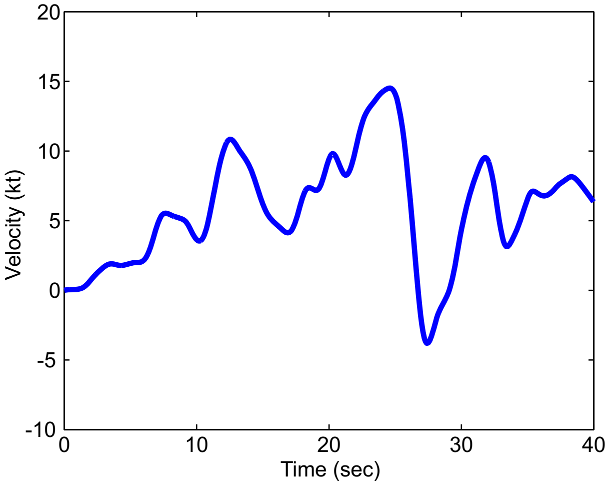

Fig. 5 The wind data in the *x* direction of the earth axis.

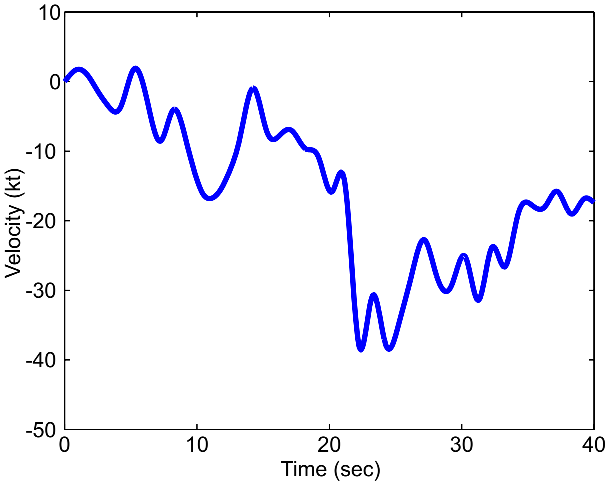

Fig. 6 The wind data in the *y* direction of the earth axis.

The simulation results are shown in Figs. 7 and 8. Figure 7 shows pitch angle, and Fig. 8 shows vertical acceleration. The aircraft flying on autopilot ran into atmospheric turbulence and the airspeed increased. The autopilot mode changed to speed control mode at 18 s. However, the autopilot could not reduce airspeed. The pitch angle increased to approximately 8.5 deg and then the stick shaker moved. Autopilot was disengaged and manual control started at 26 s. Because of quick and high-amplitude pilot control at high altitude, pitch angle oscillation occurred as shown in Fig. 7. As a result, vertical acceleration drastically changed as shown in Fig. 8. Airspeed, pitch angle, and vertical acceleration change in the accident analysis report are illustrated in Appendix. The simulation results follow the essential features of the accident well.

C. Application Results of the HACM architecture

Figures 9–15 show the application results of the HACM architecture as applied to the incident situation. This paper yields $\theta_{\min} = -11$ (deg), $\theta_{\max} = 11$ (deg), $\dot{\omega}_{\min} = -1.0\,(G)$, $\dot{\omega}_{\max} = 2.5\,(G)$, $\ddot{\omega}_{\min} = -0.3\,(G/s)$, and $\ddot{\omega}_{\max} = 0.3\,(G/s)$ by application of Eq. (2). In this simulation, we set $\sigma = 10.0$ in Eq. (6). Sampling time interval is 0.02 s. Figure 9 shows that the HACM architecture works to reduce the amplitude of pitch angle oscillation. Thus, it

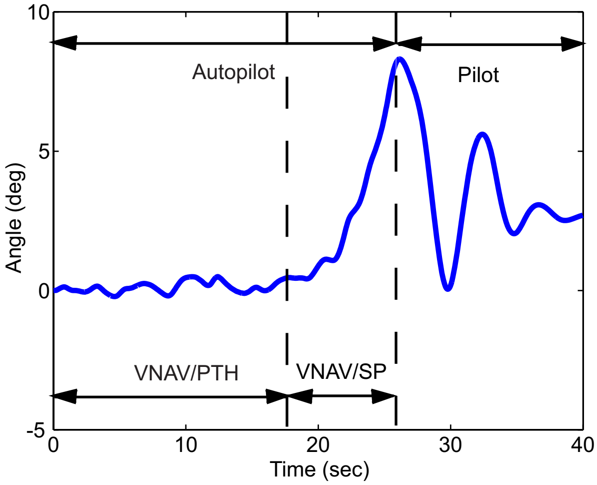

Fig. 7 Simulation results of aircraft incident: pitch angle *θ*.

