JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 16, No. 4, July-August 1993

Inverse Simulation of Large-Amplitude Aircraft Maneuvers

C. Gao* and R. A. Hess†
University of California, Davis, Davis, California 95616

Inverse simulation techniques are computational methods that determine the control inputs to a dynamic system that will produce desired system outputs. Such techniques can be useful tools for the analysis and evaluation of problems associated with maneuvering flight. As opposed to current inverse simulation methods that require numerical time differentiation in their implementation, the proposed technique is essentially an integration algorithm. It is applicable to cases where the number of inputs equals or exceeds the number of constrained outputs. The algorithm is exercised in determining the trim conditions and then the control inputs that force a nonlinear model of an F-16 fighter to complete large-amplitude maneuvers.

Introduction

INVERSE simulation is a technique for determining the control inputs that will enable a dynamic system, such as an aircraft, to produce desired outputs. Typically, these desired outputs describe specific maneuvers that the aircraft is required to perform. Solutions to inverse problems, particularly ones involving nonlinear vehicle dynamics, tend to be computationally intensive and, until recently, were impractical for all but the simplest problems. Indeed, the hiatus between some of the earliest discussions of inverse problems[1] and recent results[2-5] is attributable to the earlier lack of powerful computers and associated software. With such computers and software, however, interest in inverse problems has increased significantly.

One area in which inverse simulation would appear to have considerable promise is in the study of high-angle-of-attack maneuvering flight. This flight regime involves highly nonlinear aerodynamics and represents a significant challenge to the flight control engineer, both from the standpoint of determining appropriate control effectors[6] and in deriving and evaluating appropriate control laws.[7] The research to be described involves the development and application of an inverse simulation algorithm using a full nonlinear aerodynamic model of an F-16 aircraft.[8] Large disturbance maneuvers such as rolling about the aircraft x-body axis will be discussed. The inverse algorithm to be discussed is an "integration" type algorithm that is computationally efficient enough to be used on an IBM PC-AT.

Inverse Simulation Algorithm

Although the inverse simulation algorithm to be exercised has been discussed in the literature,[5] a review will be undertaken here.

Nominal Solutions

The input-output relationship for a nonlinear dynamic system such as an aircraft can be represented as follows:

$$y\left[(k+1)T\right] = G\left[u(kT)\right] \qquad k = 0, 1, 2, \ldots, (n_{\text{end}} - 1) \quad (1)$$

where $y[(k+1)T]$ and $u(kT)$ represent the output and input vectors, respectively, $G[\ ]$ represents a mapping function, and T represents the discretization interval. Now assume that some desired $y[(k+1)T]$ is specified and denoted $y_D[(k+1)T]$. Then Eq. (1) can be rewritten

$$G\left[u(kT)\right] - y_D\left[(k+1)T\right] = \mathbf{0} \quad (2)$$

Newton's method[9] can be employed to solve the nonlinear algebraic equation (2) in iterative fashion, i.e., an error function $F_E[u_n(kT)]$ can be defined as

$$F_E\left[u_n(kT)\right] = G\left[u_n(kT)\right] - y_D\left[(k+1)T\right] \quad (3)$$

where n represents the iteration number in the solution of Newton's method. Thus, as $n \to \infty$, $F_E \to 0$. For the system of Eqs. (2) and (3), Newton's method yields the following solution for the $n+1$ value of $u(kT)$:

$$u_{n+1}(kT) = u_n(kT) - \left(J\left\{G\left[u_n(kT)\right]\right\}\right)^{-1} F_E\left[u_n(kT)\right] \quad (4)$$

where $J\{\ \}$ represents the Jacobian matrix. The element of the Jacobian in the ith row and jth column is defined as $\partial y_i[(k+1)T]/\partial u_j(kT)$. In general, these partial derivatives cannot be evaluated analytically. In the algorithm discussed here, they are approximated in a simulation of the vehicle at each discretization interval by perturbing each control individually and determining the change in each output over the next interval. More explicitly,

$$\frac{\partial y_i\left[(k+1)T\right]}{\partial u_j(kT)} \approx \frac{y_i(u_j + \Delta u)|_{(k+1)T} - y_i(u_j)|_{(k+1)T}}{\Delta u_j} \quad (5)$$

where Δu_j represents the perturbation in u_j at kT. It should be noted that in Ref. 5 the last term in the numerator of Eq. (5) was incorrectly written as being evaluated at $t = kT$. Note that integration of the system equations occurs (forward simulation) in determining the change in each output. The integration step size Dt is usually much smaller than the discretization step size T.

