Simulation Model-Building Procedure for Dynamic Systems Integration

P. Douglas Arbuckle* and Carey S. Buttrill*
NASA Langley Research Center, Hampton, Virginia
and
Thomas A. Zeiler†
Planning Research Corporation, Hampton, Virginia

A procedure for building simulation models that are useful in aircraft dynamic systems integration is described. The objective of the procedure is increased simulation model fidelity while reducing the time required to develop and modify these models. The equations of motion for an elastic aircraft and their impact on the procedure are discussed in broad terms. A software tool that automatically generates FORTRAN code to perform tabular data lookups, the language used to develop a simulation model, and the requirements for passing information into a simulation are described. A simulation-variable nomenclature is presented. The procedure has been applied to build an open-loop F/A-18 simulation model. This example model is used to illustrate model-reduction issues.

Nomenclature

$\mathbf{C}$ = second-order momentum coupling matrix, for n elastic modes, $(n \times 3)$; $= [\boldsymbol{h}_{j1}\, \eta^j \vdots \cdots \vdots\, \boldsymbol{h}_{jn}\, \eta^j]^T$
$C_{L_{\delta_f}}$ = lift coefficient due to flap deflection
F = vector of total applied force on the aircraft, (3×1)
g_H = gravity gradient with respect to altitude
g_o = gravitational acceleration at sea level
H = altitude above mean sea level, ft
$\boldsymbol{h}_{jk}$ = vector of residual mass coupling between angular and elastic momentum, (3×1); $\boldsymbol{h}_{jk} = -\boldsymbol{h}_{jk}$
i = the imaginary part of complex number
$\mathbf{J}$ = total inertia matrix of the aircraft in its deformed state, (3×3); $= \mathbf{J}_o + \Delta\mathbf{J}_j \eta^j + \frac{1}{2}\Delta^2\mathbf{J}_{jk}\eta^j\eta^k$
$\mathbf{J}_o$ = standard inertia matrix of the aircraft in its unloaded reference condition, (3×3); $= \begin{bmatrix} I_{xx} & 0 & -I_{xz} \\ 0 & I_{yy} & 0 \\ -I_{xz} & 0 & I_{zz} \end{bmatrix}$
$\Delta\mathbf{J}_j$ = first partial derivative of the aircraft inertia matrix $\mathbf{J}$ with respect to the jth mode, (3×3); $\Delta\mathbf{J}_j = \Delta\mathbf{J}_j^T = (\partial/\partial\eta_j)\mathbf{J}$
$\Delta^2\mathbf{J}_{jk}$ = second partial derivative of the aircraft inertia matrix $\mathbf{J}$ with respect to the jth and kth modes, (3×3); $\Delta^2\mathbf{J}_{jk} = \Delta^2\mathbf{J}_{jk}^T = \Delta^2\mathbf{J}_{kj} = (\partial^2/\partial\eta_j\partial\eta_k)\mathbf{J}$
L = vector of total applied moment on the aircraft, (3×1)
l_Z = vector that is the third column of the Earth-to-body-frame direction cosine matrix, (3×1); $= [-\sin\theta \;\sin\phi\cos\theta \;\cos\phi\cos\theta]^T$
M = Mach number
$\mathbf{M}_{aug}$ = "augmented mass" matrix, see Fig. 1 for definition
M_{jk} = mass coupling between the jth and kth modes
m = total aircraft mass
n = number of elastic modes
Q_{η_j} = generalized force on the jth elastic mode
s = Laplace variable
V = velocity vector of the body-frame origin with respect to the inertial frame, (3×1)
α = angle of attack, deg
β = angle of sideslip, deg
η_j = generalized coordinate corresponding to the jth elastic mode
η^j,η^k = elastic mode generalized coordinates in the context of an indicial summation over j(or k) from 1 to n
$\boldsymbol{\eta}$ = vector of elastic mode generalized coordinates, $(n \times 1)$
ΣF_i = summation of the aerodynamic, gravity, and inertial forces
on the aircraft, (3×1)
ΣL_i = summation of the applied gravity and inertial moments on the aircraft, (3×1)
ΣQ = summation of the generalized force contributions affecting the elastic degrees of freedom, $(n \times 1)$
ω_j = vibration frequency of the jth elastic mode
ω = frequency
$\boldsymbol{\omega}$ = rotational velocity vector of the body frame with respect to the inertial frame, (3×1)

Superscripts

as = antisymmetric
sy = symmetric

Operators

$\{\dot{\ }\}$ = time rate of change of { } with respect to the Earth frame
$\{\ddot{\ }\}$ = second derivative of { } with respect to time, relative to the Earth frame
$\{^{\circ}\}$ = time rate of change of { } with respect to the body frame
$\{\ \}^T$ = transpose of { }

Introduction

THE design of future aircraft will require multidisciplinary integrated design and analysis. Agility requirements for future fighters are such that unsteady aerodynamic effects (dynamic stall, etc.) may one day become more important than classical static performance criteria.[1] Experimental forward-swept-wing configurations have demonstrated significant coupling of rigid-body pitch rate and the wing first bending elastic mode,[2] and such configurations have been proposed for future fighter designs. Flight test programs have demon-

Received Sept. 8, 1987; revision received July 18, 1988.