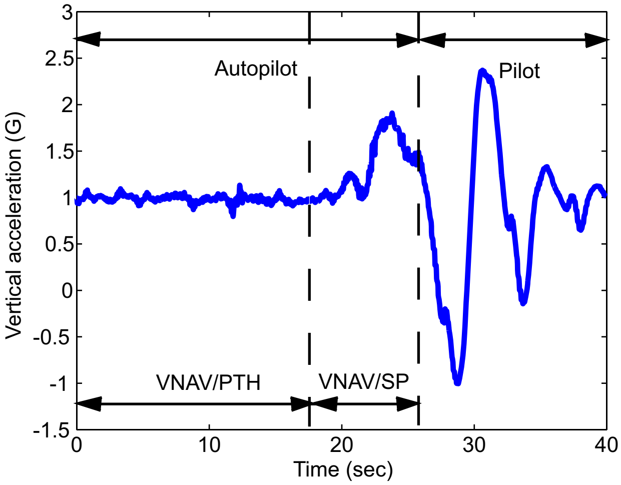

Fig. 8 Simulation results of aircraft incident: vertical acceleration *a*.

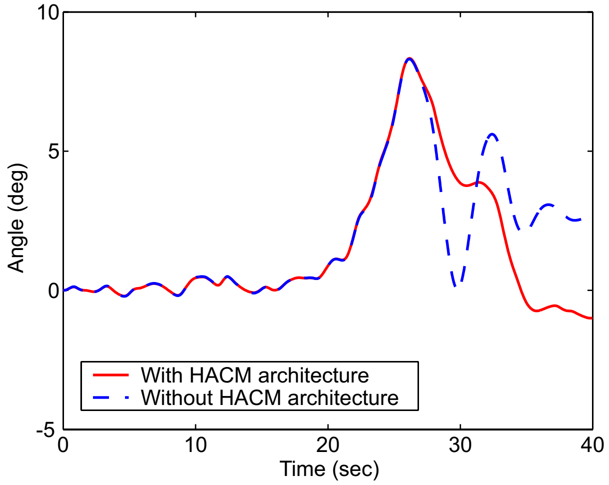

Fig. 9 Application results: pitch angle θ.

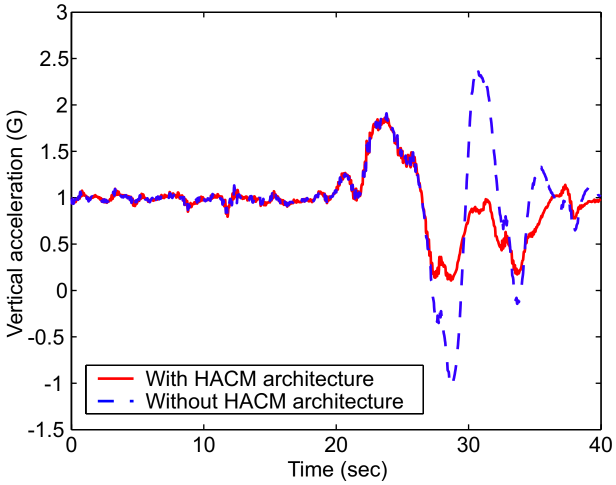

Fig. 10 Application results: vertical acceleration a.

contributes toward reducing the maximum value of vertical acceleration by approximately 1.1 G as shown in Fig. 10. Comparing with the incident result as shown in Fig. 8, the HACM architecture achieved 68% PIO reduction. Figure 11 shows the elevator input of the HACM architecture, and the HUMAN and CONTROLLER modules. As shown in Fig. 12, the ARBITER module adaptively adjusts the contribution ratios, and the elevator control of the HUMAN and CONTROLLER modules online after 26 s. Figures 13, 14, and 15 show how the change in error index $E_{ij}(t)$ corresponds to pitch angle, vertical acceleration, and change in vertical acceleration, respectively. Figure 15 shows that since pilot control exceeded the flight envelope for vertical acceleration, the contribution ratio was calculated to reduce pilot control.

