DEVELOPMENT AND FLIGHT TESTING OF A PARAMETER IDENTIFICATION ALGORITHM FOR RECONFIGURABLE CONTROL

David G. Ward, Jeffrey F. Monaco [*] Marc Bodson [†]

Barron Associates, Inc. *University of Utah*

ABSTRACT

The paper discusses the results of a series of flight tests in which the performance of a real-time parameter identification algorithm was evaluated. The identification algorithm was combined with an automatic control design procedure in order to obtain control reconfiguration capabilities. Significant challenges were encountered because of the poor information content of the signals used for identification and because of the requirements for autonomy, reliability, and fast adaptation. A modified sequential least-squares algorithm was used for identification. The algorithm is presented in the paper, and its advantages are discussed, together with the solutions that were developed to address the problems posed by the specific application. Typical parameter identification results from the flight tests are shown. The flight tests culminated in a successful landing of the aircraft under a simulated missing elevon, demonstrating the value of the methods under investigation.

[*] D.G. Ward is Senior Research Scientist and Member AIAA. J.F. Monaco is Research Scientist. Barron Associates, Inc., The Jordan Building, 1160 Pepsi Place, Suite 300, Charlottesville, Virginia 22901.

[†] M. Bodson is Associate Professor and Senior Member AIAA. Department of Electrical Engineering, University of Utah, Salt Lake City, Utah 84112.

1. INTRODUCTION

A goal of modern flight control research is improved aircraft survivability and enhanced robustness of aircraft responses to a wide array of anomalous operating conditions. The emphasis on robustness has led to studies of reconfigurable control, which is intended to provide rapid adaptation to control effector and airframe damage, imperfect characterizations of aircraft performance in new flight regimes, and less traumatic events such as release of stores or gradual component aging. A notable milestone in the development of reconfigurable control systems was the flight testing of the so-called *Self-Repairing Flight Control System* [1]. This system implemented a failure detection and identification approach to the problem. An alternative approach is based on adaptive control techniques, which do not require prior assumptions about the nature and characteristics of failures.

During the summer of 1996, a series of flight tests demonstrated an adaptive approach to reconfigurable flight control called *self-designing controller* (SDC). This indirect adaptive control architecture computed a time-varying model of the aircraft dynamics and communicated the results to an optimal control module. This module computed the effector commands required to achieve the desired aircraft responses as specified by a set of flying-qualities models. The SDC was found to be able to reconfigure rapidly after single or multiple impairments, thus making effective use of residual effector authority and greatly enhancing aircraft survivability. At the core of the algorithm was a novel on-line system identification technique that could rapidly track time-varying parameters and was robust to adverse conditions such as low excitation or correlated inputs. The objective of this paper is to present the algorithm and the results of its implementation in the flight tests.

Parameter identification of aircraft dynamics has long been a subject of interest in the aerospace community (see, *e.g.,* [2], [3], [4]). Problems such as identification of nonlinear dynamics [5] and reliable error analysis [6] continue to make it an area of active research today. The problem of parameter identification for reconfigurable flight control is similar, but brings additional difficulties, which may be categorized as follows:

- *Lack of control over the actuator signals.* A significant difficulty arises when the actuator signals are determined by a control law and cannot be freely selected. It has been shown that considerable improvements in identification performance can be obtained through optimization of the signals applied to the control surfaces [7]. Conversely, in a reconfigurable control application, the signals exhibit highly undesirable characteristics, including:

  1. high levels of correlation between the control signals and the aircraft states (in particular for linear state feedback control laws);
  2. long periods of quiescence (*e.g.,* in steady flight);
  3. highly coupled longitudinal and lateral motions (in particular for rapid roll maneuvers);
  4. excitation of nonlinear dynamics (for example inertial couplings).

  In addition, because the control signals are determined by a control law which, itself, depends on the identification results, adverse interactions are sometimes induced between the identification and the control components of the system, further aggravating the problem.

- *Increased number of parameters for failed aircrafts.* Aircrafts often loose their symmetry after failures, requiring that more parameters be identified. For example, the roll effectiveness of a symmetric tail deflection may be assumed to be zero for an unfailed aircraft, but not otherwise. In general, accounting for possible failures may prevent the ganging of some surfaces, such as spoilers, and result in a significant increase in the number of parameters to be identified.

- *Real-time operation.* The computational requirements of a parameter identification scheme for reconfigurable control must be compatible with available computers. Further, the procedure must be performed with minimal supervision and may not require extensive trial and error adjustments by the pilot.

In summary, the problem of identifying aircraft parameters poses considerable problems when real-time operation under feedback is required. This paper discusses solutions to these problems and the results of the implementation of the solutions in flight tests.

2. EQUATIONS OF MOTION AND PARAMETERIZATION FOR IDENTIFICATION

2.1 Equations of Motion

This section considers the nonlinear equations of motion of an aircraft. We begin with the nonlinear wind-axes equations of motion

$$\begin{aligned}
\dot{V}_t &= \frac{1}{m}\left(T_x \cos\alpha \cos\beta - D\right) + g_D && (1)\\
\dot{\beta} &= \frac{1}{mV_t}\left(Y_a - T_x \cos\alpha \sin\beta\right) + p\sin\alpha - r\cos\alpha + \frac{g_{Y_a}}{V_t} && (2)\\
\dot{\alpha} &= -\frac{1}{mV_t\cos\beta}\left(L + T_x \sin\alpha\right) + q - \tan\beta\left(p\cos\alpha + r\sin\alpha\right) + \frac{g_L}{V_t\cos\beta} && (3)
\end{aligned}$$

where V_t is the aircraft velocity, m is the mass of the aircraft, T_x is the component of the thrust along the body x-axis, α is the angle of attack, β is the angle of sideslip, D is the drag (along the negative wind x-axis), g_D is the component of the gravitational acceleration along the wind x-axis, Y_a is the lateral force (along the wind y-axis), (p, q, r) is a vector of body-axes rotational rates, g_{Y_a} is the component of the gravitational acceleration along the wind y-axis, L is the lift force (along the negative wind z-axis), and g_L is the component of the gravitational acceleration along the wind z-axis. For this work, we followed the ANSI/AIAA recommended practice of using wind-axes forces [8]. The components of the gravitational acceleration are given by $g_D = -g_0 \sin\gamma$, $g_{Y_a} = g_0 \cos\gamma \sin\mu$, $g_L = g_0 \cos\gamma \cos\mu$, where g_0 is the acceleration of gravity, γ is the flight path angle, and μ is the bank angle (about the velocity vector).

The angular equations of motion are resolved in the body (not wind) axes system. These equations are given by

$$\Gamma\dot{p} = I_{zz}\bar{L} + I_{xz}\bar{N} + I_{xz}\left(I_{xx} - I_{yy} + I_{zz}\right)pq + \left(I_{zz}\left(I_{yy} - I_{zz}\right) - I_{xz}^2\right)qr \tag{4}$$

$$I_{yy}\dot{q} = \bar{M} + \left(I_{zz} - I_{xx}\right)pr + I_{xz}\left(r^2 - p^2\right) \tag{5}$$

$$\Gamma\dot{r} = I_{xz}\bar{L} + I_{xx}\bar{N} + \left(I_{xx}\left(I_{xx} - I_{yy}\right) + I_{xz}^2\right)pq - I_{xz}\left(I_{xx} - I_{yy} + I_{zz}\right)qr \tag{6}$$

where $\Gamma = I_{xx}I_{zz} - I_{xz}^2$, I_{ii} is a moment of inertia, I_{ij} is a body-axes product of inertia, $\bar{L}$ is the rolling moment, $\bar{M}$ is the pitching moment, and $\bar{N}$ is the yawing moment. It is assumed that I_{xy} and I_{yz} are negligible, which may not always be valid for reconfigurable flight control, but is appropriate in most cases.

The forces and moments acting on the aircraft are functions of the aerodynamic angles, α and β, of the body-axes angular rates, p, q, and r, and of the effector positions. For the F-16 aircraft that was used in the flight tests, the effectors are δ_{tl} (left tail or elevon), δ_{tr} (right tail or elevon), δ_{fl} (left trailing-edge flap or flaperon), δ_{fr} (right trailing-edge flap or flaperon), and δ_{rud} (rudder). The F-16 effector suite also includes leading-edge flaperons. However, deflections of these surfaces are scheduled as functions of the aircraft's angle of attack in the standard F-16 (Block 40) control law, and the variables were not assumed to be available for control reconfiguration. Another command that was not used for control reconfiguration was the symmetric flaperon command. As a result, the following pseudo-effectors were considered

$$\delta_{ts} = \text{symmetric tail} \tag{7}$$

$$\delta_{ta} = \text{asymmetric tail} \tag{8}$$

$$\delta_{fa} = \text{asymmetric flaperon} \tag{9}$$

$$\delta_{rud} = \text{rudder} \tag{10}$$

2.2 Parameterization for Identification

For identification, it is necessary to represent the uncertain forces and moments of the equations of motions in terms of parameters to be determined. While one may choose to identify nondimensional stability and control derivatives, it was preferred to identify lumped, dimensionalized coefficients. The relationship between the dimensional coefficients and the standard, nondimensional derivatives is given in Appendix. While the dimensional parameters vary with Mach number and dynamic pressure, the variations are slow relative to the states and the computations were found to be simplified in this manner.