*Research Engineer. Member AIAA.
†Project Structures Engineer, Aerospace Technologies Division. Member AIAA.

strated the feasibility of improving aircraft performance by using feedback control to provide static stability, maneuver load alleviation, and/or increased flutter mode damping.[3] Current research emphasis on agile fighters with capabilities such as "point and shoot" and post-stall maneuvering using all-axis thrust-vectored control[4] suggests that aircraft angular rates may one day be limited only by concern for the pilot's ability to function. These facts and trends all point to the conclusion that future aircraft designs, particularly fighters, will be "integrated" in some fashion.

By definition, today's aircraft designs are "integrated" in a process that might be termed "subsystem integration." In this process, the aircraft subsystems are designed independently and then integrated in a manner that minimizes interactions (an example is the use of structural notch filters in control systems). The trend in integration is moving toward "functional integration." In this case many functions are considered early in the design process. To keep computations practical, several systems would be functionally integrated rather than attempt to simultaneously optimize all of the design variables. In an envisioned "configuration integration" process, traditional design constraints would be relaxed in a manner to achieve large performance gains through synergistic technology integration.[4]

Aircraft designed using functional integration techniques are uniquely characterized by both the extent to which the aircraft disciplines or subsystems are combined and by the use of embedded digital control systems to accomplish the integration. Often in these designs, system interactions that in the past were regarded as undesirable are permitted in a controlled, beneficial manner, making loss of the control system unacceptable and usually unsafe. This flight safety dependence on interacting systems is the primary distinction of functionally integrated designs from most current aircraft.[5]

Applications of aircraft dynamic systems analyses fall into three categories that are related to the three approaches to integration previously described. These categories are 1) applications to solve problems arising in the aircraft flight test development process, i.e., a "fix"; 2) applications to a "frozen" airframe configuration (before final design) to achieve some benefits; and 3) applications early in the design process that have an impact on the configuration in a manner to greatly enhance performance.[4]

The work reported in this paper was motivated by the need to develop and document a methodology for functionally integrating dynamic systems in aircraft design. One ultimate objective of this methodology is an improvement in the effectness of simulation in the design process, accomplished by increasing model fidelity while reducing the time required to develop and modify simulation models. Achieving this objective will help flying-qualities engineers and control-system designers to play a first-order role in the aircraft design process.

This paper describes the development of a procedure to build simulation models that satisfy the aforementioned objective. Design of this procedure has been driven by at least four factors:

1) The equations of motion for an elastic aircraft.

2) The use of the Advanced Continuous Simulation Language (ACSL)[6] software to develop aircraft simulation models.

3) The desire to have a paperless interface with a real-time simulation facility. This includes the ultimate capability to deliver validated aircraft control laws as FORTRAN subroutines to be inserted into the real-time simulation code.

4) The current technology level of analysis and design tools, both in terms of what quantities can be computed for use in the simulation equations of motion and in terms of what quantities are required as inputs to control-law algorithms.

Two of several philosophies developed by Radovcich[7] have also guided the procedure development. First, the computer methodology developed must be flexible and highly modular to deter obsolescence as new engineering analysis tools become available. Second, each discipline will, in general, define its own modeling requirements—modeling tasks that involve multiple disciplines will be integrated by a designated discipline.

The procedure has been exercised to build several simulation models. One of these is an F/A-18 simulation that is an open-loop model of the continuous airplane plant. The flight control system of the F/A-18 has been used to solve various problems that surfaced during flight test development.[8] This fact makes the inclusion of the control-system model into the simulation a priority. However, since this F/A-18 simulation is to be used as an initial testbed for developing and refining dynamic systems integration techniques, it is necessary that an open-loop model be developed for analysis by control-law algorithm researchers. Therefore, the model of the flight control system will be included in the simulation at a later date.

The paper is organized as follows. First, the equations of motion are presented. Second, the simulation language is described. Third, a software tool for efficiently incorporating tabular data models into the simulation code is explained. Fourth, the data requirements and information flow of the simulation model-building procedure is discussed. Fifth, a method is described for assigning variable names in aircraft simulation models based on nomenclature developed by the authors. Sixth, model size and model-reduction issues are discussed. This section is followed by concluding remarks.

Elastic Aircraft Equations of Motion

Before a vehicle simulation can be developed, the equations of motion of the vehicle must be known. Further, the equations of motion must include all terms necessary to accurately simulate the desired dynamics.