D. Discussion

"*I had no time to check the information displayed on the monitor.*" "*I cannot remember whether or not I disengaged the autopilot.*" These were pilot comments in the accident analysis report. Conventionally, a man–machine interface has been developed to support pilot decision making and cognitive tasks. The above comments show that sometimes these tools do not work well under conditions which need prompt pilot action. One of the ideas for avoiding the effects

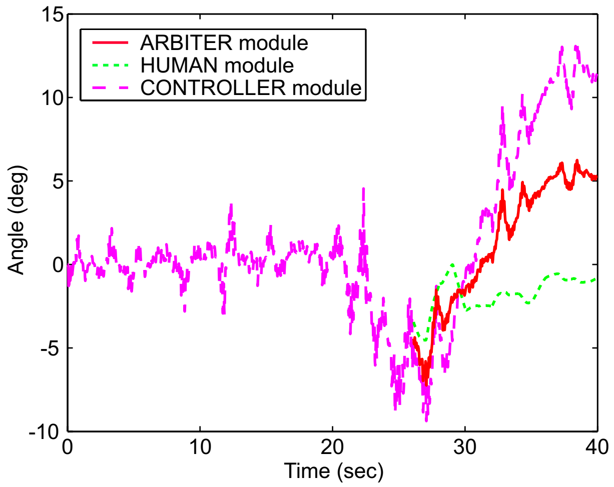

Fig. 11 Application results: elevator inputs δ_e.

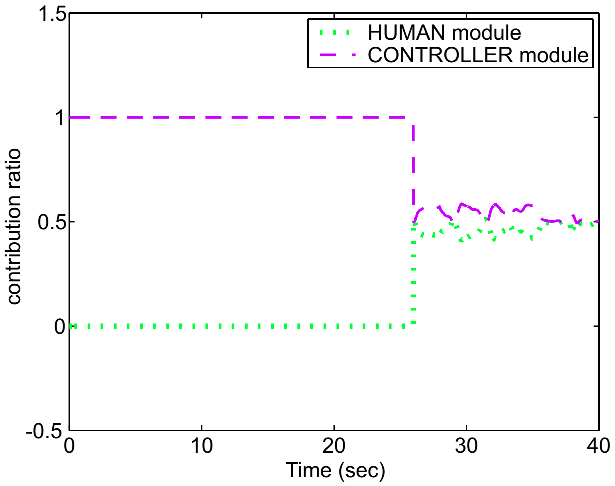

Fig. 12 Application results: contribution ratios λ_i.

of automation surprises is that the autopilot temporarily takes over pilot control thereby harmonizing automation with pilot maneuver.

The HACM architecture has a simple algorithm and structure consisting of the HUMAN, CONTROLLER, and ARBITER modules. The proposed architecture includes current automatic flight controller and inserts the ARBITER system between autopilot and pilot. This simplicity allows online application and further improvement. In [15], the first-named author added a new module, the OBSERVER module, which equips with an extended Kalman filter. In [21,22], the first-named author and Duong applied the design concept of the HACM architecture to harmonize air traffic controller and ground automation in the future air traffic management.