For identification purposes, the equations of motion are divided into separate pitch and roll/yaw equations

$$\begin{aligned} \dot{x}_{lon} - NL_{lon}(p,q,r,\alpha,\beta) &= \theta_{lon}^T \phi_{lon} && (11) \\ \dot{x}_{lat} - NL_{lat}(p,q,r,\alpha,\beta) &= \theta_{lat}^T \phi_{lat} && (12) \end{aligned}$$

where $x_{lon}^T = [q, \alpha]$ and $x_{lat}^T = [p, r, \beta]$. $NL_{lon}(p,q,r,\alpha,\beta)$ and $NL_{lat}(p,q,r,\alpha,\beta)$ represent the nonlinear terms due to inertial cross-couplings and gravitational effects, and are given by

$$NL_{lon}(p,q,r,\alpha,\beta) = \begin{bmatrix} NL_q \\ NL_\alpha \end{bmatrix} = \begin{bmatrix} \frac{(I_{zz}-I_{xx})}{I_{yy}} pr + \frac{I_{xz}}{I_{yy}} (r^2 - p^2) \\ q - \tan\beta\,(p\cos\alpha + r\sin\alpha) + \frac{g_L}{V_t \cos\beta} \end{bmatrix}, \qquad (13)$$

$$NL_{lat}(p,q,r,\alpha,\beta) = \begin{bmatrix} NL_p \\ NL_r \\ NL_\beta \end{bmatrix} = \begin{bmatrix} \frac{I_{xz}(I_{xx}-I_{yy}+I_{zz})}{\Gamma} pq + \frac{\left(I_{zz}(I_{yy}-I_{zz})-I_{xz}^2\right)}{\Gamma} qr \\ \frac{\left(I_{xx}(I_{xx}-I_{yy})+I_{xz}^2\right)}{\Gamma} pq - \frac{I_{xz}(I_{xx}-I_{yy}+I_{zz})}{\Gamma} qr \\ p\sin\alpha - r\cos\alpha + \frac{g_{Y_a}}{V_t} \end{bmatrix}. \qquad (14)$$

ϕ_{lon} and ϕ_{lat} are vectors of observations, called *regressors*, which consist of aircraft states, effector positions, and a bias term, with

$$\phi_{lon}^T = [q, \alpha, \delta_{ts}, \delta_{ta}, 1], \qquad \phi_{lat}^T = [p, r, \beta, \delta_{ts}, \delta_{ta}, \delta_{fa}, \delta_{rud}, 1]. \qquad (15)$$

θ_{lon}, θ_{lat} are matrices of parameters to be identified

$$\theta_{lon}^T = \begin{bmatrix} \theta_q^T \\ \theta_\alpha^T \end{bmatrix} = \begin{bmatrix} \theta_{q_q} & \theta_{q_\alpha} & \theta_{q_{ts}} & \theta_{q_{ta}} & \theta_{q_0} \\ \theta_{\alpha_q} & \theta_{\alpha_\alpha} & \theta_{\alpha_{ts}} & \theta_{\alpha_{ta}} & \theta_{\alpha_0} \end{bmatrix}, \qquad (16)$$

$$\theta_{lat}^T = \begin{bmatrix} \theta_p^T \\ \theta_r^T \\ \theta_\beta^T \end{bmatrix} = \begin{bmatrix} \theta_{p_p} & \theta_{p_r} & \theta_{p_\beta} & \theta_{p_{ts}} & \theta_{p_{ta}} & \theta_{p_{fa}} & \theta_{p_{rud}} & \theta_{p_0} \\ \theta_{r_p} & \theta_{r_r} & \theta_{r_\beta} & \theta_{r_{ts}} & \theta_{r_{ta}} & \theta_{r_{fa}} & \theta_{r_{rud}} & \theta_{r_0} \\ \theta_{\beta_p} & \theta_{\beta_r} & \theta_{\beta_\beta} & \theta_{\beta_{ts}} & \theta_{\beta_{ta}} & \theta_{\beta_{fa}} & \theta_{\beta_{rud}} & \theta_{\beta_0} \end{bmatrix}. \qquad (17)$$

Several comments are worth pointing out, regarding the parameterization:

- Although the equations are separated into longitudinal and lateral equations, the longitudinal and lateral variables appear in both equations. In other words, the formulation does not constitute an assumption of separation of longitudinal and lateral dynamics, but rather an assumption of which variables affect the longitudinal and lateral accelerations. In particular, it is assumed that the asymmetric flaperon does not produce any pitching acceleration. On the other hand, a symmetric tail command may produce a rolling moment (considering a possible failure). It is assumed that the two longitudinal accelerations are affected by the same variables, and that the three lateral accelerations are also affected by the same variables, although the two sets are different. As we will see in the next section, the grouping of the longitudinal and lateral equations saves computations.

- The last parameter of each row of the parameter matrices is a bias term that represents the forces or moments that are not represented by the other parameters. It includes trim-related variables of the aircraft, and also unmodelled effects such as those related to Mach number and the slow states. In a similar way, the effect of δ_{fs}, which is commanded as a (mostly) linear function of angle of attack, is contained in the α-derivatives and the bias terms.

- The equations are *linearly parameterized*, meaning that the unknown parameters appear linearly in the equations. This makes possible the use of efficient least-squares algorithms. However, the equations of motions are nonlinear. It is also possible to represent nonlinear variations with respect to angle of attack by including terms such as α^2, α^3, in the regressor variables.

- For control purposes, the nonlinear portion (NL) of the equations of motion given in (13), (14) may be linearized to obtain a linear state-space model. This state-space model is given in Appendix. Note that the result is a set of equations with coupled longitudinal and lateral dynamics.

- The nonlinear kinematic and gravitational terms are accounted for explicitly because these terms may vary rapidly with time (that is, as rapidly as the states themselves vary), and because their contribution is sometimes significant. Indeed, Fig. 1 shows the relative importance of two nonlinear effects, as observed from flight test data. On the top is a plot showing $\dot{q}$ as a solid line, and $pr(I_{zz} - I_{xx})/I_{yy} + (r^2 - p^2)I_{xz}/I_{yy}$ as a dashed line. On the bottom is a plot showing $\dot{\alpha} - q$ as a solid line, and $\tan\beta(p\cos\alpha + r\sin\alpha)$ as a dashed line. In both cases, one finds that nonlinear effects account for a visibly significant portion of the signal for a long period of time. If neglected in the context of an adaptive identification algorithm, such effects could significantly perturb the responses of the parameters. Situations such as shown in Fig. 1 are not very frequent but not unusual either. In the case of the plot on the top, the roll rate reached approximately 50 deg/s and the yaw rate reached 10 deg/s. In the case of the plot of the bottom, the angle of sideslip reached 3 degrees, the roll rate reached 12 deg/s and the yaw rate 6 deg/s. In other words, the maneuvers were not particularly aggressive.

- Finally, an option would have been to identify terms such as $(I_{zz} - I_{xx})/I_{yy}$ in the algorithm. Although the inertia terms vary significantly with loading and flight condition, it was found that the ratios were relatively constant, and nominal values were used in order to reduce the number of parameters.

3. REGULARIZED PARAMETER IDENTIFICATION ALGORITHM

3.1 The Modified Sequential Least Squares Algorithm

In the previous section, the equations of motion were cast in a set of equations of the form

$$y = {\theta^*}^T \phi \tag{18}$$

where y and ϕ are vectors of measured variables, and θ^* is a matrix of unknown parameters. Each column of θ contains the parameters corresponding to one of the components of the vector y. There are two significant difficulties with the on-line estimation of the aircraft parameters, namely data collinearities and time-varying parameters.

Data collinearities occur when any of the variables of the vector ϕ can be represented as a linear combination of the other variables. In such case, the contributions of the components of θ^* may not be determined in a unique manner. Such condition can be caused by: (1) trim flight conditions where the surface and state commands are constant, (2) constant linear state feedback where the effector commands are a linear combination of the states, or (3) "ganged" effectors (*e.g.,* combination of asymmetric flap and asymmetric tail to generate rolling moment). These collinearities are all cases of *insufficient excitation* and may lead to divergence of the parameters in an adaptive algorithm and in the presence of noise. A solution to the problem consists in extending the memory of the algorithm over periods of time that are long enough that sufficient information is available.

Time-varying parameters result from changes in flight condition (slow variations) or during impairment, stores release, or other abrupt changes (fast variations). The presence of time-varying parameters translates into the need for rapid adaptation and, therefore, the need for short memory length. However, this requirement is contradictory to the need of stretching the memory of the algorithm long enough to get uncorrelated data.

To overcome the problem of identifying time-varying parameters in a system which is often insufficiently excited, a *modified sequential least-squares* (MSLS) algorithm was developed [9]. The parameter identification algorithm incorporates constraints to prevent numerical difficulties that occur when data windows are small enough to track varying parameters. The constraints are added to a standard squared-error cost function with forgetting, to yield

$$J(\theta(t)) = \frac{1}{2}\sum_{k=t_0}^{t} \lambda^{t-k}\|y(k) - \theta(t)^T\phi(k)\|^2 + \frac{\nu}{2}\|W_0^{1/2}\left(K(t) - M\theta(t)\right)\|^2 \qquad (19)$$

where $y \in \Re^{o\times 1}$, $\phi \in \Re^{n\times 1}$ are vectors of measured signals, $\theta \in \Re^{n\times o}$ is a matrix of estimated parameters, n is the number of parameters per output y, and o is the number of signals in y. $\|.\|$ is the Euclidean norm and $0.0 < \lambda \leq 1.0$ is the "forgetting factor" used to discount prior observations. The constraints consist in linear equalities of the form

$$K(t) = M\theta(t) \qquad (20)$$

where $K(t) \in \Re^{\ell \times o}$ and $M \in \Re^{\ell \times n}$ define linear constraint relationships, with ℓ being the number of constraints. $W_0 \in \Re^{\ell \times \ell}$ in (19) is the relative penalty associated with each constraint, and ν is the area of the window over which the cost function is computed (it is included in the penalty term so that the relative influence of a fixed weight remains the same, regardless of the value of the forgetting factor used).