The elastic aircraft equations of motion are examined using Lagrangian mechanics. The aircraft is idealized as a collection of lumped masses and lumped inertias being displaced about a noninertial mean reference body-axis frame. The elastic aircraft equations of motion as developed are[9]

Translational momentum:

$$m\mathring{V} = F - m(\omega \times V) + m(g_o + g_H H)l_Z \tag{1a}$$

Angular momentum:

$$\mathbf{J}\mathring{\omega} + h_{jk}\,\eta^j\ddot{\eta}^k = L - \omega \times \mathbf{J}\omega - \mathring{\mathbf{J}}\omega$$
$$- h_{jk}\,\dot{\eta}^j\dot{\eta}^k - \omega \times h_{jk}\,\eta^j\,\dot{\eta}^k \tag{1b}$$

jth Elastic mode:

$$M_{jk}\ddot{\eta}^k - \mathring{\omega}^T h_{jk}\,\eta^k = Q_{\eta_j} - M_{jj}\omega_j^2\eta_j + 2\omega^T h_{jk}\dot{\eta}^k$$
$$+ \tfrac{1}{2}\omega^T\{\Delta\mathbf{J}_j + \Delta^2\mathbf{J}_{jk}\eta^k\}\omega \tag{1c}$$

The notation η^k indicates a summation over k from 1 to n. Also, the notation $a^T b$ is equivalent to $a \cdot b$. Equations (1) are not new. The only difference from what is typically found in the literature for aircraft is that the nonlinear inertial coupling terms are retained. These terms involve products of rigid-body angular rates, structural deformations, and structural deformation rates. The terms are neglected in typical formulations by assuming that the body-axis rotational rates are small.

The nonlinear inertial coupling terms arise from two sources. The first source is the change in the total aircraft inertia matrix due to elastic deformation. If deformation is described as a linear combination of elastic modes with their time-dependent participation coefficients, the inertia matrix is subject to first and second partial derivatives with respect to the modes. The terms $\Delta\mathbf{J}_j$ and $\Delta^2\mathbf{J}_{jk}$ are the result.

The second source of nonlinear inertial coupling arises from the fact that given a modal description of deformation, the mode shapes can only be forced to satisfy the first-order

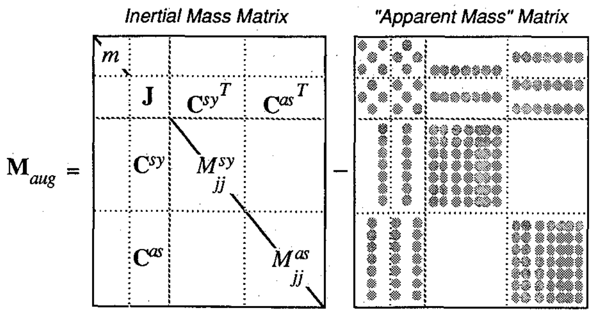

Fig. 1 Structure of "augmented mass matrix" $\mathbf{M}_{aug}$.

(practical) mean-axis conditions.[10] The term h_{jk} represents a second-order coupling between angular and modal momentum and is calculated by integrating vector cross products of the mode shapes over the aircraft.

The nonlinear inertial coupling terms make virtually no difference in linear stability derivatives calculated at straight and level flight conditions. It is conjectured that their impact on elastic aircraft response at high angular rates for a configuration with stores could be significant.[9]

The implemented form of the equations of motion in the simulation is for an aircraft that is symmetric about the body-frame x-z plane. These equations take computational advantage of the aircraft symmetry by partitioning appropriate vectors and matrices into symmetric and antisymmetric components, and can be written as

$$\mathbf{M}_{aug}\begin{bmatrix}\overset{\circ}{V}\\ \overset{\circ}{\omega}\\ \ddot{\eta}^{sy}\\ \ddot{\eta}^{as}\end{bmatrix}=\begin{bmatrix}\Sigma F_i\\ \Sigma L_i\\ \Sigma Q^{sy}\\ \Sigma Q^{as}\end{bmatrix} \tag{2}$$

This partitioning permits the known zero-valued elements in the equations to be arranged in predetermined submatrices of the "augmented mass matrix" $\mathbf{M}_{aug}$ (defined in the next section). Figure 1 shows how $\mathbf{M}_{aug}$ is partitioned. The simulation code has been written by the authors to take advantage of this known model structure. ΣF_i represents a summation of the various forces resulting from aerodynamic loads, gravity, and inertial effects. Similarly, ΣL_i represents the various moments. ΣQ^{sy} and ΣQ^{as} represent the various generalized forces affecting the symmetric and antisymmetric elastic degrees of freedom, respectively.

Simulation Language

As mentioned in the Introduction, the Advanced Continuous Simulation Language (ACSL)[6] has been used in the simulation model-building procedure. Major factors in this decision were the commercial availability of ACSL and the authors' possession of the software. Other reasons supporting the selection of ACSL include the following:

1) It can automatically generate a linear system quadruple from the full nonlinear set of differential equations.

2) It allows the creation of user-written macros and can call user-written FORTRAN subroutines.

3) It provides a rigorous blending of discrete and continuous dynamic systems.

ACSL does impose some limitations and programming rigor on its users. For example, due to the ACSL requirement that differential equations be in explicit form, the elastic aircraft equations of motion were written such that all acceleration-dependent force terms appear on the left side of Eq. (2). These force terms, which occur due to the unsteady aerodynamics, were brought into the mass matrix as "apparent mass" terms. The resulting time-dependent mass matrix was named the "augmented mass matrix" (Fig. 1).