Since the ARBITER module evaluates the performance of the HUMAN and CONTROLLER module based on simulator, the accuracy of the HACM architecture depends on the error between simulation model and actual dynamics. Past studies [12,15] showed that the architecture performs well with the assumed modeling errors. Further confirmation of its performance is necessary under various assumptions. If the aircraft dynamics is changed by failure and/or accident, the ARBITER module should include the dynamics. The autopilot parameters should also be reconfigured. Flight controller reconfiguration has been investigated based on the estimation of aircraft dynamics

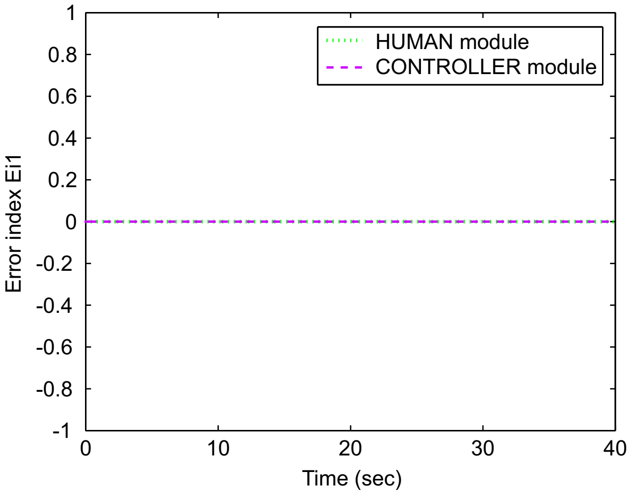

Fig. 13 Application results: error index E_{i1} corresponding to the pitch angle.

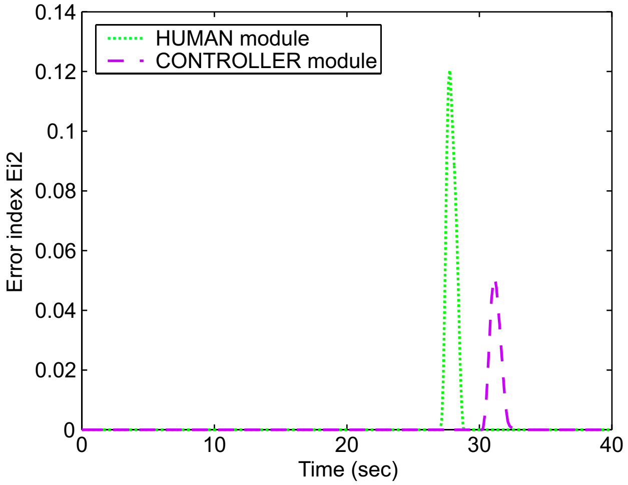

Fig. 14 Application results: error index E_{i2} corresponding to vertical acceleration.

online [23–27]. One of the next ideas is integration of these techniques into the HACM architecture. Pilot evaluation based on flight simulator experiments will help further development. More effective parameter setting should be investigated to make an even better system. Stability of the architecture should be analyzed in detail in future works.

IV. Conclusion

There are two potential benefits in employing the HACM architecture. First, the HACM architecture allows us to utilizes the advantages of both pilot and autopilot control, comprising of the abilities of both. The ARBITER module takes over pilot or autopilot control when either takes unsafe action, and generates appropriate control commands. While a pilot is manually controlling an aircraft safely, the ARBITER module works to prevent autopilot interference and to protect pilot control. On the other hand, if the pilot control induces unsafe aircraft movement, the ARBITER gives control to the autopilot. By adaptively adjusting control authorities, the proposed architecture could resolve conflicts between pilot and autopilot when switching between the two. Second, HACM architecture has a simple framework, and its algorithm is comprised of three types of modules: HUMAN, CONTROLLER, and ARBITER.

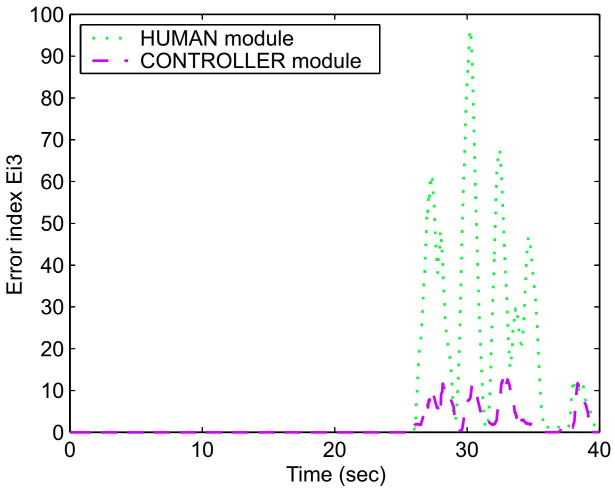

Fig. 15 Application results: error index E_{i3} corresponding to change in vertical acceleration.