Redundant Solutions

As discussed in the preceding, the physical interpretation of the Jacobian matrix J in Eq. (4) is the estimated relation between the control input vector $u \in R^p$ and the output vector $y \in R^q$ where $p = q$. This relation is expressed simply as

$$y = Ju \quad (6)$$

The iteration algorithm of Eq. (4) will fail if J is degenerate or if J is "flat" ($p > q$), since, in both cases, J^{-1} does not exist. These cases are referred to as "redundant" here since

Received July 23, 1991; presented as Paper 91-2861 at the AIAA Atmospheric Flight Mechanics Conference, New Orleans, LA, Aug. 12-14, 1991; revision received Aug. 4, 1992; accepted for publication Sept. 1, 1992.

*Graduate Student, Department of Mechanical, Aeronautical, and Materials Engineering; currently Associated Researcher, Chung Shien Institute of Science and Technology, Taiwan, Republic of China.

†Professor, Department of Mechanical, Aeronautical, and Materials Engineering. Associate Fellow AIAA.

they arise from having more controls than outputs. It is necessary to propose an alternative solution. The general solution to Eq. (6) can be written as

$$u = J^{-}y + (I - J^{-}J)z \tag{7}$$

where J^{-} is any generalized inverse defined by $JJ^{-}J = J$, and z is an arbitrary vector with proper dimension.[9] The first term on the right-hand side of Eq. (7) represents the component of u in the row space of J, and the second term represents any component in the null space of J. The most useful and meaningful solution of Eq. (7) is obtained by replacing J^{-} with J^{+}, the Moore-Penrose generalized inverse or the pseudoinverse, which gives the minimum norm, least-squares solution. When J has full row rank,

$$J^{+} = J^{T}(JJ^{T})^{-1} \tag{8}$$

When J has less than full row rank, J^{+} can be computed by singular value decomposition (SVD),[10] i.e.,

$$J^{+} = V\Sigma^{+}U^{T} \tag{9}$$

where $V(p \times p)$ and $U(q \times q)$ are orthogonal matrices, and Σ has the diagonal form

$$\Sigma = \begin{vmatrix} \sigma_1 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & \sigma_q & \cdots & 0 \end{vmatrix} \tag{10}$$

and

$$\Sigma^{+}(i,j) = \begin{cases} 0 & \text{if} \quad \Sigma(j,i) = 0 \\ \dfrac{1}{\sigma_k} & \text{if} \quad \Sigma(j,i) = \sigma_k, \qquad \sigma_k \neq 0 \end{cases} \tag{11}$$

The optimal least-squares solution of any system is the vector u defined by the equation $u = J^{+}y$. This solution must satisfy two conditions: 1) Ju equals the projection of y onto the column (output) space of J, and 2) u lies in the row (input) space of J. The row space and null space of J are orthogonal complements. The pseudoinverse of J combines the effects of two separate steps: It projects y onto the column space of J and then finds the *only* vector u in the row space that solves $Ju = y$. So if one replaces Eq. (7) by

$$u = J^{+}y \tag{12}$$

the solution u is in the sense that the null space component of the input vector will be completely suppressed.

Now one can return to the iterative procedure in Eq. (4) and replace J^{-1} by J^{+}. The SVD of the Jacobian matrix was obtained herein by using the subroutine DSVD from LINPACK.[11] Thus, the inverse simulation algorithm searches for the step control inputs $u(kT)$ with minimum corrections at each iteration and with no component in the null space of J.