Temporal and *spatial* constraints are used. Temporal constraints penalize parameter values that deviate from their previous estimate. In such case, $K(t) = M\theta(t-1)$, where $\theta(t-1)$ is the estimate at the previous time instant. Such temporal constraints result in a smoothing over time, but do not hinder the ability to track rapidly varying parameters if there is sufficient excitation to determine them. Spatial constraints may penalize parameter estimates that diverge from *a priori* estimates of their true values. These estimates can be constant, or they can be computed by on-board nonlinear models. Spatial constraints may also include constraints among the different parameters. For example, the lift and moment generated by a tail surface are related by the distance to the center of gravity, and this relationship may be incorporated as a constraint. Other constraints based on flight dynamics [10] may also be used.

Because the cost function in (19) is convex and continuously differentiable, its minimum can be found by solving $dJ(\theta)/d\theta = 0$. From (19), it can be checked that

$$\frac{dJ(\theta)}{d\theta} = \sum_{k=t_0}^{t} \lambda^{t-k}\phi(k)\phi^T(k)\theta + \nu M^T W_0 M\theta - \sum_{k=t_0}^{t} \lambda^{t-k}\phi(k)y^T(k) - \nu M^T W_0 K(t) \qquad (21)$$

where $dJ(\theta)/d\theta$ is a matrix whose i, j^{th} element is the derivative of J with respect to θ_{ij}. Solving for θ, one obtains

$$\theta(t) = \left[\sum_{k=t_0}^{t} \lambda^{t-k}\phi(k)\phi^T(k) + \nu M^T W_0 M\right]^{-1} \left[\sum_{k=t_0}^{t} \lambda^{t-k}\phi(k)y(k) + \nu M^T W_0 K(t)\right]. \qquad (22)$$

Note that the second term in the matrix inverse originates from the constraints. Without this term, the matrix could come close to singularity under low excitation conditions, resulting in a high sensitivity to noise. Conversely, the term may be adjusted to ensure the invertibility of the matrix, even in the presence of insufficient excitation. This prevention of singularities is known as *regularization* [11] and is the principle behind a number of successful batch and nonlinear regression techniques such as ridge regression [12] and the Levenberg Marquardt algorithm [13, 14]. In contrast to plain regularization, the algorithm (22) has an additional term not only in the matrix inverse on the left, but also in the matrix on the right. In [15], it was shown that this additional term provided further smoothing capabilities of the algorithm in the presence of noise.

3.2 Recursive and Sequential Implementations of the Modified Least Squares Algorithm

For the application to reconfigurable control, an implementation of the algorithm is needed that progressively computes new estimates of the parameters based on the latest data available. For that purpose, the main difficulty is the calculation of the matrix inverse. Note that the size of the matrix inverse is dependent on the number of regressor variables (*i.e.,* the dimension of ϕ), but not on the number of output variables (*i.e.,* the dimension of y). In other words, there is a considerable computational advantage to having several output variables share the same regressor. This advantage was exploited in the selection of the parameterization in section 2.2.

In [15], a *recursive* implementation of the algorithm was proposed which was similar to that of the standard recursive least-squares (RLS) algorithm and avoided the need to invert the matrix at every time instant. Because of the constraints, an approximate update of the matrix inverse had to be implemented, but the approximation was found to be satisfactory in reconfigurable flight control simulations [16]. An alternative approach, called here the *sequential* approach, consists in calculating the matrix inverse at periodic time intervals. The resulting algorithm is called the *modified sequential least-squares algorithm* (MSLS). Specifically, (22) is rewritten as

$$\theta(t) = R'^{-1}(t)s'(t) \tag{23}$$

with $R'(t)$ and $s'(t)$ given

$$\begin{aligned} R(t) &= \lambda R(t-1) + \phi(t)\phi^T(t) && (24) \\ R'(t) &= R(t) + \nu M^T W_0 M && (25) \\ s(t) &= \lambda s(t-1) + \phi(t)y(t) && (26) \\ s'(t) &= s(t) + \nu M^T W_0 k(t). && (27) \end{aligned}$$

R and s should be initialized at $R(0) = \epsilon\ I$, where ϵ is a small positive constant and $I \in \Re^{n\times n}$ is the identity matrix, and $s(0) = R(0)\theta(0)$, where $\theta(0)$ is an arbitrary vector of initial coefficients. To solve (23), it is useful to recognize that $R'(t)$ is a symmetric positive definite matrix (assuming a full rank matrix $M^T W_0 M$), and that it may therefore be decomposed into upper and lower triangular matrices via Cholesky factorization. The Cholesky decomposition yields

$$G(t)G^T(t)\theta(t) = s'(t), \qquad \text{or :} \qquad G(t)H(t) = s'(t). \tag{28}$$

Forward elimination is used to solve (28) for $H(t)$. Given $H(t)$, back-substitution will produce $\theta(t)$ from

$$G^T(t)\theta(t) = H(t). \tag{29}$$

The computational requirements of the sequential algorithm were compared to those of the recursive algorithm and to those of a sequential algorithm based on a Givens-rotation algorithm. The three algorithms (Cholesky, Givens, and Recursive) were evaluated on a variety of sample problems, simulation data, and flight data. The results of these algorithms were qualitatively similar, so that the selection was based on ease of implementation and computational efficiency. For a five-parameter identification problem (typical of the application under consideration), the recursive algorithm required approximately 255 multiplies, the Cholesky technique required approximately 330 multiplies, and the Givens approach required approximately 590 multiplies. The Cholesky method was chosen because it represented significant savings over the Givens approach and was only slightly more computationally demanding than the recursive approach. The major advantage of the sequential approach over the recursive implementation was that computational load could be adjusted by performing the matrix inverse at a time interval compatible with available resources.

4. OFF-LINE PARAMETER IDENTIFICATION RESULTS

4.1 Batch Identification Results

Evaluation of the identification algorithm was performed using *Lockheed Martin's* nonlinear 6DOF simulation of the F-16 and flight data provided by *Calspan.* First, exhaustive experiments were conducted using a batch form of the identification algorithm (LS). This algorithm is the same as the algorithm discussed in the previous section, but with $\lambda = 1$ and no constraints. The results revealed interesting issues. First, errors were found with simulation data due to misalignment of channels. Because parameters in the simulation were updated in a specific order, certain signals were not properly aligned with other signals dependent on the calling structure of the program modules. Identification algorithms are quite sensitive to data alignment and, interestingly, alignment problems were also encountered in the flight tests.

It was also found that the estimates of M_α provided by identification algorithm were significantly different from those originating from a *trim and linearize* procedure implemented in *Lockheed Martin*'s simulation code. The discrepancies were explained by the fact that the identification algorithm computed a "lumped" M_α that included leading-edge flap effects. These effects tend to stabilize the aircraft and move M_α towards the negative direction. It is an issue in such comparisons that identification often determines "lumped" parameters, that group together effects that may be considered different.

Table 1 compares the estimated stability and control derivatives with data alignment issues resolved and with leading-edge flap effects included. The derivatives provided by the trim and linearize procedure are compared to those obtained by the identification algorithm for simulated data and for flight test data. The flight condition is at Mach 0.78 and 20,000 *ft* altitude. From Table 1, batch LS parameter identification results are found in good agreement with the simulations's trim and linearization estimates of the longitudinal stability and control derivatives. The match between the simulation data and the flight data is also good, although there remained a difference in the parameter M_α that could not be explained. The difference in the parameter M_0 is not a significant one, because any discrepancy in the zero reference line of the tail surfaces would translate into a change in the parameter M_0 equal to the angular difference multiplied by $M_{\delta_{ts}}$. In other words, the difference in M_0 could be attributed to a bias in the calibration of the control surfaces' position of only 3 to 4 degrees.

4.2 Results with the Sequential Implementation

In this section, we compare RLS and MSLS identification results on the same flight test data as used for batch identification. The RLS algorithm is identical to the MSLS algorithm, except for the absence of temporal and spatial constraints. The objective of this section is to demonstrate the significant advantages that these constraints provide. The temporal constraints discourage large parameter excursions between samples, while the spatial constraints discourage variations from *a priori* or expected values.

Fig. 2 shows the estimates of $\theta_{q_{ts}}$ and $\theta_{q_{ta}}$ obtained using the RLS algorithm. In the figures, the solid lines correspond to the RLS parameter estimates, and the dotted lines correspond to the "true" parameters obtained from batch identification. In Fig. 2, the RLS algorithm is seen to estimate $\theta_{q_{ts}}$ with reasonable accuracy. However, the estimate of $\theta_{q_{ta}}$ provided by the RLS is much in error. Because the flight data consisted of pitch-axis maneuvering, the asymmetric elevon deflection was negligible and there was insufficient information to determine this parameter.

Fig. 3 shows MSLS parameter identification results for the same flight data. As before, the solid line corresponds to the parameter estimate, and the dotted line represents the "true" parameter. The dashed line is a *spatial* constraint. In Fig. 3, the MSLS estimate of $\theta_{q_{ts}}$ is slightly smoother than the corresponding RLS estimate, although the two parameter estimates are qualitatively similar. The MSLS estimate of θ_{qta} is markedly improved over the corresponding RLS parameter. Despite the absence of appreciable asymmetric elevon activity, the MSLS estimate is stable and well-contained, because temporal and spatial constraints serve to regularize θ_{qta} during periods of low excitation.

It could be argued that the constraint simply biases the parameter towards the true value, and that this value would not be known, in general. However, a spatial constraint was also applied to the symmetric tail contribution to pitching moment. Yet, because there was sufficient symmetric elevon activity, the MSLS ignored the spatial constraint and faithfully tracked the true parameter. In other words, the spatial constraint does not have to be equal to the true parameter for the identified parameter to converge correctly in the presence of sufficient excitation. It only helps to maintain the parameter to a reasonable value when there is not sufficient information to determine that parameter.