This simplicity enables us to apply the architecture online. In addition, it will be advantageous to add further modules to develop an even better system.

This paper culminates in the ultimate purpose of the HACM architecture. The concepts and structures of the HACM architecture were summarized based on findings in our past studies. The architecture was applied to a past aircraft incident related to PIO. The effectiveness in reducing the impact of PIO was confirmed through numerical simulation. Simulation results showed 68% of PIO reduction. Finally, benefits and future works were summarized

Appendix

A. Nonlinear equations for aircraft dynamics

Based on Roskam [28], the nonlinear dynamics of a B747-100 flying at 40,000 ft were utilized in this simulation. In order to simplify the simulation, we focused on the longitudinal movement of the aircraft. Altitude h, airspeed V (airspeed following x direction of the body axis u, airspeed following z direction of the body axis w), and pitch angle θ and pitch angle velocity q were calculated using the following equations [29].

Altitude:

$$\dot{h} = -u \sin\theta + v \sin\phi \cos\theta + w \cos\phi \cos\theta + W_z \tag{A1}$$

where v is the airspeed following y direction of the body axis, ϕ the roll angle, and W_z the wind component following z direction of the earth axis.

Airspeed:

$$\dot{u} = \frac{F_x}{m} - q(w + w_g) + r(v + v_g) - g \sin\theta - \dot{u}_g \tag{A2}$$

$$\dot{w} = \frac{F_z}{m} - p(v + v_g) + q(u + u_g) + g \cos\theta \cos\phi - \dot{w}_g \tag{A3}$$

where m is the mass of the aircraft, g the gravity constant, F_x the external force following x direction of the body axis, F_z the external force following z direction of the body axis, r the yaw angular acceleration, p the roll angular acceleration u_g the wind component following x direction of the body axis, v_g the wind component following y direction of the body axis, and w_g the wind component following z direction of the body axis.

The terms F_x and F_z in (A2) and (A3) were calculated as shown in following equations

$$F_x = T - D \cos\alpha + L \sin\alpha \tag{A4}$$

$$F_z = -D \sin\alpha - L \cos\alpha \tag{A5}$$

where

$$D = \tfrac{1}{2}\rho V^2 S C_D \tag{A6}$$

$$L = \tfrac{1}{2}\rho V^2 S C_L \tag{A7}$$

$$\alpha = \tan^{-1}\left(\frac{w}{u}\right) \cong \frac{w}{u} \tag{A8}$$

and where T denotes the thrust, D the drag, L the lift, α the angle of attack, C_D the drag coefficient, C_L the lift coefficient, ρ the air density, and S the wing area.
Pitch angle and pitch angle velocity:

$$\dot{\theta} = q\cos\phi - r\sin\phi \tag{A9}$$

$$\dot{q} = \frac{1}{I_y}\{\bar{M} + (I_z - I_x)pr + I_{xz}(r^2 - p^2)\} \tag{A10}$$

where I_x is the inertia moment about x axis (body axis), I_y the inertia moment about y axis (body axis), I_z the inertia moment about z axis (body axis), I_{xz} the product of inertia $\int xz\,\mathrm{d}m$, and $\bar{M}$ the pitching moment.

Pitching moment $\bar{M}$ in (A10) is given in the following equation

$$\begin{aligned}\bar{M} &= \frac{1}{2}\rho V^2 S\bar{c}C_m \\ &= \frac{1}{2}\rho V^2 S\bar{c}\left\{C_{m0} + C_{m\alpha}\alpha + C_{m\dot{\alpha}}\bar{c}\frac{\dot{\alpha}}{2V} + C_{mq}\bar{c}\frac{q}{2V} + C_{m\delta e}\delta_e\right\}\end{aligned} \tag{A11}$$

where

$$\dot{\alpha} = \frac{\mathrm{d}}{\mathrm{d}t}\tan^{-1}\left(\frac{w}{u}\right) = \frac{\dot{w}u - w\dot{u}}{u^2 + w^2} \tag{A12}$$

and where $\bar{c}$ is the average aerodynamic cord, C_m the moment coefficient, C_{m0} the pitching-moment coefficient at zero α, $C_{m\alpha}$ the α derivative which takes place in the change of the moment m, $C_{m\dot{\alpha}}$ the $\dot{\alpha}$ derivative which takes place in the change of the moment m, C_{mq} the q derivative which takes place in the change of the moment m, $C_{m\delta_e}$ the δ_e derivatives which takes place in the change of the moment m, and δ_e the elevator angle.

