Development of Reduced-Order Models for Aeroelastic Analysis and Flutter Prediction Using the CFL3Dv6.0 Code

Walter A. Silva *
NASA Langley Research Center
Hampton, Virginia 23681-0001

Robert E. Bartels †
NASA Langley Research Center
Hampton, Virginia 23681-0001

A reduced-order model (ROM) is developed for aeroelastic analysis using the CFL3D version 6.0 computational fluid dynamics (CFD) code, recently developed at the NASA Langley Research Center. This latest version of the flow solver includes a deforming mesh capability, a modal structural definition for nonlinear aeroelastic analyses, and a parallelization capability that provides a significant increase in computational efficiency. Flutter results for the AGARD 445.6 Wing computed using CFL3D v6.0 are presented, including discussion of associated computational costs. Modal impulse responses of the unsteady aerodynamic system are then computed using the CFL3Dv6 code and transformed into state-space form. Important numerical issues associated with the computation of the impulse responses are presented. The unsteady aerodynamic state-space ROM is then combined with a state-space model of the structure to create an aeroelastic simulation using the MATLAB/SIMULINK environment. The MATLAB/SIMULINK ROM is used to rapidly compute aeroelastic transients including flutter. The ROM shows excellent agreement with the aeroelastic analyses computed using the CFL3Dv6.0 code directly.

Introduction

EARLY mathematical models of unsteady aerodynamic response capitalized on the efficiency and power of superposition of scaled and time- shifted fundamental responses, also known as convolution. Classical models of two-dimensional airfoils in incompressible flow[1] include Wagner's function[2](response to a unit step variation in angle of attack), Kussner's function[3](response to a sharp-edged gust), Theodorsen's function[4](frequency response to sinusoidal pitching motion), and Sear's function (frequency response to a sinusoidal gust). As geometric complexity increased from airfoils to wings to complete configurations, the analytical derivation of these types of response functions became impractical and the numerical computation of linear unsteady aerodynamic responses, in the frequency domain, became the method of choice.[5]

When geometry- and flow-dependent nonlinear aerodynamic effects became significant, appropriate nonlinear aerodynamic equations were solved using time-integration techniques. Coupling the nonlinear aerodynamic equations with a linear structural model provides a direct simulation of aeroelastic phenomena. This direct simulation approach for solving nonlinear aeroelastic problems has yielded a very powerful simulation capability with two primary challenges. The first challenge is the associated computational cost of this simulation, which increases with the fidelity of the nonlinear aerodynamic equations to be solved. Computational cost may be reduced via the implementation of parallel processing techniques, advanced algorithms, and improved computer hardware processing speeds. The second, more serious, challenge is that the information generated by these simulations cannot be used effectively within a preliminary design environment. Any attempt to incorporate the output of these aeroelastic simulations within a design environment inevitably becomes design by trial-and-error, a completely impractical approach. As a result, the integration of traditional, computational aeroelastic simulations into preliminary design activities involving disciplines such as aeroelasticity, aeroservoelasticity

*Senior Research Scientist, Aeroelasticity Branch, NASA Langley Research Center, Hampton, Virginia;

†Aerospace Engineer, Aeroelasticity Branch, NASA Langley Research Center,Hampton, Virginia;

(ASE), and optimization is, at present, a costly and impractical venture.

The goal behind the development of reduced-order models (ROMs) is aimed precisely at addressing these two challenges. Development of a ROM entails the development of a simplified mathematical model that captures the dominant dynamics of the original system. This alternative mathematical representation of the original system is, by design, in a mathematical form suitable for use in a multidisciplinary, preliminary design environment. As a result, interconnection of the ROM with other disciplines is possible, thereby addressing the second challenge. The simplicity of the ROM yields significant improvements in computational efficiency as compared to the original system, thereby addressing the first challenge.

At present, the development of CFD-based ROMs is an area of active research at several industry, government, and academic institutions.[6] Development of ROMs based on the Volterra theory is one of several ROM methods currently under development.[7–11] Reduced-order models based on the Volterra theory have been applied successfully to Euler and Navier-Stokes models of nonlinear unsteady aerodynamic and aeroelastic systems. Volterra-based ROMs are based on the creation of linearized and nonlinear unsteady aerodynamic impulse responses that are then used in a convolution scheme to provide the linearized and nonlinear responses of the system to arbitrary inputs. In this setting, the linearized and nonlinear impulse responses are the ROMs of the particular nonlinear system under investigation. Upon transformation of the linearized and nonlinear impulse responses into state-space form, the state-space models generated can also be considered ROMs.