5. FLIGHT TEST RESULTS

5.1 VISTA/F-16 Implementation

The *VISTA/F-16* aircraft is sketched on Fig. 4. The aircraft contains a *Variable Stability System* (VSS) implemented on three *Rolm Hawk/32* computers. It was found that each of these Hawk/32 computers had approximately the throughput of an *Intel 80286* processor, so that limits on computational power were a major consideration. In a typical application, the VSS computes effector commands that make the VISTA responses imitate the behavior of some other aircraft. For this project, however, the VSS software was replaced by an adaptive control software. The self-designing control software was segmented so that its computations could be shared by the three Hawk computers. One of the Hawk computers was used for the parameter identification algorithm and, because of computational limitations, was updated at approximately 13Hz.

For safety, the VISTA digital flight control computer has logic that continuously monitors the VSS-computed commands and reverts control to the primary F-16 control laws if VSS commands are deemed unsafe. The logic is known as the *Vehicle Integrity Monitor* (VIM), and is responsible for ensuring that the VSS system does not violate any pre-specified flight envelope or structural safety limits. In those limits are structural limits that prevent any actuator commands from putting twist on the fuselage when the velocity exceeds 275 knots. Because a typical reconfiguration scenario is one in which asymmetric flaperon is used to counteract rolling moments generated by a tail command in the case of a tail failure, all the flight tests had to be limited to speeds below 275 knots.

5.2 Measurements

The identification algorithm requires the measurement of the state variables α, β, q, p, and r. As Fig. 4 shows, the VISTA F-16 aircraft is equipped with an AOA (angle of attack) cone, two AOS (angle of sideslip) cones, and rate gyros. Together, these sensors provide measurements of all the state variables.

The values of the control signals are also required. It was decided to use the measured positions instead of the commanded variables. The positions were provided by synchros connected to the actuators. In part, the choice of using actual *vs.* commanded control variables was due to the observation of a significant hysteretic behavior of the actuators. Symmetric and asymmetric actuator positions were computed back from the left and right positions. This computation was performed because the pseudo-effectors were the variables used by the control law, and because the use of pseudo-effectors allowed to impose certain useful limits on the parameters (see section 5.5).

The derivatives of the state variables are also needed for identification. In general, they may be calculated by filtered differentiation. However, in this application, it was found advantageous to use accelerometer measurements. Referring again to Fig. 4, note that the aircraft is equipped with three accelerometer stations (one at the pilot, and two slightly above and below the center of gravity). From those accelerometer measurements, the three rotational accelerations were reconstructed using standard formulas and the knowledge of the distances between the accelerometer stations. While accelerometer measurements are noisy, it was found that the noise had relatively little impact on the parameter identification algorithm, which performed the required smoothing.

The derivatives $\dot{\alpha}$ and $\dot{\beta}$ can also be reconstructed from acceleration measurements, referred to the center of gravity of the aircraft. In fact, the signal $\dot{\alpha} - q + \tan\beta\,(p\cos\alpha + r\sin\alpha) - g_L/(V_t\cos\beta)$ which appears in (11), (13) is the wind z-axis accelerometer signal at the center of gravity divided by $V_t\cos\beta$. In this application, complementary filters from the standard VSS software provided clean signals for $\dot{\alpha}$ and $\dot{\beta}$, and the signals were used for identification.

5.3 Constraints and Supervisory Functions

The identification algorithm incorporated a number of constraints on the parameters. Temporal constraints forced the parameters to be near previous values. Spatial constraints included constraints derived from flight mechanics, and *a priori* values. For the maneuvers flown in the flights tests, many of the A-matrix terms were constant or varied as simple functions of flight condition. Using simulation data, quadratic functions were calculated for the expected values of the A-matrix terms as functions of flight condition.

A variety of supervisory functions were also considered for the parameter identification. These functions included: (1) abrupt change detection, (2) adaptive levels of active noise injection, (3) removal of parameters associated with poor excitation, (4) freezing of the parameter estimates during periods of insufficient excitation, and (5) clamping of the parameter values. Eventually, the only supervisory function that was implemented was parameter clamping. This function was implemented as a precaution against unforeseen parameter divergence. The values at which the parameters were clamped were determined by considering a variety of conditions and obtaining the maximum and minimum values from simulation data. To allow for failure situations, the limits on the contribution of the asymmetric tail to the pitch equations were set to be ± 1/2 that of the symmetric tail. Likewise, the limits on the contribution of the symmetric tail to the roll/yaw equations were set to be ± 1/2 that of the asymmetric tail.

5.4 Failure Simulation

The software had a self-contained failure module which emulated the effect of various types of failures. Single or multiple failures could be triggered at desired times. The failure module intercepted the control law commands, modified them, and sent the modified commands to the actuators. For 100% effective (unimpaired) surfaces, the actuator commands were not modified by the failure simulation, and for 0% effective control surfaces, the effector was commanded to "float" parallel to the local flow, thus approximating a missing surface. A partially missing actuator was commanded between the floating position and the desired displacement.

Ineffective left and right tail surfaces were simulated by commanding the corresponding surface to the negative of the local angle of attack. The local angle of attack is different from the aircraft angle of attack due to downwash effects at the tail. The downwash at the tail stems from the aerodynamic interaction between the wings and the tail surfaces, particularly the vortex-sheet motion of the airflow behind the aircraft wing, and reduces the effective angle of attack at the tail. Data for the F-16, provided by researchers at *Wright Laboratory*, showed that the downwash variation with aircraft angle of attack was approximately linear within the restricted flight envelope allowed by the vehicle integrity monitor. This linear relationship was used for failure simulation.

Ineffective left or right flaperons were simulated by commanding the effectors to the negative of the aircraft angle of attack. Reduction in the effective angle of attack at the wings, owing to interaction of the fuselage with the wingbody, was assumed to be negligible, and the local angle of attack at the flaperon was taken to be the aircraft angle of attack. Similarly, an ineffective rudder was simulated by commanding the rudder to the aircraft angle of sideslip.

Prior to activation of the failure, the failure module simply passed actuator commands through, without modification. Upon activation, the failure was eased in, *i.e.,* effector positions were transitioned from their current values to those of the failed control surfaces over a period of several seconds. Ground simulation and flight test experiments led to a choice of a time constant such that the actual effectiveness was 50% of the desired value after 1 *sec.*, 75% after 2 *sec.*, 87.5% after 3 *sec.*, etc. While batch and piloted simulations indicated that the adaptive algorithm could readily compensate for abrupt (instantaneous) changes in effectiveness, the progressive transtion was required so that actuators were not driven at their rate limits and that safety (VIM) constraints were not violated during the transition from unfailed to failed operation.

In addition to modifying the actuator commands, the failure simulation needed to give to the parameter identification process the information that it would receive if the actuator had been displaced by the full commanded value. This result was achieved by adding to the measured actuator position the difference between the commanded position sent to the failure simulation module and the commanded position sent to the actuator. This simple adjustment was adequate because the actuator dynamics were fast and close to linear.

5.5 Parameter Identification Results

The *self-designing controller* (SDC) underwent flight testing during the period from May 14, 1996, to July 9, 1996. Five test flights were performed, each lasting approximately one hour. Due to the limited amount of flight time, the SDC tests did not employ a *build-up* approach whereby various components of the software and algorithm would have been tested sequentially. Rather, the test plan for the first flight called for the entire algorithm and associated software to be engaged; then, depending on the results, experiments of increasing or decreasing difficulty would be tried. This kind of approach was possible because of the safety provided the VISTA F-16 Vehicle Integrity Monitor module.

As for the simulations, a significant amount of time was spent making sure that the data was aligned and correct. Because the flight tests included the evaluation of the reconfigurable control law, considerable effort was also spent adjusting the desired control responses for flying qualities. The remaining time was spent addressing and resolving problems with the parameter identification. Few such problems were encountered, but during one of the flights, there was a divergent nose-down transient exhibited by the SDC. Post-flight analysis revealed that the pitch excursion resulted from a parameter going to zero. This parameter estimated the contribution of the symmetric elevon to the pitch acceleration. Inspection and analysis of the data revealed that the problem was due to small errors in the inertial cross-coupling compensation, which manifested itself during multi-axes maneuvers where the cross-couplings were significant. The pitch rate equation was the most susceptible to these problems because of large roll and yaw rates attained for relatively mild turns (recall Fig. 1).

To remedy the problem, two modifications of the parameter identification routines were implemented. Different forgetting factors were applied for the different parameters, and the forgetting factors as well as the spatial and temporal weighting matrices were optimized to avoid temporary excursions of the parameters to undesirable values. The new settings were obtained off-line using a guided-random search algorithm. Settings were chosen which minimized the weighted-sum-squared error between key estimates and true parameters in simulations.

A second modification to the identification software was an adjustment in the "clamping" routines that limited parameter estimates to predefined minimum and maximum values. These extrema were chosen to encompass all values that the "true" parameters could possibly take on at any combination of loading, mass property configurations, flight condition, and potential failure scenarios. Originally, the upper limit on the contribution of symmetric elevon to pitch acceleration had been set at 0.0. Note that this parameter is normally negative, so that an upper limit is in fact a lower limit on the effectiveness. For the parameter to be equal to zero, both left and right elevons would have to be simultaneously 0% effective. Because such a failure scenario is not likely and, further, would not allow any successful reconfiguration, the upper limit for the parameter was modified and set at a small negative value (-1.00). Other parameter limits were modified similarly. In subsequent flight tests, the parameters rarely encountered the limits, although the limits eliminated undesirable responses that occurred in such instances.