The property of uniqueness of the inverse solution will depend on the invertibility of the Jacobian matrix. For the nominal case (square Jacobian), if J is full rank during the iterative searching process, then the inverse solution for the defined output is unique. However, if J is singular or if it is a redundant case, then the inverse solution is not unique.

Discussion

The preceding solution method is referred to as an "integration" inverse solution since it uses the equations of motion in a forward fashion and the only differentiation that occurs is that shown in Eq. (5). This is a differentiation with respect to control inputs rather than time.

In terms of solution accuracy and sensitivity, the integration inverse method appears to be superior to "differentiation" methods[2-4] that require time differentiation of the Euler angles and their derivatives throughout the solution process. The necessity of calculating the derivatives of intermediate state variables in iterative fashion in these latter approaches constitutes their primary disadvantage in terms of accuracy and computational effort.

The inverse simulation process begins with the definition of a desired output vector time sequence $y_D[(k+1)T]$ that describes the maneuver to be studied. The vector thus defined must, of course, lie in a reachable domain, i.e., it must be able to be generated by physically realizable control inputs. The characteristics of the $y_D[(k+1)T]$ can obviously have profound effects on the required control inputs. Whereas forward simulation can generally be described as a process by which applied inputs are integrated and hence smoothed, inverse simulation, by its nature, is a process by which desired outputs are effecively differentiated. Thus, y_D must possess continuous derivatives of sufficiently high order to insure continuity in the resulting required control inputs.

Input Smoothing

In some of the inverse simulations to be discussed here and in those in previous work,[5] the control inputs exhibited high-frequency oscillations superimposed on a low-frequency waveform. This was especially evident in "off-axis" control inputs. It is hypothesized that this phenomenon may be caused by a multiple solution problem or by numerical inaccuracy for small control inputs. For example, consider the case where a solution of Eq. (2) lies near a relative minimum of $F_E[u_n(kT)]$. It is possible that another solution may exist in the neighborhood of the minimum and that the solution of Eq. (2) by way of Eq. (4) may alternate between these solutions for successive time steps in the inverse simulation. Because these input oscillations were of such high frequency and are filtered by the vehicle dynamics, it is reasonable to replace the oscillatory inputs with filtered ones in determining realistic control inputs that a pilot might generate. For the purpose of display, the oscillations were removed from the solutions by using a fifth-order Butterworth filter. This digital low-pass filter had a cutoff frequency of 10 rad/s. The control inputs were passed through the filter twice (forward and backward) to avoid time shifting in the output data.

F-16 Fighter Example

Vehicle Model Description

In 1979, Nguyen et al.[8] conducted a real-time piloted simulation to evaluate the high-angle-of-attack (AOA) characteristics of the F-16 fighter. The simulation was conducted on the NASA Langley differential maneuvering simulator with particular emphasis on the effects of various levels of longitudinal relaxed static stability. The aerodynamic data used in the simulation were based on low-speed wind-tunnel tests of subscale models. The simulation model of an F-16 fighter discussed in Ref. 8 is employed herein as a vehicle for exercising the inverse simulation methodology outlined in the preceding section.

The primary aerodynamic controls include deflections of the stabilator δ_h, aileron δ_a, and rudder δ_r. In addition, wing leading-edge flap deflection δ_{lef} was included as was engine thrust T_h. The leading-edge flap is a high-lift device and is important to enhancing the maneuverability at high AOA. The leading-edge flap deflection was scheduled with AOA and the ratio of dynamic to static pressure $\bar{q}/p_s$ according to

$$\delta_{lef} = 1.38\frac{(2s+7.25)}{(s+7.25)}\alpha - 9.05\frac{\bar{q}}{p_s} + 1.45 \text{ deg} \tag{13}$$

The leading-edge flap actuator was modeled as a first-order lag of 0.136 s. Maximum flap deflection was 25 deg. Assuming incompressible flow for the low airspeed flight conditions, the

static pressure p_s and air density were obtained from standard atmosphere tables.