Another ROM method, different from the Volterra-based ROM approach, is the Proper Orthogonal Decomposition (POD) technique. The POD is a method that is used extensively at several research organizations for the development of reduced-order models. A thorough review of POD research activities can be found in Beran and Silva.[6] In addition, a review of the issues involved in the development of reduced-order models for fluid-structure interaction problems is provided by Dowell and Hall.[12] A topic of recent interest is the potential development of hybrid POD/Volterra methods. These hybrid techniques would combine the spatial resolution possible with POD methods with the low dimensionality and computational efficiency of Volterra methods.

The linearization of a nonlinear aeroelastic model is an important first step towards understanding the nature and magnitude of nonlinear aeroelastic phenomena. The response of a linearized system about a nonlinear steady-state condition can be obtained via several methods. Some of these methods include the order reduction of state-space models using various techniques.[13,14] One method for building a linearized, low-order, frequency-domain model from CFD analysis is to apply the exponential (Gaussian) pulse input.[15] This method is used to excite an aeroelastic system, one mode at a time, using a smoothly-varying, small-amplitude Gaussian pulse. The time-domain aeroelastic responses due to the exponential pulse input are transformed into frequency-domain generalized aerodynamic forces (GAFs). These linearized GAFs can then be used in standard linear aeroelastic analyses.[16] Raveh et al[17] applied this method but replaced the exponential pulse input with step and impulse inputs. Raveh[18] also performed parametric variations in order to better understand the numerical issues associated with impulse and step responses, particularly for nonlinear problems. Guendel and Cesnik[19] applied the Aerodynamic Impulse Response (AIR) technique, based on the Volterra theory, to the PMARC aerodynamic panel code. The PMARC/AIR code was applied to a simplified High Altitude Long Endurance (HALE) aircraft for rapid linear and nonlinear aeroelastic analysis of the vehicle.

As mentioned above, various inputs can be used in the time domain (CFD code) to generate GAFs in the frequency domain in order to perform standard, frequency-domain aeroelastic analyses. But if time-domain aeroservoelastic (ASE) analyses are desired, the frequency-domain GAFs are transformed back into the time domain using traditional rational function approximation (RFA) techniques. These techniques include, for example, the well-known Rogers approximation[20] and the Minimum State technique.[21] The RFA techniques transform frequency-domain GAFs into state-space (time domain) models amenable for use with modern control theory and optimization. The process just described transforms time-domain information (CFD results) into frequency-domain information only to have the frequency-domain information transformed back into the time domain.

Gupta et al[22] and Cowan et al[23,24] applied a set of flight testing inputs to an unsteady CFD code and used the information to create a linear ARMA (autoregressive moving average) model that was transformed into state-space form. Although this technique is applied entirely within the time domain, the shape of the inputs applied to the CFD code requires tailoring in order to excite a specific frequency range, resulting in an iterative process. In a similar vein, Rodrigues[25] developed a state-space model for an airfoil in transonic flow using a transonic small-disturbance algorithm. In this paper, a direct approach for efficiently generating linearized unsteady aerodynamic state-space models is presented. Although the present application of the method deals with linearized responses based on linearized impulse responses (linearized Volterra kernels), the method can be formally extended to address nonlinear aeroelastic phenomena via the use of nonlinear

impulse responses (nonlinear Volterra kernels).

The goal of this paper is to develop linearized, unsteady aerodynamic state-space models for prediction of flutter and aeroelastic response using the the parallelized, aeroelastic capability of the CFL3Dv6 code. The results to be presented provide an important validation of the various phases of the ROM development process. As such, this paper begins with a brief outline of the various phases of the process. This outline is followed by a description of the CFL3Dv6 code and a description of the CFD-based impulse response technique. The Eigensystem Realization Algorithm (ERA),[26] which transforms an impulse response into state-space form, is described. Flutter results for the AGARD 445.6 Aeroelastic Wing using the CFL3Dv6 code are presented, including computational costs. Unsteady aerodynamic state-space models are then generated and coupled with a structural model within a MATLAB/SIMULINK[27] environment for rapid calculation of aeroelastic responses including flutter. Aeroelastic responses computed directly using the CFL3Dv6 code are compared with the aeroelastic responses computed using the CFD-based ROM within the MATLAB/SIMULINK environment.