Parameter identification results, using the optimal settings, are shown in Fig. 5. A missing elevon was simulated. The solid lines show the estimated parameters and the dashed lines show the "true" parameters (the quotes refer to the fact that the true parameters are only estimates of the parameters based on other identification results). The progressive engagement of the failure is visible from the responses and the identified parameters are found to converge to the true parameters. Convergence of the asymmetric tail parameters is slower because of the lower level of activity in that signal. As expected, one finds that the pitching effectiveness of the symmetric tail deflection is halved after the failure. Also, the pitching effectiveness of the asymmetric deflection moves from zero to the same effectiveness as that of the symmetric deflection. An identical, but reversed picture, is observed for the rolling moment.

Fig. 6 shows the pitch rate and elevon commands during landing with a (simulated) 100% missing left elevon. The plot on the top shows the pitch rate and the desired pitch rate specified by the flying qualities model. The plot below shows the elevon positions. Time histories begin 35 sec. into the flight record, at the time of SDC engagement and failure activation. They terminate at touchdown approximately 150 sec. into the record. The figure shows that there were no significant transients upon activation of the impairment. The left elevon moved to a position equal to approximately one degree, and floated at the local angle of attack. The position of the right elevon moved in the opposite direction to maintain trim of the aircraft, and provided the moments needed to achieve stabilization and the desired pitch rate.

Despite the effects of a simulated failure being compounded by a significant crosswind (15 *kts.*) and gusty conditions during the landing, flying qualities were satisfactory for all axes, and, as noted by the flight crew, performance of the adaptive control law was consistent and predictable for the entire flight. To the authors' knowledge, this was the first time that an aircraft with simulated damage had been landed under reconfigurable control.

6. CONCLUSIONS

The identification of aircraft parameters for use in reconfigurable control poses significant problems. The problems are mostly due to the poor information content of the signals used for identification combined with the need for rapid adaptation. As a solution to this problem, this paper proposed a modified least-squares algorithm which incorporates temporal and spatial constraints. This novel on-line system identification technique was found to rapidly track time-varying parameters and to be robust to adverse conditions such as low excitation or correlated inputs. A significant part of the design was the selection of an appropriate model structure, which accounted for the nonlinear dynamics of importance with a minimal number of parameters. The choice of temporal and spatial constraints was also critical to the performance of the algorithm. A series of flight tests resulted in a landing under a simulated failure condition and on-line reconfigurable control, and demonstrated the success of the identification algorithm in determining the parameters of the aircraft in real-time.

7. ACKNOWLEDGEMENTS

The authors would like to express their sincere appreciation to Mr. Phil Chandler, Dr. Meir Pachter, and Mr. Mark Mears, of *Wright Laboratory*, for their contributions to the project. The test pilots were Mr. John Ball, Mr. Jeff Peer, Mr. Joseph Sweeney, and Maj. Kevin Christensen. Their feedback and enthusiasm for the project are gratefully acknowledged. This work was sponsored by the Directorate of Mathematical and Computer Sciences, Air Force Office of Scientific Research, Bolling AFB, DC, under SBIR Phase II Contract F49620-94-C-0087. The third author was also sponsored by the Air Force Office of Scientific Research, Air Force Materiel Command, USAF, under grant number F49620-95-1-0341. Dr. Marc Jacobs is thanked for his support as program manager. Technical supervision was supplied by the USAF Flight Dynamics Directorate, Wright Laboratory, Air Force Materiel Command, Wright-Patterson AFB, OH. Mr. Phil Chandler, in particular, is thanked for his supervision, leadership, and help in overcoming the obstacles towards completing the ambitious flight test program. *Lockheed Martin Tactical Aircraft Systems* (Ft. Worth, TX) and *Calspan SRL Corporation* (Buffalo, NY) participated as subcontractors and performed batch simulation evaluations and flight test support, respectively. Special thanks are due to Mr. David

Bodden, Mr. John Virnig, and Mr. Tom Landers. The U.S. Government is authorized to reproduce and distribute reprints of this paper for Governmental purposes notwithstanding any copyright notation. The views and conclusions contained in the paper are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Office of Scientific Research or the U.S. Government.

8. REFERENCES

[1] J. Urnes, R. Yeager, and J. Stewart, "Flight Demonstration of the Self-Repairing Flight Control System in a NASA F-15 Aircraft," presented at the *National Aerospace Electronics Conference,* Dayton, OH, 1990.

[2] R. Maine and J. Murray, "Application of Parameter Estimation to Highly Unstable Aircraft," *J. Guidance, Control, and Dynamics*, vol. 11, no. 3, pp. 213-219, 1988.

[3] K. Iliff, "Parameter Estimation for Flight Vehicles," *J. Guidance, Control, and Dynamics*, vol. 12, no. 5, pp. 609-622, 1989.

[4] M. Tischler, "System Identification Methods for Aircraft Flight Control Development and Validation," in *Advances in Aircraft Flight Control,* M. Tischler ed., Taylor & Francis, Bristol, PA, 1996.

[5] T. Trankle and S. Bachner, "Identification of a Nonlinear Aerodynamic Model of the F-14 Aircraft," *J. Guidance, Control, and Dynamics*, vol. 18, no. 6, pp. 1292-1297, 1995.

[6] E. Morelli and V. Klein, "Accuracy of Aerodynamic Model Parameters Estimated from Flight Test Data," *J. Guidance, Control, and Dynamics*, vol. 20, no. 1, pp. 74-80, 1997.

[7] E. Morelli, "Flight Test Validation of Optimal Input Design and Comparison to Conventional Inputs," *Proc. of the Atmospheric Flight Mechanics Conference,* New Orleans, LA, 1997, pp. 573-583.

[8] Anonymous, *Recommended Practice for Atmospheric and Space Flight Vehicle Coordinate Systems,* ANSI/AIAA Standard R-004-1992, American National Standard Institute, New York, NY, 1992.

[9] D. Ward and R. Barron, "A Self-Designing Receding Horizon Optimal Flight Controller," in *Proc. American Control Conference,* Seattle, WA, 1995, pp. 3490-3494.

[10] P. Chandler, M. Pachter, and M. Mears, "System Identification for Adaptive and Reconfigurable Control," *J. Guidance, Control, and Dynamics*, vol. 18, no. 3, pp. 516-524, 1995.

[11] T. Söderström and P. Stoica, *System Identification*, Prentice Hall, Englewood Cliffs, NJ, 1989.

[12] V. Cherkassky, J. Friedman, and H. Wechsler, *From Statistics to Neural Networks, Theory and Pattern Recognition Applications*, Springer-Verlag, Berlin, 1993.

[13] L. Ljung and T. Söderström, *Theory and Practice of Recursive Identification*, MIT Press, Cambridge, MA, 1983.

[14] G. Seber and C. Lee, *Advances in Neural Information Processing Systems*, Wiley and Sons, New York, NY, 1989.

[15] M. Bodson, "An Adaptive Algorithm with Information-Dependent Data Forgetting," in *Proc. American Control Conference*, Seattle, WA, 1995, pp. 3485-3489.

[16] M. Bodson and J. Groszkiewicz, "Multivariable Adaptive Algorithms for Reconfigurable Flight Control," *IEEE Trans. on Control Systems Technology,* vol. 5, no. 2, pp. 217-229, 1997.

9. APPENDIX

9.1 Relationship between the Parameters and the Dimensionless Derivatives

The forces and moments acting on the aircraft can be defined in terms of dimensionless aerodynamic coefficients, with: $T_x = \bar{q}SC_T$, $D = \bar{q}SC_D$, $Y_a = \bar{q}SC_{Y_a}$, $L = \bar{q}SC_L$, $\bar{L} = \bar{q}SbC_l$, $\bar{M} = \bar{q}S\bar{c}C_m$, $\bar{N} = \bar{q}SbC_n$, where $\bar{q} = \frac{1}{2}\rho V_t^2$ is the free-stream dynamic pressure, S is the wing reference area, b is the wing span, and $\bar{c}$ is the mean aerodynamic chord of the wing. The dimensionless coefficients are parameterized as follows

$$
\begin{aligned}
C_m &= C_{m_0} + C_{m_\alpha}\alpha + C_{m_{\dot\alpha}}\dot\alpha + C_{m_q}q + C_{m_{ts}}\delta_{ts} + C_{m_{ta}}\delta_{ta} && (30)\\
C_L &= C_{L_0} + C_{L_\alpha}\alpha + C_{L_{ts}}\delta_{ts} + C_{L_{ta}}\delta_{ta} && (31)\\
C_l &= C_{l_0} + C_{l_\beta}\beta + C_{l_p}p + C_{l_r}r + C_{l_{ts}}\delta_{ts} + C_{l_{ta}}\delta_{ta} + C_{l_{fa}}\delta_{fa} + C_{l_{rud}}\delta_{rud} && (32)\\
C_n &= C_{n_0} + C_{n_\beta}\beta + C_{n_p}p + C_{n_r}r + C_{n_{rud}}\delta_{rud} && (33)\\
C_{Y_a} &= C_{Y_0} + C_{Y_\beta}\beta + C_{Y_{rud}}\delta_{rud}. && (34)
\end{aligned}
$$