The model in Ref. 8 also used differential deflection of the stabilator in combination with aileron inputs for roll control. However, the stability derivatives associated with this differential stabilator input were not included in the reference. Finally, the speed brake and nonlinear engine model included in Ref. 8 were not used in this analysis.

To summarize, the four control inputs used in the inverse simulation were symmetric stabilator deflection δ_h, rudder deflection δ_r, aileron deflection δ_a, and engine thrust T_h.

The 6-degree-of-freedom equations of motion employed 48 nondimensional aerodynamic derivative parameters that are nonlinear functions of angle of attack α, angle of sideslip β, and stabilator angle δ_h in tabular form and contain a total of 13,000 data points. The data include an AOA range of -20 to $+90$ deg and a sideslip angle range of -30 to $+30$ deg. Effects of Mach number, Reynolds number, or aeroelastic effects were not included in the model. For further description of the equations of motion and aerodynamic data, the reader is referred to Ref. 8.

Trimming Procedure

The output definition for an inverse simulation always begins from a condition of trimmed flight, here interpreted as a flight condition with zero angular acceleration and zero linear acceleration of the vehicle center of mass. This means

$$p = q = r = \phi = \psi = \beta = a_y = \delta_r = \delta_a = 0$$
$$\theta = \alpha \tag{14}$$

where a_y is the y-body axis component of the acceleration of the vehicle center of mass. The aircraft is considered "trimmed" when, with the conditions of Eq. (14),

$$a_x = a_z = M = 0 \tag{15}$$

where M represents the pitching moment. These quantities are nonlinear algebraic functions of α, δ_h, δ_{lef}, T_h, velocity, and altitude. By assigning the initial vehicle altitude and velocity (or Mach number), the trimming procedure must find T_h, α, and δ_h such that Eq. (15) is satisfied. A Newton-Raphson iteration algorithm is again employed. Let

$$f = \begin{bmatrix} a_x(x) \\ a_z(x) \\ M(x) \end{bmatrix} \tag{16}$$

where

$$x = \begin{bmatrix} T_h \\ \alpha \\ \delta_h \end{bmatrix} \tag{17}$$

A linear approximation to the function in Eq. (16) can be given by

$$f(x) = f(x_0) + F(x_0)(x - x_0) \tag{18}$$

where $F(x_0)$ is the Jacobian of f evaluated at x_0. Since it is desired to make $f = 0$, Newton's method is again employed as

$$x_{n+1} = x_n - \left[F(x_n)\right]^{-1} f(x_n) \tag{19}$$

where $F(x_n)$ is obtained by a "perturbation" method similar in philosophy to that of Eq. (5) except that algebraic rather than differential equations are involved.

Inverse Simulations

The inverse simulation of three maneuvers will now be demonstrated: a high AOA, bank-to-bank rolling maneuver about the aircraft x-body axis, a low AOA coordinated turn, and a 360-deg roll about the x-body axis. Unless stated otherwise, $T = 0.10$ s and $Dt = 0.01$ s. The trim altitude for all maneuvers was approximately 9000 m (30,000 ft).

Bank-to-Bank Rolling About x-Body Axis

A sinusoidal desired bank command was used in this maneuver. Here, the desired or constrained output variables are

$$V_d(kT) = 87.9 \text{ m/s}$$
$$\theta_d(kT) = \alpha\left[(k-1)T\right]$$
$$\psi_d(kT) = 0 \tag{20}$$
$$\phi_d(kT) = \Phi_M \cdot \sin\left[\Omega_r(kT)\right]$$

where Φ_M is the amplitude of the roll angle (10 deg for $\alpha_{\text{trim}} = 24$ deg) and $\Omega_r = 0.1$ Hz. Note that $\Theta = \alpha$ to maintain level flight. The angle of attack just listed corresponds to the initial trim angle of attack for the constrained velocity in Eq. (20). This angle of attack is not constrained and varies throughout the maneuver. Inverse and forward simulation results are shown in Fig. 1. In these and the remaining figures, the following convention is used. In the control variable time histories, solid lines represent smoothed inputs (used in the forward simulation) whereas the dotted lines represent the results of inverse simulation. In the time histories for the remaining variables, solid lines represent the results of the forward simulation with the smoothed inputs, whereas dotted lines represent inverse simulation results. As can be seen, the forward simulation results match the inverse results quite well.