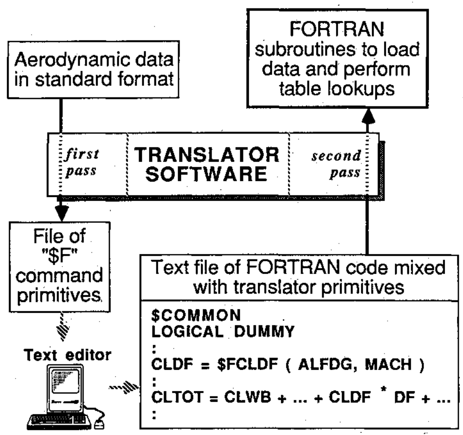

Fig. 2 Use of translator software to generate FORTRAN code for simulation model.

The time dependence of the augmented mass matrix, due to the apparent mass terms and the submatrices $\mathbf{J}$, $\mathbf{C}^{as}$, and $\mathbf{C}^{sy}$, causes a potential matrix inversion each time the equations of motion are evaluated. The second-order Runge-Kutta integration algorithm used in the simulation requires two evaluations of the equations of motion per time step. Therefore, a possible requirement was presented of inverting a matrix of size $(6+n)\times(6+n)$ two times per integration time step. This matrix inversion would have imposed a significant computational burden on the simulation, so other options were explored.

It was observed that the diagonal elements of the augmented mass matrix were, for aircraft, always large relative to the other matrix elements. With this insight, a zero-pivoting Gaussian elimination algorithm was used to make the augmented mass matrix lower triangular and therefore solvable without matrix inversion. This approach ensured a rigorous computation of the aircraft's motion (and its linear system quadruple) yet introduced no measurable computational penalty in the ACSL environment.

The philosophy adopted for using ACSL in the simulation model-building procedure is that all physically continuous dynamics are modeled in ACSL. This includes actuator and sensor dynamics, engine dynamics, and the equations of motion. Physically discrete dynamics, such as a digital flight control system, are not coded in ACSL. These dynamics are coded in FORTRAN and accessed by the simulation through subroutine calls that occur at discrete time increments.

Software Tool for Tabular Data Models

Typical applications for table lookup functions in aircraft simulation include 1) calculation of total aerodynamic coefficients; 2) use of engine performance tables; and 3) implementation of control laws with scheduled parameters.

Most real-time simulation facilities use a specific file format for storing tabular data values to be used by function lookup routines. This file format is defined as the input to a software tool that aids in building FORTRAN code to perform table lookups, and was adopted for use in the simulation model-building procedure as a step toward achieving the desired paperless interface with a real-time simulation facility. The authors have added to the functionality of this FORTRAN code-building software tool.[11] For discussion purposes, the resulting software is called "translator software." A schematic representation of the use of the translator software in the simulation model-building procedure is shown in Fig. 2.

The translator software first reads a file of tabular data values to determine the functional dependencies of the data and the ranges of the independent variable values. The functional dependencies are automatically written in one-line FORTRAN-like statements called "primitives." As an example, the tabular data for the F/A-18 aerodynamics model contains a table lookup for lift coefficient due to flap deflection as a function of angle of attack and Mach number, or

$$C_{L_{\delta_f}} = f(\alpha, M) \tag{3}$$

The primitive for this lookup is written as

$$\text{CLDF} = \$\text{FCLDF(ALFDG,MACH)} \tag{4}$$

The "$F" string is the operator that identifies the primitives for the translator software.

The engineer uses a text editor to manipulate and combine primitives, as well as to input standard FORTRAN statements that compute quantities for the simulation and provide internal documentation. The translator software then uses the file augmented by the engineer to automatically generate the necessary FORTRAN subroutines to load the tabular data and compute the lookups. For a typical steady aerodynamics model, up to 80% of the FORTRAN code lines may be generated without human intervention. At least 50% of the FORTRAN code lines are generated for typical engine models.

Since the translator-software file format has been adopted as a "standard" for the simulation model-building procedure, all tabular data will be converted via automated software tools to the appropriate format[11] as a condition for use in the procedure. One such tool has been developed to convert output from the U.S. Air Force digital Data Compendium (DATCOM) program. Similar tools for converting wind-tunnel data, output from computational fluid dynamics codes, or an engine deck could be developed easily.

Data Requirements and Information Flow

The input quantities required by the simulation are defined by the equations-of-motion formulation and include normally computed quantities – geometry and mass data, nonlinear steady aerodynamic data, engine data – as well as quantities from the structural model (mode shapes, frequencies, etc.) and from the unsteady aerodynamics model (generalized forces).

Nonlinear Steady Aerodynamics Model

The steady aerodynamics and its documentation were obtained from an operational real-time F/A-18 simulation. The steady aerodynamics model consists of FORTRAN subroutines that access a tabular data base of aerodynamic coefficients. These subroutines could not be directly used in the simulation model-building procedure since they access machine-dependent assembly language routines.