The dimensionless derivatives are related to the parameters in (16), (17) by the following equations

$$
\begin{bmatrix} \theta_{q_q} \\ \theta_{q_\alpha} \\ \theta_{q_{ts}} \\ \theta_{q_{ta}} \\ \theta_{q_0} \end{bmatrix}
= \frac{\bar{q}S\bar{c}}{I_{yy}}
\begin{bmatrix} C_{m_q} \\ C_{m_\alpha} + C_{m_{\dot\alpha}}\theta_{\alpha_\alpha} \\ C_{m_{ts}} + C_{m_{\dot\alpha}}\theta_{\alpha_{ts}} \\ C_{m_{ta}} + C_{m_{\dot\alpha}}\theta_{\alpha_{ta}} \\ C_{m_0} + C_{m_{\dot\alpha}}\theta_{\alpha_0} \end{bmatrix},
\quad
\begin{bmatrix} \theta_{\alpha_q} \\ \theta_{\alpha_\alpha} \\ \theta_{\alpha_{ts}} \\ \theta_{\alpha_{ta}} \\ \theta_{\alpha_0} \end{bmatrix}
= \frac{\bar{q}S}{mV_t\cos\beta}
\begin{bmatrix} C_{L_q}(\approx 0.0) \\ C_{L_\alpha} + \frac{T_x}{\bar{q}S}\frac{\sin\alpha}{\alpha} \\ C_{L_{ts}} \\ C_{L_{ta}} \\ C_{L_0} + \frac{mg_L}{\bar{q}S} \end{bmatrix},
\quad (35)
$$

$$
\begin{bmatrix} \theta_{p_p} \\ \theta_{p_r} \\ \theta_{p_\beta} \\ \theta_{p_{ts}} \\ \theta_{p_{ta}} \\ \theta_{p_{fa}} \\ \theta_{p_{rud}} \\ \theta_{p_0} \end{bmatrix}
= \frac{\bar{q}Sb}{\Gamma}
\begin{bmatrix} I_{zz}C_{l_p} + I_{xz}C_{n_p} \\ I_{zz}C_{l_r} + I_{xz}C_{n_r} \\ I_{zz}C_{l_\beta} + I_{xz}C_{n_\beta} \\ I_{zz}C_{l_{ts}} + I_{xz}C_{n_{ts}} \\ I_{zz}C_{l_{ta}} + I_{xz}C_{n_{ta}} \\ I_{zz}C_{l_{fa}} + I_{xz}C_{n_{fa}} \\ I_{zz}C_{l_{rud}} + I_{xz}C_{n_{rud}} \\ I_{zz}C_{l_0} + I_{xz}C_{n_0} \end{bmatrix},
\quad
\begin{bmatrix} \theta_{r_p} \\ \theta_{r_r} \\ \theta_{r_\beta} \\ \theta_{r_{ts}} \\ \theta_{r_{ta}} \\ \theta_{r_{fa}} \\ \theta_{r_{rud}} \\ \theta_{r_0} \end{bmatrix}
= \frac{\bar{q}Sb}{\Gamma}
\begin{bmatrix} I_{xx}C_{n_p} + I_{xz}C_{l_p} \\ I_{xx}C_{n_r} + I_{xz}C_{l_r} \\ I_{xx}C_{n_\beta} + I_{xz}C_{n_\beta} \\ I_{xx}C_{n_{ts}} + I_{xz}C_{l_{ts}} \\ I_{xx}C_{n_{ta}} + I_{xz}C_{l_{ta}} \\ I_{xx}C_{n_{fa}} + I_{xz}C_{l_{fa}} \\ I_{xx}C_{n_{rud}} + I_{xz}C_{l_{rud}} \\ I_{xx}C_{n_0} + I_{xz}C_{n_0} \end{bmatrix},
(36)
$$

$$\begin{bmatrix} \theta_{\beta_p} \\ \theta_{\beta_r} \\ \theta_{\beta_\beta} \\ \theta_{\beta_{ts}} \\ \theta_{\beta_{ta}} \\ \theta_{\beta_{fa}} \\ \theta_{\beta_{rud}} \\ \theta_{\beta_0} \end{bmatrix} = \frac{\bar{q}S}{mV_t} \begin{bmatrix} C_{Y_p}(\approx 0.0) \\ C_{Y_r}(\approx 0.0) \\ C_{Y_\beta} - \frac{T_x}{\bar{q}S}\cos\alpha\frac{\sin\beta}{\beta} \\ C_{Y_{ts}}(\approx 0.0) \\ C_{Y_{ta}}(\approx 0.0) \\ C_{Y_{fa}}(\approx 0.0) \\ C_{Y_{rud}} \\ C_{Y_0} + \frac{mg_{ya}}{\bar{q}S} \end{bmatrix}. \tag{37}$$

Some parameters are typically small, as indicated in the vectors. However, in order to share computations among the different variables of the longitudinal and lateral channels, all the coefficients were actually identified.

9.2 Linearized State-Space Model

To obtain a linear state-space model for control derivations, the nonlinear portion (NL) of the equations of motion given in (13), (14), must be linearized with respect to the states. Normally, one would require linearization with respect to the effector positions as well. However, the NL equations are functions only of the aircraft states. As a result

$$\begin{aligned} NL &= NL_{t_\ell} + \left.\frac{\partial NL}{\partial x}\right|_{t_\ell} (x - x_{t_\ell}) + H.O.T. \\ &\approx \left.\frac{\partial NL}{\partial x}\right|_{t_\ell} x + NL_{t_\ell} - \left.\frac{\partial NL}{\partial x}\right|_{t_\ell} x_{t_\ell} \end{aligned} \tag{38}$$

where t_ℓ is the time at which the linearization occurs. Note that it is not required for the aircraft to be linearized at a trim (or equilibrium) condition, and that the linearization may only be accurate over a short time horizon.

The original equations (11), (11), may be linearized as

$$\dot{x} = Ax + B\delta + d \tag{39}$$

where:

$$x^T = [q, \alpha, p, r, \beta], \qquad \delta^T = [\delta_{ts}, \delta_{ta}, \delta_{fa}, \delta_{rud}] \tag{40}$$

$$A = \left.\begin{bmatrix} \theta_{q_q} & \theta_{q_\alpha} & \frac{\partial(NL_q)}{\partial p} & \frac{\partial(NL_q)}{\partial r} & 0 \\ \{\theta_{\alpha_q}\} + \frac{\partial(NL_\alpha)}{\partial q} & \theta_{\alpha_\alpha} + \frac{\partial(NL_\alpha)}{\partial \alpha} & \frac{\partial(NL_\alpha)}{\partial p} & \frac{\partial(NL_\alpha)}{\partial r} & \frac{\partial(NL_\alpha)}{\partial \beta} \\ \frac{\partial(NL_p)}{\partial q} & 0 & \theta_{p_p} + \frac{\partial(NL_p)}{\partial p} & \theta_{p_r} + \frac{\partial(NL_p)}{\partial r} & \theta_{p_\beta} \\ \frac{\partial(NL_r)}{\partial q} & 0 & \theta_{r_p} + \frac{\partial(NL_r)}{\partial p} & \theta_{r_r} + \frac{\partial(NL_r)}{\partial r} & \theta_{r_\beta} \\ 0 & \frac{\partial(NL_\beta)}{\partial \alpha} & \{\theta_{\beta_p}\} + \frac{\partial(NL_\beta)}{\partial p} & \{\theta_{\beta_r}\} + \frac{\partial(NL_\beta)}{\partial r} & \theta_{\beta_\beta} + \frac{\partial(NL_\beta)}{\partial \beta} \end{bmatrix}\right|_{t_\ell} \tag{41}$$

$$B = \left.\begin{bmatrix} \theta_{q_{ts}} & \theta_{q_{ta}} & 0 & 0 \\ \theta_{\alpha_{ts}} & \theta_{\alpha_{ta}} & 0 & 0 \\ \theta_{p_{ts}} & \theta_{p_{ta}} & \theta_{p_{fa}} & \theta_{p_{rud}} \\ \theta_{r_{ts}} & \theta_{r_{ta}} & \theta_{r_{fa}} & \theta_{r_{rud}} \\ \{\theta_{\beta_{ts}}\} & \{\theta_{\beta_{ta}}\} & \{\theta_{\beta_{fa}}\} & \theta_{\beta_{rud}} \end{bmatrix}\right|_{t_\ell} \tag{42}$$

and:

$$d = \left.\begin{bmatrix} \theta_{q_0} + NL_q - \frac{\partial(NL_q)}{\partial p}p - \frac{\partial(NL_q)}{\partial r}r \\ \theta_{\alpha_0} + NL_\alpha - \frac{\partial(NL_\alpha)}{\partial q}q - \frac{\partial(NL_\alpha)}{\partial \alpha}\alpha - \frac{\partial(NL_\alpha)}{\partial p}p - \frac{\partial(NL_\alpha)}{\partial r}r - \frac{\partial(NL_\alpha)}{\partial \beta}\beta \\ \theta_{p_0} + NL_p - \frac{\partial(NL_p)}{\partial q}q - \frac{\partial(NL_p)}{\partial p}p - \frac{\partial(NL_p)}{\partial r}r \\ \theta_{r_0} + NL_r - \frac{\partial(NL_r)}{\partial q}q - \frac{\partial(NL_r)}{\partial p}p - \frac{\partial(NL_r)}{\partial r}r \\ \theta_{\beta_0} + NL_\beta - \frac{\partial(NL_\beta)}{\partial \alpha}\alpha - \frac{\partial(NL_\beta)}{\partial p}p - \frac{\partial(NL_\beta)}{\partial r}r - \frac{\partial(NL_\beta)}{\partial \beta}\beta \end{bmatrix}\right|_{t_\ell} . \tag{43}$$