Coordinated Turn

Here the coordinated turn is defined in terms of desired bank angle time histories with zero sideslip, much as a pilot

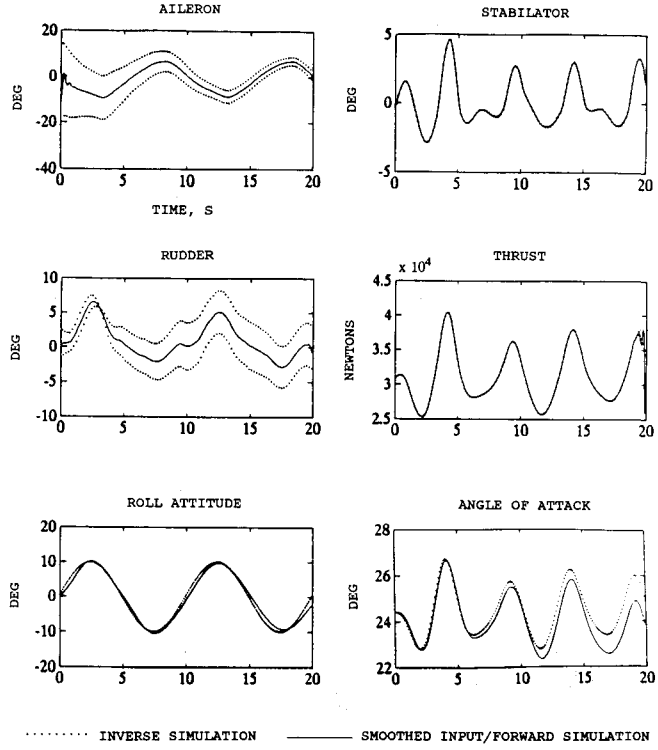

Fig. 1 Inverse and forward simulation results for bank-to-bank rolling maneuver.

would define it. The desired or constrained bank angle is given by

$$\phi_d(kT) = \begin{cases} \dfrac{\phi_s}{(16\cdot 57.3)}\left[f_1(kT)\right] \text{ rad} & \text{for } 0 \le kT < 5 \text{ s} \\ \dfrac{\phi_s}{57.3} \text{ rad} & \text{for } 5 \le kT < 15 \text{ s} \\ \dfrac{\phi_s}{57.3}\left[f_2(kT)\right] \text{ rad} & \text{for } 15 \le kT \le 20 \text{ s} \end{cases} \quad (21)$$

where

$$f_1(kT) = \left[\cos 3\pi\left(\frac{kT}{5}\right) - 9\cos\pi\left(\frac{kT}{5}\right) + 8\right]$$

$$f_2(kT) = \left\{1 - \frac{1}{16}\left[\cos 3\pi\frac{(kT-15)}{5} - 9\cos\pi\frac{(kT-15)}{5} + 8\right]\right\} \quad (22)$$

where $\phi_s = 60$ deg for $\alpha_{\text{trim}} = 7.9$ deg. The remaining three constraint equations are

$$\theta_d(kT) = \theta_{\text{trim}}$$

$$\beta_d(kT) = 0 \quad (23)$$

$$V_d(kT) = 152 \text{ m/s}$$

The inverse and forward simulation results are shown in Fig. 2.