Description of Methods

The following subsections describe the parallelized, aeroelastic version of the CFL3Dv6 code and the two primary phases of the ROM development process. The first phase involves the identification of unsteady aerodynamic impulse responses; the second phase involves the transformation of these impulse responses into state-space form. Step responses can be used in lieu of impulse responses since these functions are related via differentiation and both functions provide equivalent levels of excitation to a given system. Preference of one function over the other will be mentioned when appropriate.

CFL3Dv6 Code

The computer code used in this study is the CFL3Dv6 code, which solves the three-dimensional, thin-layer, Reynolds averaged Navier-Stokes equations with an upwind finite volume formulation.[28,29] The code uses third-order upwind-biased spatial differencing for the inviscid terms with flux limiting in the presence of shocks. Either flux-difference splitting or flux-vector splitting is available. The flux-difference splitting method of Roe[30] is employed in the present computations to obtain fluxes at cell faces. There are two types of time discretization available in the code. The first-order backward time differencing is used for steady calculations while the second-order backward time differencing with subiterations is used for static and dynamic aeroelastic calculations. Furthermore, grid sequencing for steady state and multigrid and local pseudo-time stepping for time marching solutions are employed.

One of the important features of the CFL3D code is its capability of solving multiple zone grids with one-to-one connectivity. Spatial accuracy is maintained at zone boundaries, although subiterative updating of boundary information is required. Coarse-grained parallelization using the Message Passing Interface (MPI) protocol can be utilized in multiblock computations by solving one or more blocks per processor. When there are more blocks than processors, optimal performance is achieved by allocating an equal number of blocks to each processor. As a result, the time required for a CFD-based aeroelastic computation can be dramatically reduced.

In this paper, multiblock MPI parallel aeroelastic computations, including flutter, for the AGARD 445.6 Aeroelastic Wing are performed using 96 flowfield blocks. In order to achieve an optimal division of grid points, it is necessary to place flow field block boundaries near a moving solid surface (the wing). The multiblock boundary and interior movement scheme allows the user to place block boundaries near surfaces as necessary for optimal parallelization. Boundaries interior to the fluid domain near a surface respond to the local surface motion. As the wing moves, block boundaries move to maintain integrity of block interfaces and the airfoil surface.

Because the CFD and computational structural mechanics (CSM) meshes usually do not match at the interface, CFD/CSM coupling requires a surface spline interpolation between the two domains. The interpolation of CSM mode shapes to CFD surface grid points is done as a preprocessing step. Modal deflections at all CFD surface grids are first generated. Modal data at these points are then segmented based on the splitting of the flow field blocks. Mode shape displacements located at CFD surface grid points of each segment are used in the integration of the generalized modal forces and in the computation of the deflection of the deformed surface. The final surface deformation at each time step is a linear superposition of all the modal deflections.

ROM Development Process

An outline of the ROM development process is as follows:

1. Implementation of impulse response technique into aeroelastic CFD code;

2. Computation of impulse responses for each mode of an aeroelastic system using the aeroelastic CFD code;

3. Impulse responses generated in Step 2 are input into the ERA;

4. Evaluation/validation of the state-space models generated in Step 3;

Steps 1 and 2 are described in greater detail in the references that address Volterra-based Reduced-Order

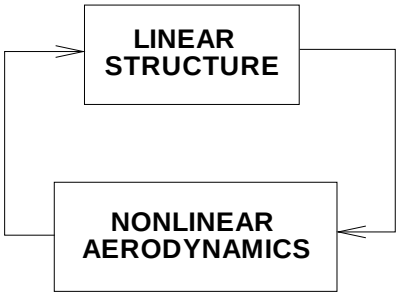

Fig. 1 Coupling of structure and aerodynamics within an aeroelastic CFD code.

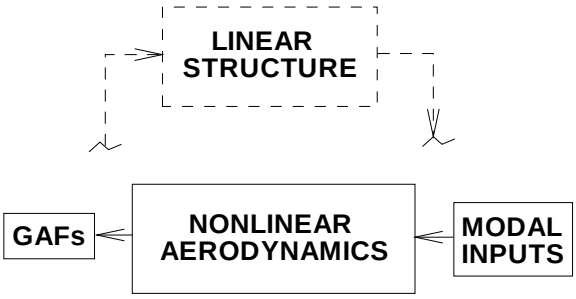

Fig. 2 Identification of generalized aerodynamic forces (GAFs).