B. Autopilot design

As shown in Fig. 4, the designed autopilot consisted of a coupler, filter1, filter 2, PID controller, and feedback gains [29].
Coupler: The coupler generates the pitch command θ_c using the differential of the airspeed $\dot{u}$ as follows

$$\theta_C = Ku \times \dot{u} \tag{A13}$$

In this paper we define $Ku = -0.03$. In order to limit the change in the target pitch angle, we give the following conditions

$$\text{If } \dot{\theta}_C(t_n) < -\varepsilon, \text{ then } \theta_C(t_n) = \theta_C(t_{n-1}) - \varepsilon\Delta t \tag{A14}$$

$$\text{If } \dot{\theta}_C(t_n) > \varepsilon, \text{ then } \theta_C(t_n) = \theta_C(t_{n-1}) + \varepsilon\Delta t \tag{A15}$$

where t_n means the time corresponds to time step n, Δt corresponds to the step time used in the simulation. In this paper $\varepsilon = 3.0$ (deg/s).
Filters (*Filter* 1 *and* 2): We utilized the following two-dimensional low-pass filter $F(s)$

$$F(s) = \frac{\omega_f^2}{s^2 + 2\varsigma_f\omega_f s + \omega_f^2} \tag{A16}$$

where $\omega_f = 1.0$, $\varsigma_f = 0.8$.

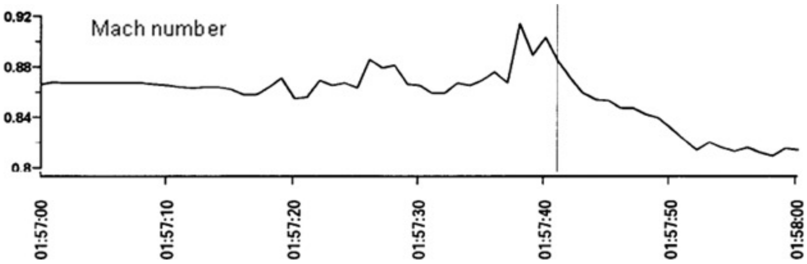

Fig. A1 Incident data: airspeed (MACH number) [8].

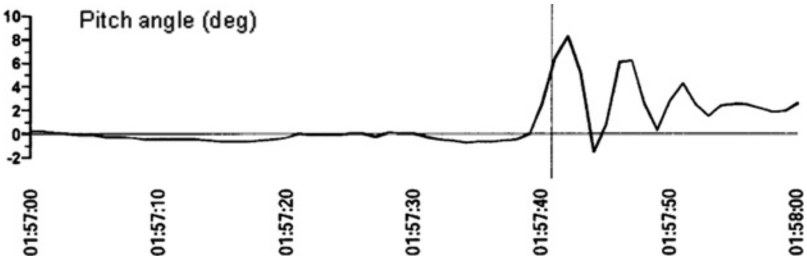

Fig. A2 Incident data: pitch angle [8].