The F/A-18 steady aerodynamics data base is defined within the following range of paramaters:

$$-10 \text{ deg} \le \alpha \le 90 \text{ deg}$$
$$-20 \text{ deg} \le \beta \le 20 \text{ deg}$$
$$0.2 \le M \le 2.0$$
$$0 \text{ ft} \le H \le 60{,}000 \text{ ft}$$

The data base included various terms (including "flex-rigid ratios") to model the quasistatic-elastic deformation of the aircraft from its "jig shape" at various flight conditions. The simulation model-building procedure is designed to directly include the structural modes and their impact on the aerodynamics; therefore, the procedure uses the "jig shape" aerodynamics for the F/A-18. The jig shape aerodynamics were recovered by setting appropriate terms to values determined for the lowest dynamic pressure in the data base (for example, flex-rigid ratios were set to one).

The data base was converted to the translator-software file format and processed as described previously (see "Software Tool for Tabular Data Models"). The steady-flow aerodynamics model documentation was used extensively during the editing process of combining and manipulating primitives. The translator software was then used to generate the FORTRAN subroutines that load the tabular aerodynamics data into program memory and perform linear interpolation lookups as required by the simulation.

Engine Model

The F404 engine model and its documentation were also obtained from the existing real-time F/A-18 simulation. The engine model consists of FORTRAN subroutines that access a tabular data base of engine performance and dynamics parameters. These subroutines also access machine-dependent software and thus could not be directly used in the simulation model-building procedure.

Both the throttle-commanded steady-state thrust level and the dynamic response characteristics of the engine model are based on the engine airflow rate as determined from a table lookup. Afterburner dynamics are switched in at a threshold based on the engine airflow and commanded thrust. A model of a hypothetical thrust-vectoring system is also included and may be optionally activated.

The data base was processed in a manner largely analogous to the steady aerodynamics data base. The engine dynamics, being physically continuous, were modeled in ACSL, in keeping with the philosophy described previously (see "Simulation Language").

Structural Model

A NASTRAN beam half-model (Fig. 3) of the F/A-18 was obtained from the airframe manufacturer and translated into the Engineering Analysis Language (EAL).[12] This translation was performed since the authors prefer to use the data management and processing capabilities of EAL. The structural model was analyzed to determine the free-free vibration modes with both symmetric and antisymmetric boundary conditions and to determine internal modal load coefficients.

With external FORTRAN programs, the mode shapes were analyzed to determine their satisfaction of the "practical" mean-axis conditions.[10] The mean-axis calculation was accomplished using the lumped masses and their modal displacements.[9] Although the free-vibration eigenproblem was solved for free-free boundary conditions, the modes did not satisfy the practical mean-axis conditions to machine accuracy. Therefore, small translational and rotational corrections were computed and applied to the mode shapes. These corrections preserved the mode shapes, spatially reorienting them in order to satisfy the pratical mean-axis conditions. Therefore the load coefficients computed within EAL remained valid. Using these corrected mode shapes, the nonlinear inertial coupling quantities and generalized mass terms required by the equations of motion were computed and arranged on a data file[11] to be loaded into the simulation.

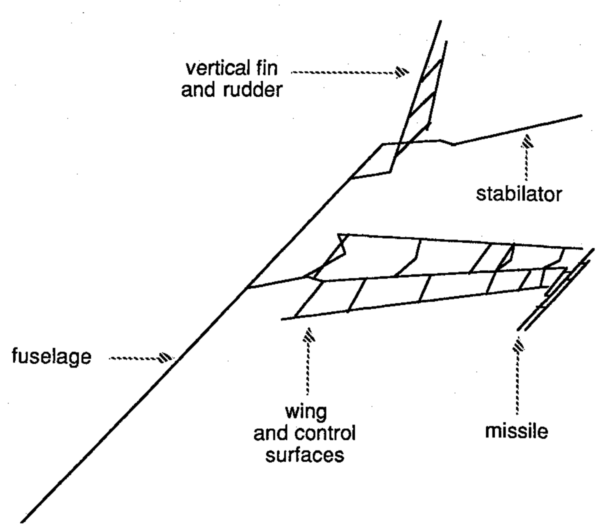

Fig. 3 Finite-element beam half-model of the F/A-18.

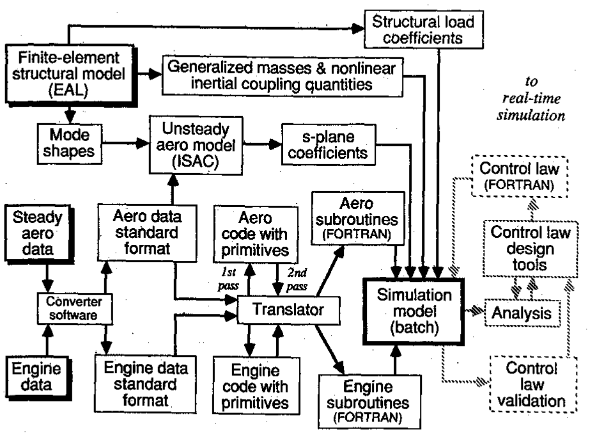

Fig. 4 Information flowchart for simulation model-building procedure.