Only the non-zero partials of the nonlinear terms have been included in the matrices. Additionally, parameters that are small have been put in {braces}. The partials of the nonlinear terms can be derived from (13), (14), and are given by

$$\frac{\partial(NL_q)}{\partial p} = \frac{(I_{zz} - I_{xx})}{I_{yy}}r - 2p\frac{I_{xz}}{I_{yy}}, \qquad \frac{\partial(NL_q)}{\partial r} = \frac{(I_{zz} - I_{xx})}{I_{yy}}p + 2r\frac{I_{xz}}{I_{yy}} \tag{44}$$

$$\frac{\partial(NL_\alpha)}{\partial q} = 1.0, \qquad \frac{\partial(NL_\alpha)}{\partial \alpha} = \tan\beta\,(p\sin\alpha - r\cos\alpha) + \frac{1}{V_t\cos\beta}\left(\frac{\partial g_L}{\partial \alpha}\right) \tag{45}$$

$$\frac{\partial(NL_\alpha)}{\partial p} = -\tan\beta\cos\alpha, \qquad \frac{\partial(NL_\alpha)}{\partial r} = -\tan\beta\sin\alpha \tag{46}$$

$$\frac{\partial(NL_\alpha)}{\partial\beta} = -\sec^2\beta\left(p\cos\alpha + r\sin\alpha - \frac{g_L}{V_t}\sin\beta\right) \tag{47}$$

$$\frac{\partial(NL_p)}{\partial q} = \frac{I_{xz}\left(I_{xx} - I_{yy} + I_{zz}\right)}{\Gamma}p + \frac{\left(I_{zz}\left(I_{yy} - I_{zz}\right) - I_{xz}^2\right)}{\Gamma}r \tag{48}$$

$$\frac{\partial(NL_p)}{\partial p} = \frac{I_{xz}\left(I_{xx} - I_{yy} + I_{zz}\right)}{\Gamma}q, \qquad \frac{\partial(NL_p)}{\partial r} = \frac{\left(I_{zz}\left(I_{yy} - I_{zz}\right) - I_{xz}^2\right)}{\Gamma}q \tag{49}$$

$$\frac{\partial(NL_r)}{\partial q} = \frac{\left(I_{xx}\left(I_{xx} - I_{yy}\right) + I_{xz}^2\right)}{\Gamma}p - \frac{I_{xz}\left(I_{xx} - I_{yy} + I_{zz}\right)}{\Gamma}r \tag{50}$$

$$\frac{\partial(NL_r)}{\partial p} = \frac{\left(I_{xx}\left(I_{xx} - I_{yy}\right) + I_{xz}^2\right)}{\Gamma}q, \qquad \frac{\partial(NL_r)}{\partial r} = -\frac{I_{xz}\left(I_{xx} - I_{yy} + I_{zz}\right)}{\Gamma}q \tag{51}$$

$$\frac{\partial(NL_\beta)}{\partial\alpha} = p\cos\alpha + r\sin\alpha + \frac{1}{V_t}\left(\frac{\partial g_{Y_a}}{\partial\alpha}\right), \qquad \frac{\partial(NL_\beta)}{\partial p} = \sin\alpha \tag{52}$$

$$\frac{\partial(NL_\beta)}{\partial r} = -\cos\alpha, \qquad \frac{\partial(NL_\beta)}{\partial\beta} = \frac{1}{V_t}\left(\frac{\partial g_{Y_a}}{\partial\beta}\right) \tag{53}$$

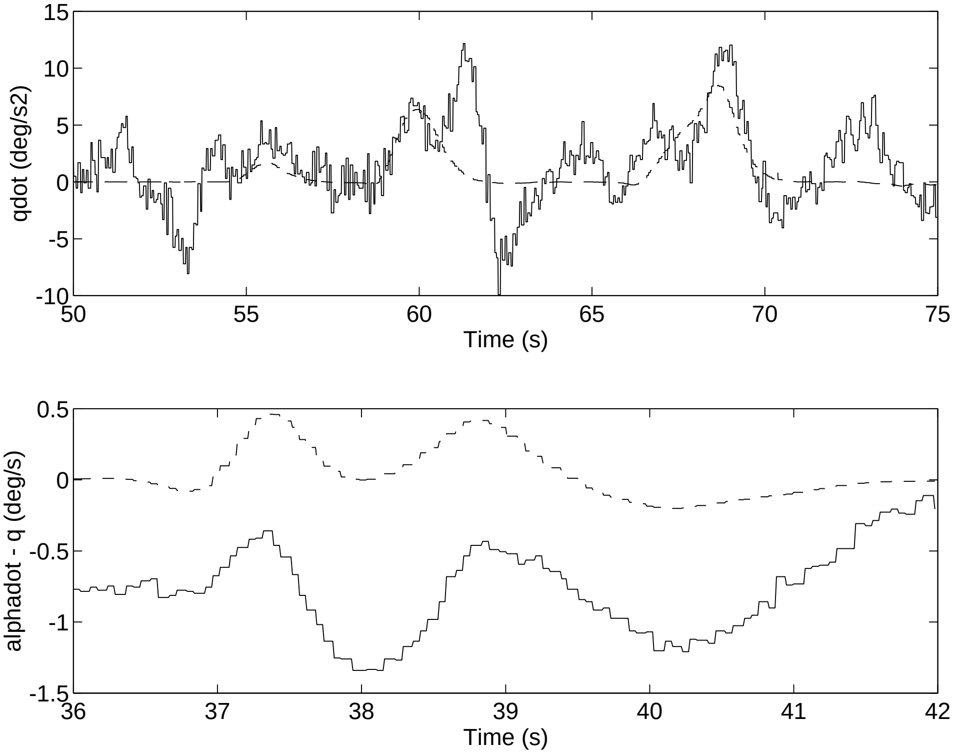

Figure 1: Contribution of Nonlinear Effects to the Longitudinal Responses

Table 1: Longitudinal Stability and Control Derivatives

Parameter	Trim & Linearize (Sim.)	Batch LS (Sim. Data)	Batch LS (Flight Data)
M_Q	-0.64	-0.43	-0.42
M_α	-1.62	-1.91	-3.09
$M_{\delta_{ts}}$	-14.3	-14.0	-12.6
M_0	-23.6	-22.2	18.7
Z_Q	0.99	1.00	0.99
Z_α	-0.91	-0.91	-0.88
$Z_{\delta_{ts}}$	-0.14	-0.14	-0.13
Z_0	2.08	2.09	2.55