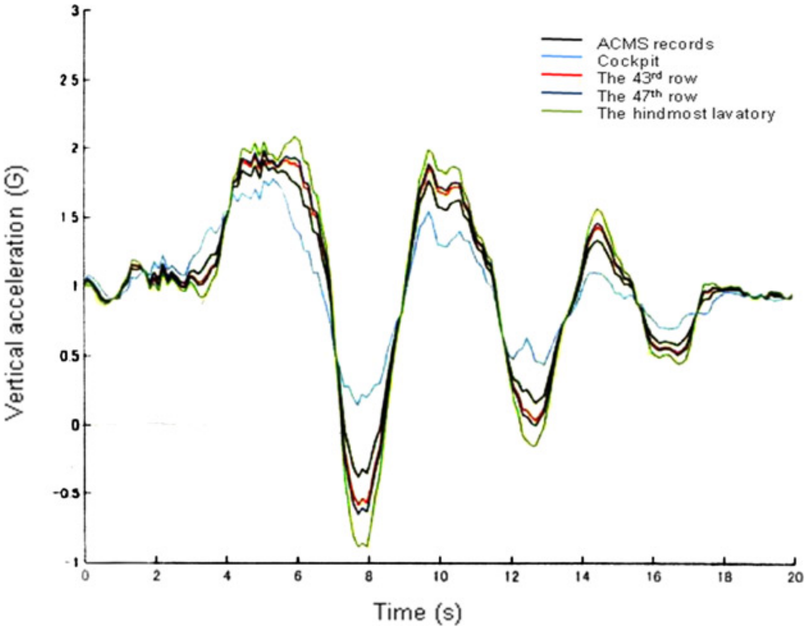

Fig. A3 Incident data: vertical acceleration [8].

PID controller: The following PID controller $C(s)$ was utilized in this autopilot

$$C(s) = K_P + \frac{K_I}{s} + K_D s \tag{A17}$$

where $K_P = -1.0$, $K_I = -0.5$, $K_D = -1.0$.
Feedback gains: The simulation used the following feedback gains: $K_a = 0.01$, $K_q = 0.75$.

C. Incident situation

Figures A1–A3 show airspeed, pitch angle, and vertical acceleration records in the accident analysis report [8].

Acknowledgments

This research was supported by grants from the Japan Society for the Promotion of Science. The authors are grateful to the anonymous reviewers for their constructive comments and to Dr. Claus Gwiggner for the editorial assistance.

References

[1] Hawkins, F. K., *Human Factors in Flight*, Gower Technical Press, Hampshire, 1987.

[2] Federal Aviation Administration Human Factors Team, "The Interfaces between Flightcrews and Modern Flight Deck Systems," http://ocw.mit.edu/NR/rdonlyres/Aeronautics-and-Astronautics/16-422Spring2004/5C356BFF-2264-4A2C-8845-B1904BFEB783/0/interfac.pdf, June 1996.

[3] Miyagi, M., *Serious Accidents and Human Factors*, AIAA, New York, NY, 2005, Chap. 2.

[4] National Transportation Safety Board, http://www.ntsb.gov/ntsb.

[5] Plane Crash Info.com, http://www.planecrashinfo.com/index.html.

[6] Aircraft and Railway Accidents Investigation Commission, "Aircraft Accident Analysis Report 96-5 B1816," Tokyo, 1996 (in Japanese).

[7] Aircraft and Railway Accidents Investigation Commission, "Aircraft Accident Analysis Report 99-8 JA8580," Tokyo, 1999 (in Japanese).

[8] Aircraft and Railway Accidents Investigation Commission, "Aircraft Accident Analysis Report AA2006-1," Tokyo, 2006 (in Japanese).

[9] Rushby, J., "Using Model Checking to Help Discover Mode Confusions and Other Automation Surprises," *Reliability Engineering and System Safety*, Vol. 75, 2002, pp. 167–177.
doi: 10.1016/S0951-8320(01)00092-8

[10] Joshi, A., Miller, S. P., and Heimdahl, M. P. E., "Mode Confusion Analysis of a Flight Guidance System Using Formal Methods," *Proceedings of the 22nd Digital Avionics Systems Conference*, Indianapolis, Oct. 2003.

[11] Vakil, S. S., and Hansman, R. J., "Approaches to Mitigating Complexity-driven Issues in Commercial Autoflight Systems," *Reliability Engineering and System Safety*, Vol. 75, 2002, pp. 133–145.
doi: 10.1016/S0951-8320(01)00090-4

[12] Itoh, E., and Suzuki, S., "A New Architecture to Coordinate Automation with Pilot Maneuver," *Transactions of the Aeronautical and Astronautical Society of the ROC*, Vol. 37, No. 3, 2005, pp. 203–214.

[13] Itoh, E., and Suzuki, S., "Resolving Conflicts between Pilot and Automation," *Proceedings of the 4th Eurocontrol Innovative Research Workshop & Exhibition*, Bretigny-sur-Orge, Dec. 2005, pp. 191–200.