Unsteady Aerodynamics Model

The mode shapes previously described were used in the Interaction of Structures, Aerodynamics, and Controls (ISAC)[13] programs to compute generalized unsteady aerodynamic loads using doublet-lattice unsteady lifting surface theory for a range of reduced frequencies and Mach numbers. For each Mach number, a rational function approximation (in the Laplace s-plane) for the transfer function of the unsteady aerodynamic loads was determined by a least-squares fit to a table of oscillatory loads ($s = i\omega$) at various reduced frequencies and for a selected set of aerodynamic lags.[14] The lag time constants can also be optimized, if desired, to minimize the least-squares error.[15] The s-plane fits are placed on a data file[11] to be loaded into the simulation.

The doublet-lattice theory used by the ISAC programs yields satisfactory results when the aircraft is modeled well by lifting surfaces (wings and tails). When motion of an elastic mode is predominantly normal to the lifting surfaces, computation of the modal elastic loads due to these motions (elastic-elastic effects) are generally good for the subsonic regime. On the other hand, rigid-body modes have significant motion due to nonlifting surfaces, such as the fuselage. Therefore, the rigid-body loads due to rigid-body motions (rigid-rigid effects) are usually not predicted well by doublet-lattice theory. The rigid-body loads due to motions of the elastic modes are called the rigid-elastic effects on the generalized aerodynamic forces. The accuracy of these effects as computed using the ISAC programs lies between the elastic-elastic and rigid-rigid effects.

In general, the rigid-elastic effects computed using the ISAC programs do not match, at zero frequency, the loads predicted by accurate computational steady aerodynamics codes. The use of a computational steady aerodynamics code for calculation of the quasistatic rigid-elastic effects is an iterative and resource-intensive process involving calculation of the airframe shape under load and the aerodynamic loads for the deformed airframe. The quasistatic rigid-elastic data computed in this manner could then be input to the ISAC programs to replace the doublet-lattice zero-frequency values. With more accurate zero-frequency information, the least-squares fit in the s-plane could then proceed. A related approach is the calculation of correction factors to the doublet-lattice data to improve zero-frequency matching.[16]

For the F/A-18 simulation, an experimental tabular data base was available for rigid-body loads due to rigid-body motions and control surface deflections. Therefore, this tabular data base was used by the simulation instead of the rigid-rigid effects predicted by the ISAC programs. The rigid-elastic effects as computed using the ISAC programs were used directly in the simulation. The "error" introduced by this implementation was not quantified since it was regarded as temporary.

Structural Loads

Selected load coefficients were placed in a data file[11] to be loaded into the simulation so that time histories of internal structural loads could be computed and displayed for a set of predetermined structural stations. Alternatively, a file of modal coordinate (η_j) time histories from the simulation can be made available to the structural analyst for more detailed and extensive loads analyses.

Information Flow

Figure 4 shows a flowchart of the basic components of the simulation model-building procedure as solid lines; the dotted lines indicate the links between the procedure and other elements of the dynamic systems integration methodology. The heavy dashed lines indicate successfully exercised pathways to existing tools. The light dashed lines indicate pathways that are planned but have not yet been developed.

The starting points for building a simulation model using the procedure are the EAL structural model, steady aerodynamic data, and engine data. The aerodynamic and engine data are converted to the translator-software file format[11] and processed to generate the steady aerodynamics model and engine model as previously described. Simultaneously, the EAL model is exercised to generate mode shapes for input to the ISAC programs. The EAL model also generates information for the load-coefficients file as well as the generalized mass terms and nonlinear inertial-coupling quantities file. The steady aerodynamics data is supplied to the ISAC programs and can be used to compute correction factors for the doublet-lattice jig-shape aerodynamics at zero frequency. The ISAC programs subsequently compute the s-plane coefficients for the rational function approximation of the unsteady aerodynamics. Ideally, the various data files and FORTRAN subroutines to be used by the ACSL simulation would become available at approximately the same time (generally this is not possible). After verifying and validating the complete simulation model, it can be exercised to generate results as required.

Variable Nomenclature

Development of the simulation model-building procedure, like the development of a real-time simulation, has involved the efforts of several engineers and programmers and has included receipt of data, documentation, and several FORTRAN subroutines from outside sources. This effort has given the authors a keen sense of the desirability of a standard variable nomenclature for coding simulation models. Without a standard variable nomenclature, it was generally impossible to understand the computer code without having the documentation in hand.

The authors found this situation to be undesirable and developed a method, used by all personnel working on the project, for naming code variables. This should not be confused with issuing a variable dictionary for project personnel to use. The philosophy behind the variable nomenclature method was that any given variable name could be constructed from a set of predefined mnemonics[17] (base names; prefix and suffix modifiers), given an appropriate set of contruction rules.[11] The successful test of this nomenclature method is defined when two individuals, given the same symbol, construct the same FORTRAN variable name without consultation.