Models (ROMs) such as Refs. 3-7. The basic premise of Volterra-based ROMs is the extraction of linear and nonlinear kernel functions that capture the input-output functional relationship between, for example, unsteady motion of a wing (input) and the resultant loads created by that motion (output). For Volterra-based ROMs, these kernel functions are linearized and nonlinear impulse response functions. The relevant aspects of Step 1 and Step 2 are discussed in the next two subsections. Details of Step 3 are presented in the third subsection. Step 4 is presented in the Results section of the paper.

CFD-Based Discrete Unit Impulse Response Technique

An aeroelastic system can be viewed as the coupling of an unsteady aerodynamic system (flow solver) with a structural system (Figure 1). The present study focuses on the development of an unsteady aerodynamic ROM (Figure 2) that is then coupled to a structural model for aeroelastic analyses.

A standard technique for computing linearized generalized aerodynamic forces (GAFs) for an aeroelastic system with n modes using a CFD code is the application of a Greens function (influence function) approach. Using the CFD code, each mode is individually excited to obtain the response of all the modes to this excitation. This process is applied to all n modes, resulting in an n by n "matrix" of responses. The term "matrix" is in quotes to indicate that the responses obtained using this method are usually time-domain functions rather than constants that usually populate a standard matrix.

This technique is a linearization by virtue of the fact that, in a computational aeroelastic analysis, the input to the nonlinear flow solver is the total physical deformation of the wing consisting of the summed total of all the modes of interest. By applying a separate excitation to each mode through the nonlinear flow solver, the total nonlinear aeroelastic response is being approximated by a linear superposition of its individual responses. For a linear flow solver, this approach would be exact. Consistent with this assumption, this approximation is valid only for small input amplitudes. This is not necessarily a drawback as, quite often, the linearized dynamic aeroelastic response about a nonlinear steady (or static aeroelastic) condition is a reasonable representation of the nonlinear aeroelastic system under investigation.

There are three types of modal excitation inputs that are typically used when implementing this technique. The first is a brute-force approach based on the input of sine waves of individual frequencies. The individual modal responses to these inputs for n modes and r frequencies requires n times r separate code evaluations. In addition, the time length required for each one of these evaluations can be quite large (i.e., computationally expensive) in order to get an adequate number of cycles for adequate frequency resolution, especially for the lower frequencies. This approach is clearly the least efficient.

A second, more elegant approach, involves the use of an exponential (Gaussian) pulse.[15] The exponential pulse can be shaped to excite a particular range of frequencies. Because an exponential pulse excites a pre-selected frequency range, only one code evaluation is required per mode. This is a significant computational savings compared to the brute-force approach, but shape optimization of the exponential pulse may be required when targeting a particular frequency range. In addition, the exponential pulse appears to be strictly limited to linearized analyses. Whereas the impulse function finds formal application

to nonlinear problems via the Volterra theory, the inclusion of the exponential pulse within a Volterra-type theoretical framework is undefined.

A third, recently-developed, approach consists of replacing the exponential pulse input with a unit impulse.[8–11] The unit impulse excites the entire frequency range of a system so that shape optimization to excite a particular range of frequencies is not necessary. In addition, due to the simplicity of the input and the short amount of time required for convergence, each solution is computed with significant computational efficiency. Raveh et al[11] applied this technique successfully to the AGARD 445.6 Aeroelastic Wing using step and impulse responses. Convolution of the step responses with sinusoids of varying frequency yielded frequency-domain GAFs that were then used for frequency-domain aeroelastic analyses. If desired, a more direct approach for computing frequency-domain GAFs is to apply a Fast Fourier Transform (FFT) to the impulse responses. An example of this approach is presented in a subsequent section.

System/Observer/Controller Identification Toolbox (SOCIT)

In structural dynamics, the realization of discrete-time state-space models that describe the modal dynamics of a structure has been enabled by the development of algorithms such as the Eigensystem Realization Algorithm (ERA)[26] and the Observer Kalman Identification (OKID)[31] Algorithm. These algorithms perform state-space realizations by using the Markov parameters (discrete-time impulse responses) of the systems of interest. These algorithms have been combined into one package known as the System/Observer/Controller Identification Toolbox (SOCIT)[32] developed at NASA Langley Research Center.

The primary algorithm within the SOCIT group of algorithms used for the present system realization is known as the Eigensystem Realization Algorithm (ERA). A brief summary of the basis of this algorithm follows.