[14] Itoh, E., and Suzuki, S., "A New Approach to Automation that Takes Account of Adaptive Nature of Pilot Maneuver," *Proceedings of World Automation Conference* (*WAC 2006*), Budapest, July 2006.

[15] Itoh, E., "A New Architecture to Harmonize Automation and Pilot Maneuver," *Proceedings of the 25th Congress of International Council of Aeronautical Sciences* (*ICAS 2006*), Hamburg, Sept. 2006.

[16] Jacobs, R. A., Jordan, M. I., Nowlan, S. J., and Hinton, G. E., "Adaptive Mixture of Local Experts," *Neural Computation*, Vol. 3, 1991, pp. 79–87.
doi: 10.1162/neco.1991.3.1.79

[17] Gomi, H., and Kawato, M., "Recognition of Manipulated Objects by Motor Learning with Modular Architecture Networks," *Neural Networks*, Vol. 6, 1993, pp. 485–497.
doi: 10.1016/S0893-6080(05)80053-X

[18] Wolpert, D. M., and Kawato, M., "Multiple Paired Forward and Inverse Models for Motor Control," *Neural Networks*, Vol. 11, 1998, pp. 1317–1329.
doi: 10.1016/S0893-6080(98)00066-5

[19] Haruno, M., Wolpert, D. M., and Kawato, M., "Multiple Paired Forward–Inverse Models for Human Motor Learning and Control," *Advances in Neural Information Processing Systems*, edited by M. S. Kearns, S. A. Salla, and D. A. Cohn, Vol. 11, MIT Press, Cambridge, MA, 1999, pp. 31–37.

[20] Haruno, M., Wolpert, D., and Kawato M., "MOSAIC Model for Sensorimotor Learning and Control," *Neural Computation*, Vol. 13, 2001, pp. 2201–2220.
doi: 10.1162/089976601750541778

[21] Itoh, E., and Duong, V., “Analyzing Interference between Ground Automation and Air Traffic Controllers from a Control Theory Approach,” *Proceedings of the 7th AIAA Aviation Technology, Integration and Operation Conference*, Belfast, Sept. 2007, AIAA-2007-7741.

[22] Itoh, E., and Duong, V., “Preventing Interferences between Air Traffic Controller and Future Ground Automation from a Control Theory Approach,” *Proceedings of the 6th Eurocontrol Innovative Research Workshop and Exhibition*, Bretigny-sur-Orge, Dec. 2007.

[23] Velo, J. G., and Walker, B. K., “Aerodynamic Parameter Estimation for High-Performance Aircraft Using Extended Kalman Filter,” *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 6, 1997, pp. 1257–1259.

[24] Song, Y., Campa, G., Napolitano, N., Seanor, B., and Perhinschi, M. G., “Online Parameter Estimation Techniques Comparison Within a Fault Tolerant Flight Control System,” *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 3, 2002, pp. 528–537.

[25] Jategaonkar, R. V., and Plaetschke, E., “Algorithm for Aircraft Parameter Estimation Accounting for Process and Measurement Noise,” *Journal of Aircraft*, Vol. 26, No. 4, 1989, pp. 360–372.
doi: 10.2514/3.45769

[26] Bossi, J. A., and Bryson, A. E., Jr., “Disturbance Estimation for a STOL Transport During Landing,” *Journal of Guidance, Control, and Dynamics*, Vol. 5, No. 3, 1982, pp. 258–262.
doi: 10.2514/3.56165

[27] Tanaka, N., Suzuki, S., Masui, K., and Tomita, T., “Restructurable Guidance and Control for Aircraft with Failures Considering Gust Effects,” *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 3, 2006, pp. 671–679.
doi: 10.2514/1.15680

[28] Roskam, J., *Airplane Design Part VI*, Design Analysis & Research, Lawrence, 2000.

[29] Etkin, B., and Reid, L. D., *Dynamics of Flight Stability and Control*, 3rd ed., Wiley, Hoboken, NJ, 1996, Chaps. 4, 8.

Ella Atkins
Associate Editor