ACSL provides the capability for creating and maintaining a file-based simulation variable dictionary.[6] The implementation of the dictionary capability quickly identifies variables that have undefined values and definitions. The ACSL dictionary was installed as an option in the F/A-18 simulation and was used in the development of the variable nomenclature method. The dictionary file developed for use with the F/A-18 simulation is used as the permanent documentation recorded of all assigned variable names in the main simulation program.

Model Size and Model Reduction

One of the philosophies inherent in the development of dynamic systems integration methodology and the simulation model-building procedure is that the resulting large, high-fidelity simulation will be run in a batch mode and will be regarded as a "truth" model. Further, the real-time simulation model as well as models for dynamics analysis and control law design, will be reduced models of the "truth" simulation.

The technical problem is how to reduce the truth simulation model to a size suitable for real-time simulation, dynamics analysis, or control-law design while maintaining the desired accuracy. It is probable that new techniques will be required to accomplish this reduction in a reasonably expedient way. Some insight into the amount of model reduction required is provided by considering the F/A-18 simulation developed with the model-building procedure.

The number of states in an open-loop F/A-18 simulation depends strongly on the number of elastic modes required to describe the structural dynamics and the number of aerodynamic lags required to describe the unsteady aerodynamics. For the F/A-18 simulation implementation, the states are broken down as follows:

6	Rigid-body
5	Altitude/quaternions
14	Engine (core plus afterburner)
3	Gusts
19	Control surfaces (first-order actuators, three surfaces; second-order actuators, eight surfaces)
40	Elastic modes (20 modes)
104	Aerodynamic lags (four-lag formulation)
191	TOTAL states

Reducing the number of aerodynamic lags by one in the unsteady aerodynamic formulation reduces the number of total model states by six plus the number of elastic modes (26 for the F/A-18 simulation). Unfortunately, as the number of aerodynamic lags are reduced, obtaining an accurate rational function approximation of the unsteady aerodynamics becomes more difficult and time consuming. Research has been conducted to develop techniques to reduce the number of aerodynamic lags while preserving a given level of approximation error.[15] These techniques will be applied in the future as appropriate to compute unsteady aerodynamics in the simulation model.

The aerodynamic lag formulation poses a particular problem for implementing a real-time simulation version of the truth simulation model. The dynamics of the lag states are determined by the range of reduced frequencies necessary to model the aircraft dynamics under investigation. The original implementation of the unsteady aerodynamic data base for the open-loop F/A-18 simulation is such that at $M = 0.9$, sea level, the eigenvalue of the fastest aerodynamic lag state is a real pole at approximately -349 rad/s. To insure that this state does not become numerically unstable, the simulation uses a time step of 2 ms, which is not achievable in most current real-time simulation facilities. Real-time simulation studies will typically consider a lower range of model frequencies and a reduced number of elastic degrees of freedom than used in the truth simulation. For such studies, it may be possible to choose the aerodynamic lag pole magnitudes such that the real-time simulation time step can be increased to about 10 ms.

Up to this point, the authors have virtually ignored the CPU and memory requirements of simulations built using the procedure. This was done primarily to ensure that the methodology developed would be relatively independent of the computer hardware and software environment in which the procedure would be exercised. Although relatively machine-independent, the open-loop F/A-18 simulation is usually exercised on a MicroVAX II‡ minicomputer. In this environment, the simulation operates at approximately 300 times slower than real time and uses approximately 250,000 words of memory. The execution speed of the simulation depends strongly on the selected time step; the above execution speed is for a time step of 2 ms.

There are several modifications that could be made to the open-loop F/A-18 simulation to decrease its CPU and memory requirements. Most modifications to reduce the memory requirements involve more input/output operations as the program executes. The extra programming effort and decreased CPU efficiency are deemed not worth any memory savings, especially on the MicroVAX. However, scheduling the simulation subroutine calls to the engine and steady aerodynamics models to occur at time steps that are more appropriate to their dynamics (i.e., every 10–50 ms vs every 2 ms) would result in significant CPU savings. The rational function approximation coefficients for the unsteady aerodynamics model are linearly interpolated as a function of Mach number. Scheduling this interpolation with the engine and steady aerodynamics would also result in additional efficiency.

‡MicroVAX is a trademark of Digital Equipment Corporation.

Technology Limitations

Current tools that compute unsteady aerodynamics for aircraft simulation are generally only valid for low values of α that are in the linear region of the lift-curve slope. Therefore, accurate unsteady aerodynamics for a high-performance aircraft cannot be computed with confidence for all tactically significant regions of the flight envelope. This fact inhibits design opportunities to exploit unsteady aerodynamics effects in important situations.

Thrust-vectoring models made available to date do not account for the interactions between the vectored engine exhaust and the local flow over the tail. These interactions could be significant for aircraft like the F/A-18, where small changes in the downwash field at the horizontal stabilizer impact the entire configuration's aerodynamics. Developing tools that can accurately compute these interactions to yield useful parameters for aircraft simulation remains a challenge.