A finite dimensional, discrete-time, linear, time-invariant dynamical system has the state-variable equations

$$x(k+1) = Ax(k) + Bu(k) \tag{1}$$

$$y(k) = Cx(k) + Du(k) \tag{2}$$

where x is an n-dimensional state vector, u an m-dimensional control input, and y a p-dimensional output or measurement vector with k being the discrete time index. The transition matrix, A, characterizes the dynamics of the system. The goal of system realization is to generate constant matrices (A, B, C) such that the output responses of a given system due to a particular set of inputs is reproduced by the discrete-time state-space system described above.

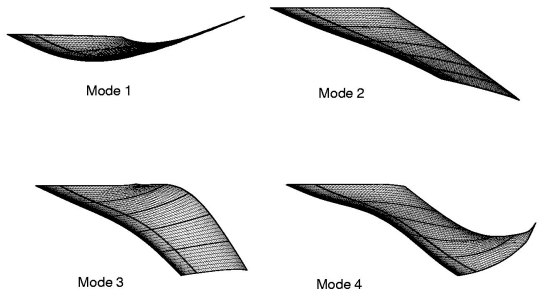

Fig. 3 Aeroelastic modes for the AGARD 445.6 Wing.

For the system of Eqs. (1) and (2), the time-domain values of the systems discrete-time impulse response are also known as the Markov parameters and are defined as

$$Y(k) = CA^{k-1}B \tag{3}$$

with B an (n x m) matrix and C a (p x n) matrix. The ERA algorithm begins by defining the generalized Hankel matrix consisting of the discrete-time impulse responses for all input/output combinations. The algorithm then uses the singular value decomposition (SVD) to compute the A, B, and C matrices.

In this fashion, the ERA is applied to unsteady aerodynamic impulse responses to construct unsteady aerodynamic state-space models. The next section presents computational aeroelastic and unsteady aerodynamic results for the CFL3Dv6 code and for the state-space ROM.

Results

The AGARD 445.6 Aeroelastic Wing has been used extensively by several authors to validate computational methods.[16,22,33] Although the aeroelastic behavior of this wing is fairly benign (weakly nonlinear), the aeroelastic data from the flutter test of this wing provides a good starting point for validation of computational techniques.[34] The wing is a 45-degree swept-back wing with a NACA 65A004 airfoil section, panel aspect ratio of 1.65, and a taper ratio of 0.6576. The shapes of the first four structural modes for this wing are presented in Figure 3. The modes are first bending, first torsion, second bending and second torsion. The corresponding modal frequencies in vacuo are 9.60, 38.2, 48.35, and 91.54 Hz. Additional details regarding this wing can be found in the references.

Full CFD Flutter Solution

This section presents results based on the traditional full CFD flutter solution. The flutter solution

is obtained by iterating between the nonlinear aerodynamic system (flow solver) and the structural system at a given Mach number and dynamic pressure entirely within the CFD code. The output of this solution consists of a time history of the generalized coordinates of the aeroelastic system. Depending on the nature of this aeroelastic response (divergent or convergent), a new dynamic pressure is selected and a corresponding flutter solution is computed. This iterative process is used to define the flutter boundary at several Mach numbers. The results presented in this paper are for the solution of the Euler equations within CFL3Dv6.

Figure 4 presents the response of each of the four generalized coordinates at a Mach number of 0.9, a dynamic pressure of 89.3 psf, and a structural damping value (g) of zero. The divergent nature of the first mode indicates that this condition is above the flutter boundary. By performing similar analyses at different dynamic pressures, a dynamic pressure of 75 psf was defined as the flutter dynamic pressure (neutral stability point) for this Mach number. The corresponding flutter frequency was 14.8 Hz. The aeroelastic response at a dynamic pressure of 75 psf is presented as Figure 5, indicating the neutral stability of the aeroelastic system at this condition. These solutions were computed using a non-dimensional time step of 0.3 with 5 subiterations per time step and use of multigrid capability for error reduction and convergence acceleration.

Comparison of flutter results for the full CFD analysis with 0.03 structural damping, the analysis of Lee-Rausch[16] (0.03 structural damping), and the experimental results of Yates et al[34] are presented in Table 1 and Table 2 for Flutter Speed Index (FSI) and Flutter Frequency Ratio (FFR), respectively. Results from the full CFD flutter analysis are consistent with those from Lee-Rausch[16] and other Euler flutter results.[22]

Table 1 Comparison of Flutter Speed Index (FSI) with published results.