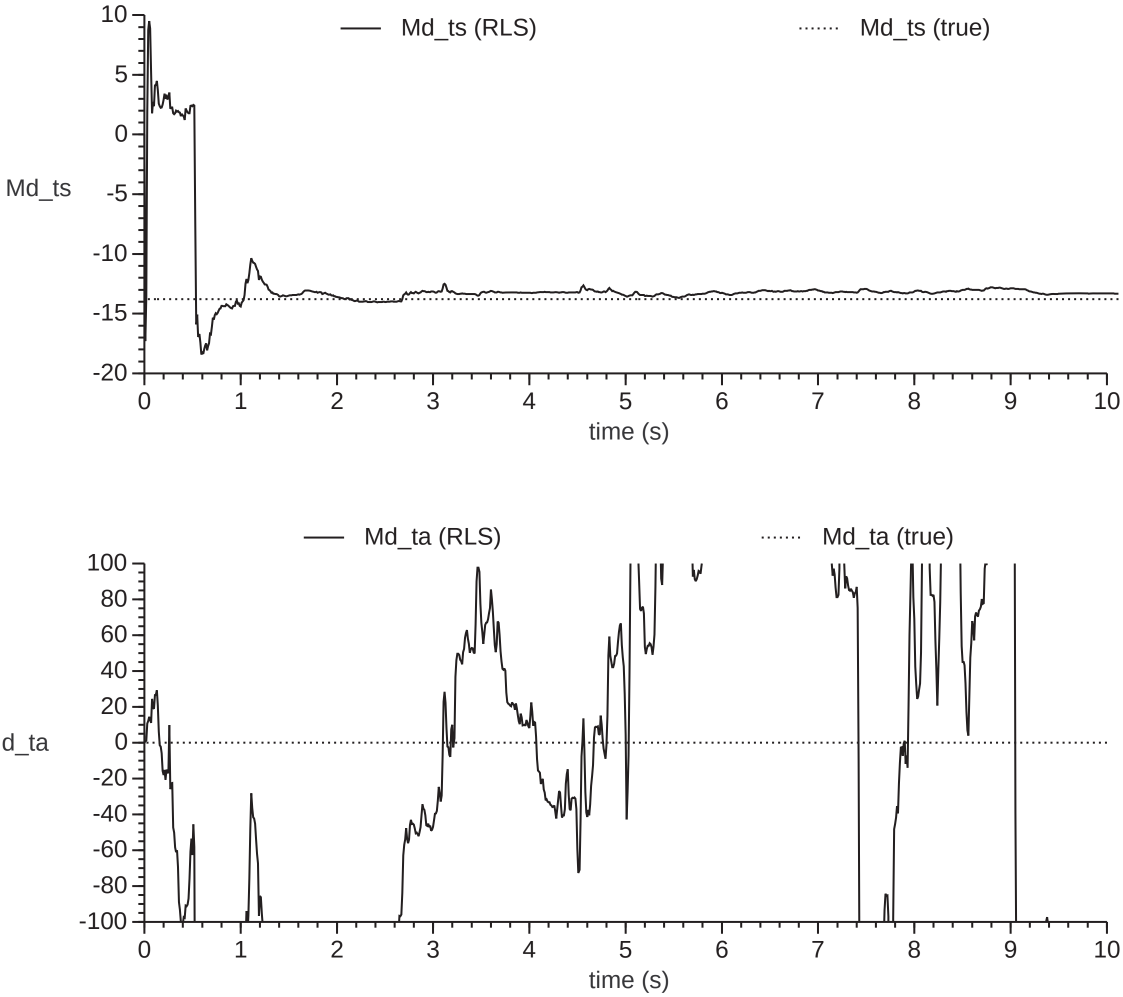

Figure 2: RLS Estimates of $\theta_{q_{ts}}$ and $\theta_{q_{ta}}$

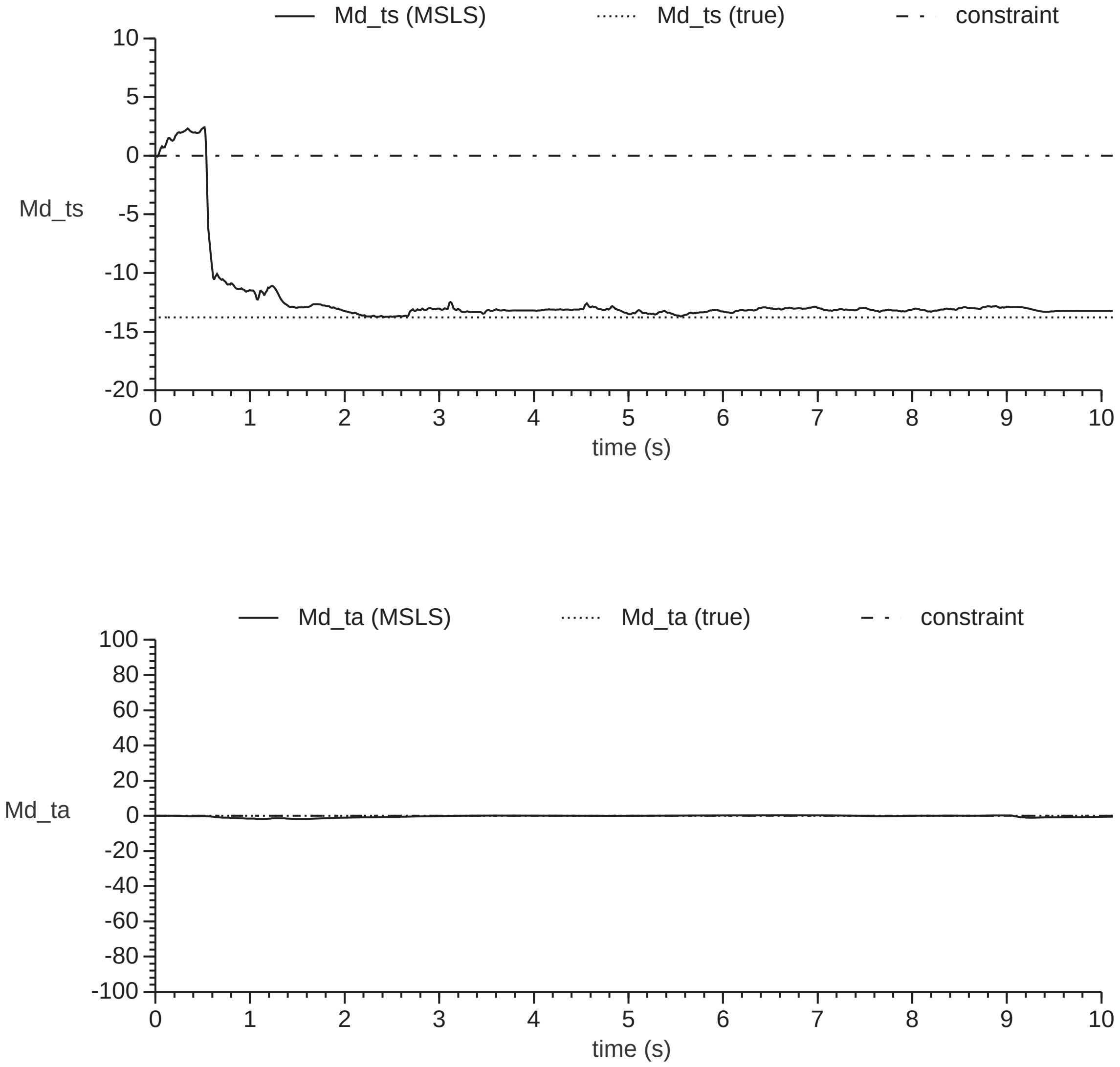

Figure 3: MSLS Estimates of $\theta_{q_{ts}}$ and $\theta_{q_{ta}}$

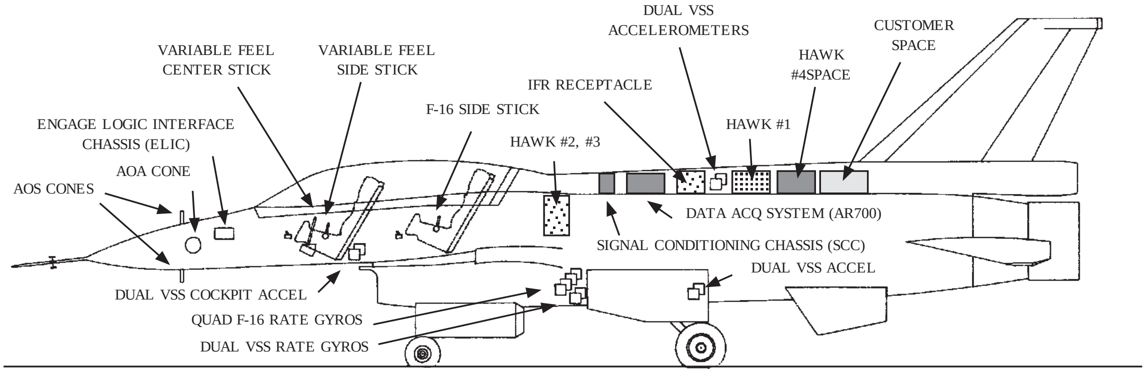

Figure 4: VISTA F-16 Aircraft

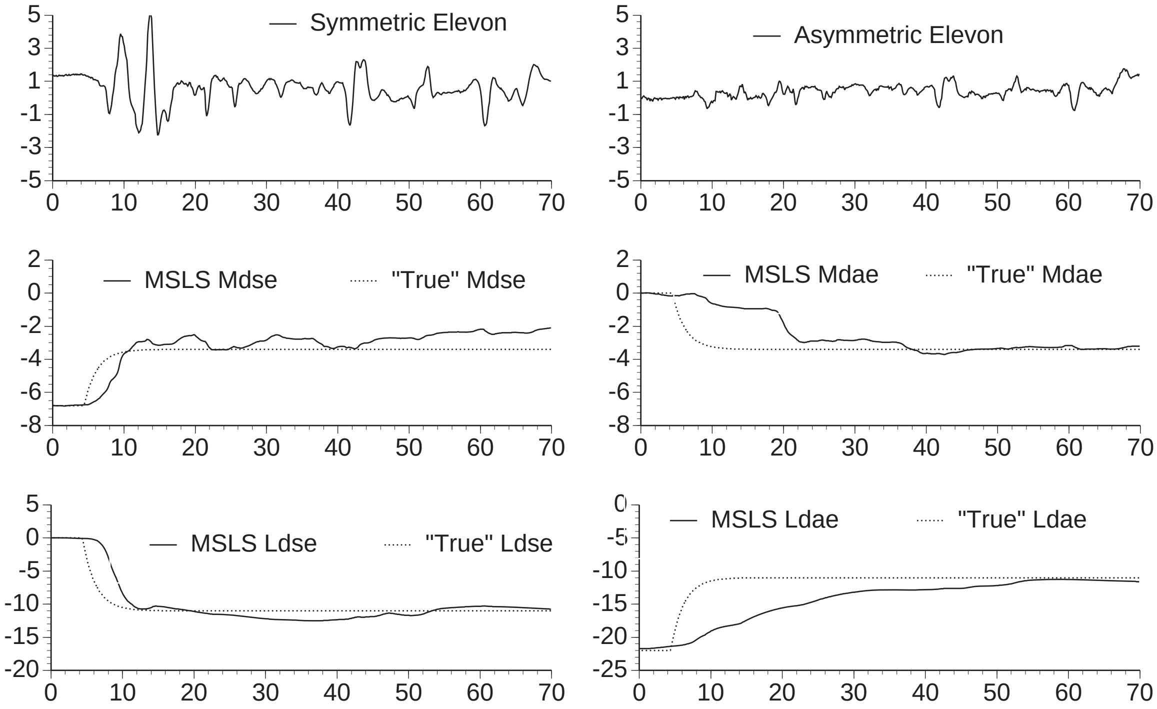

Figure 5: Elevon Control Derivatives from Flight Data

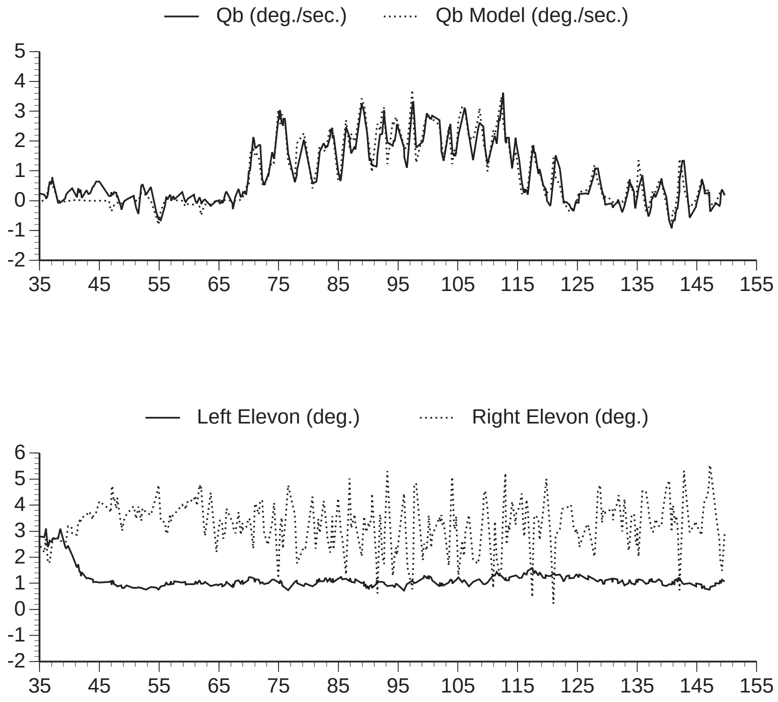

Figure 6: SDC Landing with Missing Left Elevon