Concluding Remarks

High-fidelity simulation models can be constructed in a manner that minimizes the time required to develop and modify the simulation. This paper describes a procedure for systematically assembling information from the structures and aerodynamics disciplines to quickly build a suitable high-fidelity simulation for use in dynamics analysis and integrated control-system design. The equations of motion, the features of the simulation programming language, and the ability to compute appropriate quantities, dictate the flow and format of information in the model-building procedure. The tabular data typically used in aerodynamic and engine models can be efficiently incorporated into the simulation by using an appropriate software tool. A standard method for naming simulation code variables is preferable to a predefined dictionary of variable names.

The simulation model-building procedure has been used to build several simulations; construction of an open-loop simulation model of the F/A-18 aircraft was used to demonstrate the procedure. The open-loop F/A-18 simulation demonstrates how strongly the model size (number of states) depends on the number of elastic modes and the number of aerodynamic lags. The high-fidelity simulation models that result from the procedure can be executed in a batch mode. However, given current computer technology, model-reduction techniques must be applied in order to generate simulation models that can be operated in real time.

Development of this simulation model-building procedure offers several new opportunities. First, batch simulation models can be built in parallel with real-time simulation models, yielding the obvious advantage of maintaining an "independent" check simulation. Second, control-law algorithm researchers can apply their algorithms to "real" nonlinear dynamics problems in a timely manner, helping to bridge the gap between control theoreticians and control-law designers. Third, metrics can be developed that could be used to evaluate a control-system design by properly exercising a batch simulation, analytically determining the regions in a simulation test matrix that require a real pilot-in-the-loop for proper analysis.[11]

References

[1]Mabey, D. G. and Chambers, J. R., "Unsteady Aerodynamics–Fundamentals and Applications to Aircraft Dynamics," AGARD-AR-222, May 1985.

[2]Miller, G., Wykes, J., and Brosnan, M., "Rigid Body-Structural Mode Coupling on a Forward Swept Wing Aircraft," AIAA Paper 82-0683, May 1982.

[3]Schwanz, R. C., "Consistency in Aircraft Structural and Flight Control Analysis," AGARD-CP-228, Paper 5, April 1977.

[4]Hood, R. V., Dollyhigh, S. M., and Newsom, J. R., "Impact of Flight Systems Integration on Future Aircraft Design," AIAA Paper 84-2459, Oct. 1984.

[5]Mackall, D. A., "Qualification Needs for Advanced Integrated Aircraft," AIAA Paper 85-1865, Aug. 1985.

[6]*Advanced Continuous Simulation Language (ACSL) Reference Manual*, 4th ed., Mitchell and Gauthier Associates, Concord, MA, 1986.

[7]Radovcich, N. A., "Preliminary Aeroelastic Design of Structures (PADS) Methods Development and Application," 56th AGARD Structures and Materials Panel Meeting, AGARD-CP-354, April 1983.

[8]Harschburger, H. E. and Moomaw, R. F., "Experience with the F/A-18 Digital Flight Control System," IEEE/AIAA 5th Digital Avionics Systems Conference, Seattle, WA, Oct.–Nov. 1983.

[9]Buttrill, C. B., Zeiler, T. A., and Arbuckle, P. D., "Nonlinear Simulation of a Flexible Aircraft in Maneuvering Flight," AIAA Paper 87-2501, Aug. 1987.

[10]Waszak, M. R. and Schmidt, D. K., "On the Flight Dynamics of Aeroelastic Vehicles," AIAA Paper 86-2077, Aug. 1986.

[11]Arbuckle, P. D., Buttrill, C. S., and Zeiler, T. A., "A New Simulation Model Building Process for Use in Dynamic Systems Integration Research," AIAA Paper 87-2498, Aug. 1987.

[12]Whetstone, W. D., *EISI-EAL Engineering Analysis Language Reference Manual*, Engineering Information Systems, Inc., San Jose, CA, July 1983.

[13]Peele, E. L. and Adams, W. M., Jr., "A Digital Program for Calculating the Interaction Between Flexible Structures, Unsteady Aerodynamics, and Active Controls," NASA TM-80040, 1979.

[14]Adams, W. M., Jr., Tiffany, S. H., Newsom, J. R., and Peele, E. L., "STABCAR–A Program for Finding Characteristic Roots of Systems Having Transcendental Stability Matrices," NASA TP-2165, June 1984.

[15]Tiffany, S. H. and Adams, W. M., Jr., "Nonlinear Programming Extensions to Rational Function Approximations of Unsteady Aerodynamics," AIAA Paper 87-0854, April 1987.

[16]Wieseman, C. D., "Methodology for Matching Experimental and Analytical Aerodynamic Data," AIAA Paper 88-2392, April 1988.

[17]Mitchell, E. E. L., "Techniques for the Specification and Operation of a Standard Missile Simulation," Raytheon Corp., Bedford, MA, Rept. BA-7786, Dec. 1973.