M	Exp.	Lee-Rausch	CFL3Dv6
		g=0.03	g=0.03
0.9	0.370	0.352	0.350
0.96	0.308	0.275	0.279

Table 2 Comparison of Flutter Frequency Ratio (FFR) with published results.

M	Exp.	Lee-Rausch	CFL3Dv6
		g=0.03	g=0.03
0.9	0.422	0.425	0.394
0.96	0.365	0.343	0.315

The computational cost for one flutter solution (at a given Mach number and dynamic pressure) is 71 CPU hours for the number of cycles shown in Figures 4 and 5. This is the total CPU cost but, using 96 processors, the actual execution time is approximately 45 minutes on an Origin 2000 cluster. The total time elapsed from the moment the job is submitted for execution, however, can vary depending on the number of other jobs (from different users) in the queue for the computer resources. The total elapsed time for a single flutter solution can therefore range from 45 minutes to several hours if the job queue is busy. In addition, although four cycles of the lowest frequency mode appear to be sufficient for visually determining the stability of the system, accurate computation of the relevant aeroelastic frequency and damping requires additional cycles. If the number of cycles is doubled to eight cycles, the computational costs increase proportionately to 142 CPU hours and 90 minutes of execution time. The to-

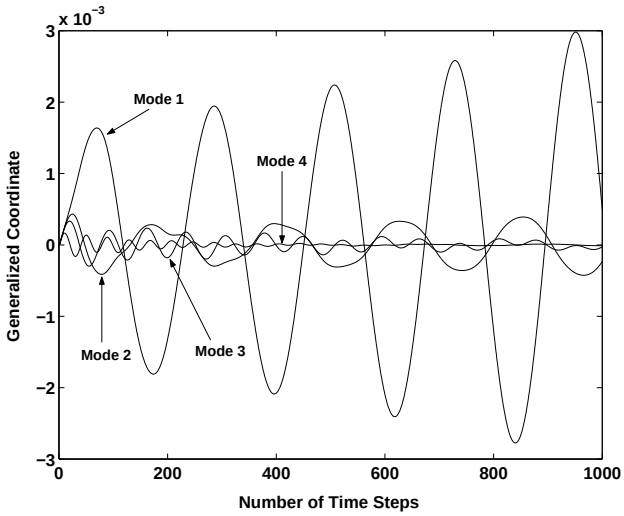

Fig. 4 Aeroelastic transients in terms of generalized coordinates at M=0.9 and Q=89.3 psf.

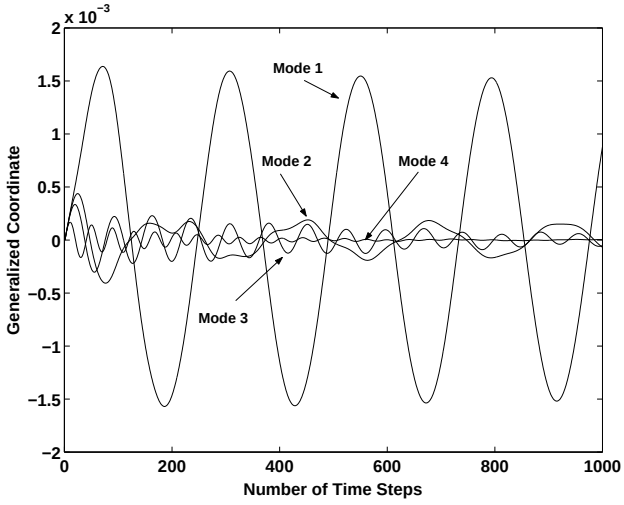

Fig. 5 Aeroelastic transients in terms of generalized coordinates at M=0.9 and Q=75.0 psf.

tal time elapsed can range from 90 minutes to several hours depending on the number of jobs in the queue.

These costs are, of course, a function of the aeroelastic properties of the system under investigation. The use of parallel processing clearly provides significant improvement in computational time. However, the computational costs (CPU) are still high because parallelization, obviously, does not reduce the amount of computation that needs to be done. Nonetheless, this is a significant improvement over computations performed on a serial platform.

Assuming four dynamic pressure solutions per Mach number, the cost of computing a flutter point (at one Mach number) is 568 CPU hours, requiring at least 360 minutes of execution time. The actual time invested, however, can be on the order of days since the value of dynamic pressure selected for the subsequent analysis depends on the results obtained from the previous analysis. If additional analyses involving parametric variations of structural parameters (damping and frequencies) are needed, additional flutter solutions would be required, increasing computational costs (CPU and time). Finally, as can be seen, the output of traditional CFD-based flutter analyses are aeroelastic transients which provide frequency and damping information at a given flight condition. These transients can certainly be used to define the flutter boundary of the aeroelastic system under investigation but do not comprise a mathematical model of the system itself. In order to develop a mathematical model of the system itself, a ROM is needed. The next sections present results for the development and validation of a ROM using the CFL3Dv6 code.