360-Deg Roll About x-Body Axis

This maneuver is defined by the following desired or constrained output variables:

$$\theta_d(kT) = \theta_{\text{trim}}$$

$$\psi_d(kT) = 0 \quad (24)$$

$$V_d(kT) = 152 \text{ m/s}$$

$$\phi_d(kT) = \frac{2\pi}{16}\left[\cos 3\pi\left(\frac{kT}{10}\right) - 9\cos\pi\left(\frac{kT}{10}\right) + 8\right] \text{ rad}$$

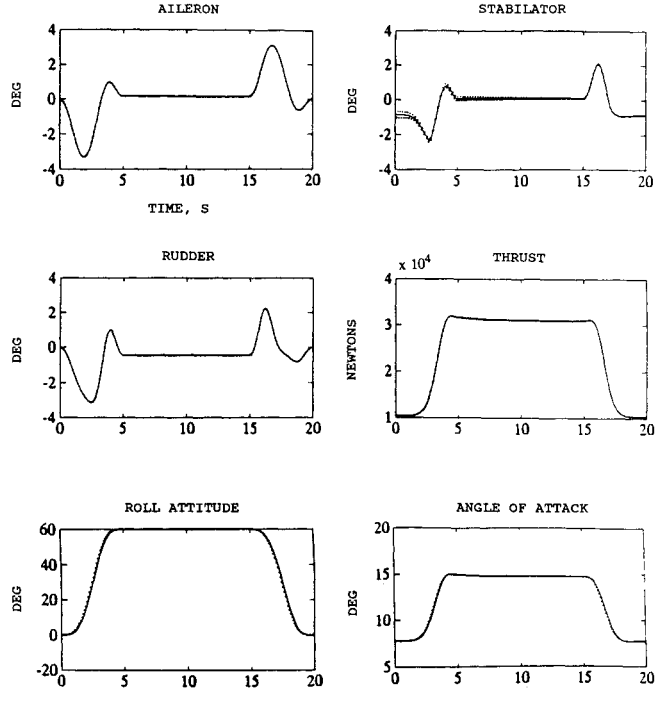

Fig. 2 Inverse and forward simulation results for coordinated turn.

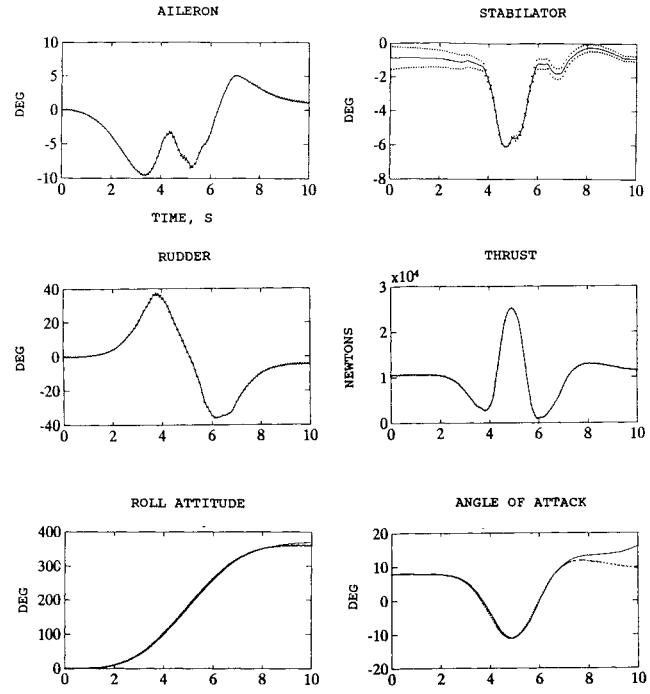

Fig. 3 Inverse and forward simulation results for 360-deg roll about x-body axis.

Here the AOA is unconstrained to allow the inverse solution to exist because the AOA has to change from positive to negative when the aircraft becomes inverted. Figure 3 shows the inverse simulation results for this maneuver. It was found that allowing $T = 0.05$ s produced better results than those for $T = 0.1$ s, and it is the smaller T value that was used to produce the results of Fig. 3. Note that this is a very slow roll, taking 10 s to complete.