Unsteady Aerodynamic System Identification

Step/Impulse Responses

Identification of the unsteady aerodynamic system begins with the excitation of each mode using a step or impulse input. Although the frequency content of both responses is identical, the use of the impulse response is beneficial when computing the frequency-domain generalized aerodynamic forces (GAFs) and in the application of the ERA code for the generation of state-space models. Raveh et al[17] indicate improved numerical robustness for the step response over the impulse response. Selection of one input over the other may depend on the particular configuration and problem under investigation.

Consistent with the linearization process described in a previous section and in order to reduce the possibility of numerical problems with aeroelastically-deforming grids, small amplitudes are used with this technique. The mode-by-mode excitation for the AGARD 445.6 Aeroelastic Wing using impulse and step inputs is performed using the first four elastic modes of the wing. The mode-by-mode excitation technique provides the unsteady aerodynamic response

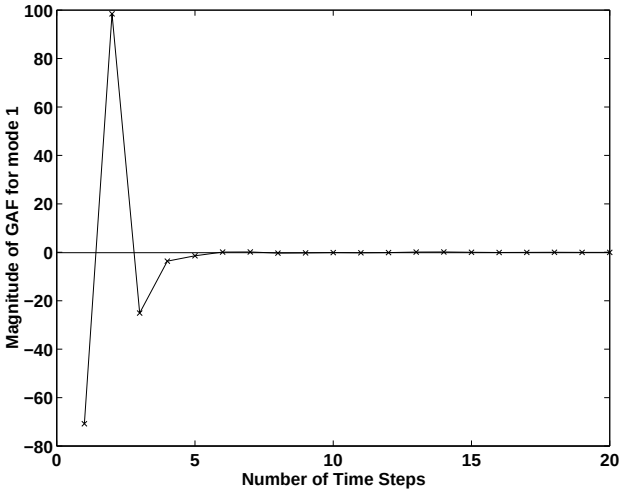

Fig. 6 Impulse response in mode 1 due to mode 1, M=0.9.

in all four modes due to an excitation of one of the modes. In this fashion, the matrix of four-by-four response functions is developed, resulting in a total of sixteen response functions.

Figure 6 presents the impulse response of the first mode due to an impulse input in the first mode. This response was computed using a nondimensional time step of 0.3, a modal amplitude of 0.001 with 5 subiterations and multigrid for improved convergence and error minimization. As can be seen, the response is well-behaved and numerically stable.

An important point in the generation of step and impulse responses is the need to maintain the rate-of-change of the excitation input (the modal velocity in this case) to a reasonable value. For an aeroelastic analysis, the modal velocity is defined as the modal amplitude of the excitation input divided by the nondimensional time step. Values on the order of unity appear to be the most robust although values as high as ten have worked. The adherence to this range of values for the modal velocity provides the necessary numerical stability to generate these responses for unsteady motions with deforming grids. For rigidly deforming grids, such as plunging and pitching motions, this limitation can be relaxed since the rigid-body amplitudes and velocities can be defined independently. For an aeroelastically-deforming grid, the modal amplitude is input explicitly while the modal velocity is computed implicitly based on the amplitude of motion and the time step. This is consistent with results obtained by Raveh et al[11] and Silva and Raveh.[35]

Successful and accurate identification of impulse and step responses requires careful consideration of time/frequency resolution issues. In addition, the effect of input amplitude on the convergence of the solution and verification of linear/nonlinear behavior must be addressed. These issues are extremely important

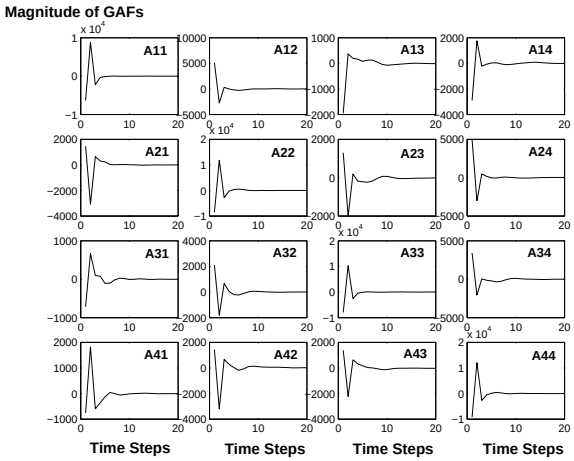

Fig. 7 Impulse response GAFs for all four modes.