Discussion

The overall results illustrate the capability of the proposed generalized inverse simulation technique to handle a very complicated nonlinear aircraft model in various maneuvers. The trimming procedure provides proper initial conditions for initiating the inverse simulations. Different combinations of the aircraft state variables, i.e., Euler angles, linear and angular velocity components, and aerodynamic angles can be used to describe the large-amplitude aircraft maneuvers. The flexibility of using all of the state variables to define maneuvers appears superior to other approaches such as the attitude projection method.[3]

All of the computations discussed herein, including implementation of the F-16 model, were completed on an IBM PC-AT desktop computer with an 80286 CPU and an 80237 coprocessor using LINPACK, FORTRAN, and MATLAB software. A typical inverse simulation of a maneuver required on the order of 35 min of computer time for $T = 0.1$s and $Dt = 0.01$ s.

The inverse simulations themselves were nominal in that the number of controls equaled the number of desired or constrained outputs. Thus the redundant problem was not encountered here, although it has been addressed in other applications, e.g., Ref. 5. However, it is simple to incorporate other inputs such as the speed brake in this problem.

It would be desirable to minimize the oscillatory control inputs that sometimes occur in the inverse simulation. Algorithm modifications that may alleviate this problem are 1) allowing the discretization interval T to be adaptive with respect to the value of the error function F_E in Eq. (4), and 2) improving the accuracy of the Jacobian matrix by perturbing the system with multiple inputs Δu, both positive and

negative in sign, and using the average values of the various resulting $\partial y_i/\partial u_j$ as elements of the Jacobian $\boldsymbol{J}$.

Conclusions

Based on the research described herein, the following conclusions can be drawn:

1) A generalized technique for inverse simulation formulated as an integration process appears to offer advantages over those that require time differentiation.

2) The technique can and has been applied to inverse problems in which the number of controls equals or exceeds the number of constrained outputs.

3) The technique can be used to find the trim flight condition of a complicated aircraft/system model, a necessary part of an inverse simulation.

4) The technique was successfully applied to large-amplitude maneuvers of a realistic aircraft model with significant aerodynamic nonlinearities.

References

[1]Jones, R. T., "A Simplified Application of the Method of Operators to the Calculation of Disturbed Motions of an Airplane," NACA TR 560, 1936.

[2]Kato, O., and Suguira, I., "An Interpretation of Airplane General Motion and Control as Inverse Problem," *Journal of Guidance, Control, and Dynamics,* Vol. 9, No. 2, 1986, pp. 198–204.

[3]Kato, O., "Attitude Projection Method for Analyzing Large-Amplitude Maneuvers," *Journal of Guidance, Control, and Dynamics,* Vol. 13, No. 1, 1990, pp. 22–29.

[4]Thomson, D. G., and Bradley, R., "Development and Verification of an Algorithm for Helicopter Inverse Simulation," *Vertica,* Vol. 14, No. 2, 1990, pp. 185–200.

[5]Hess, R. A., Gao, C., and Wang, S. H., "A Generalized Technique for Inverse Simulation Applied to Aircraft Flight Control," *Journal of Guidance, Control, and Dynamics,* Vol. 14, No. 5, 1991, pp. 920–926.

[6]Schreiner, J., Wood, N., Erickson, H., and Guyton, R., "Application of Tangential Sheet Blowing on the Forebody of an F-18 Like Configuration for High Angle of Attack Maneuverability," AIAA Paper 89-0097, 1989.

[7]Bugajski, D., Enns, D., and Elgersma, M., "A Dynamic Inversion Based Control Law with Application to the High Angle-of-Attack Research Vehicle," AIAA Paper 90-3407, 1990.

[8]Nguyen, L. T., Ogburn, M. E., Gilbert, W. P., Kibler, K. S., Brown, P. W., and Deal, P. L., "Simulator Study of Stall/Post-Stall Characteristics of a Fighter Airplane with Relaxed Longitudinal Static Stability," NASA TP 1538, Dec. 1979.

[9]Rao, C. R., and Mitra, S. K., *Generalized Inverse of Matrices and Its Application,* Wiley, New York, 1971.

[10]Goloub, G. J., and Van Loan, D. F., *Matrix Computations,* Johns Hopkins, Baltimore, MD, 1983.

[11]Dongarra, J. J., et al., *LINPACK: User's Guide,* Society for Industrial and Applied Mathematics, Philadelphia, PA, 1979.