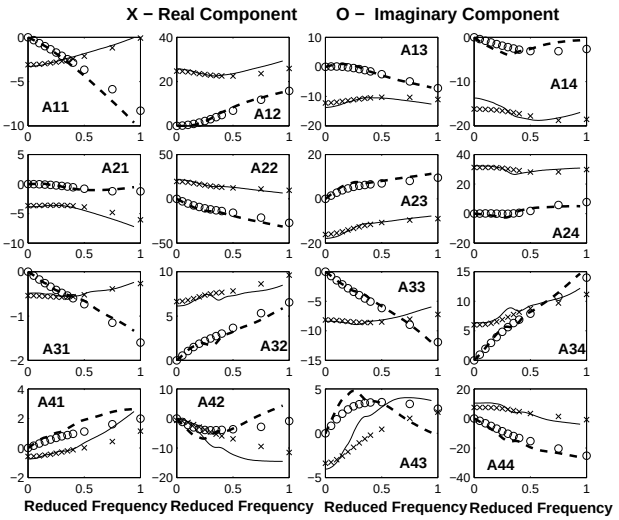

Fig. 8 Comparison of frequency-domain GAFs from the impulse response GAFs with GAFs from a linear unsteady aerodynamic method.

and are discussed in greater detail in the Appendix.

Time- and Frequency-Domain GAFs

For the present four-mode aeroelastic system, four separate analyses were performed to compute the necessary impulse response GAFs. Figure 7 presents the sixteen impulse response GAFs due to each of the four modes at a Mach number of 0.9 computed using a modal amplitude of 0.001 and a nondimensional time step size of 0.3. Using a total of 2000 time steps and accounting for various nondimensional parameters, this nodimenstional time step size and number of time steps translates to a reduced frequency resolution of 0.009. Clearly, all impulse responses are well behaved. It is noticed that the GAFs with the largest magnitudes correspond to the "diagonal" responses (A11, A22, A33, and A44). This makes physical sense since a mode exhibits the largest response to its own excitation.

Once all of the impulse response GAFs were computed for all of the modes, an FFT of each impulse response GAF yielded the frequency-domain GAF. Figure 8 presents a comparison of the resultant FFTs of the impulse response GAFs from Figure 7 with frequency-domain results computed using a linear unsteady aerodynamic method.[11] As can be seen, the comparison is very good for most of the GAFs with some discrepancies at the higher-frequency modes. Additional analyses are required to determine if these differences are due to physically nonlinear effects or if they are due to numerical/computational differences. This type of comparison with a linear unsteady aerodynamic code can also be used to ascertain the level of linear/nonlinear content of the CFD-based unsteady aerodynamics. The results of Figure 8 show a close correlation of the linearized (CFD-based) GAFs with the fully linear GAFs.

ROM Flutter Solution

Unsteady Aerodynamic State-Space Models

The ERA was then used to transform the impulse response GAFs from Figure 7 into state-space form. This process is performed within MATLAB and executes quickly. Several options are available to allow the user to reduce the size of the resultant state-space matrices depending on the desired frequency range or importance of particular modes. For the present analysis, no order reduction of this type was performed in order to establish a baseline performance level to which subsequent order reductions could be compared in future analyses. The resultant system quadruple (A,B,C,D) is of 196th order with four inputs and four outputs corresponding to the four modes. Although this is a high order, it is important to mention that this state-space model contains the entire range of unsteady aerodynamic frequencies extracted from the CFD code. Significant reductions in order can be achieved by defining a frequency bandwidth of interest, analogous to the procedure in the frequency domain when rational function approximations are developed. Given modern computational power, however, this high-order system poses no computational issues with respect to memory storage or computational speed for aeroelastic analyses. The order, however, may need to be reduced for subsequent ASE design studies.

The state-space model of the CFD-based unsteady aerodynamic system can be used to compute the response to arbitrary inputs without costly re-execution of the CFD code. Figure 9 is a comparison of the responses in the first mode due to an input consisting of a narrow exponential pulse applied to all four modes simultaneously. One of the responses in the figure was computed using CFL3Dv6 directly while

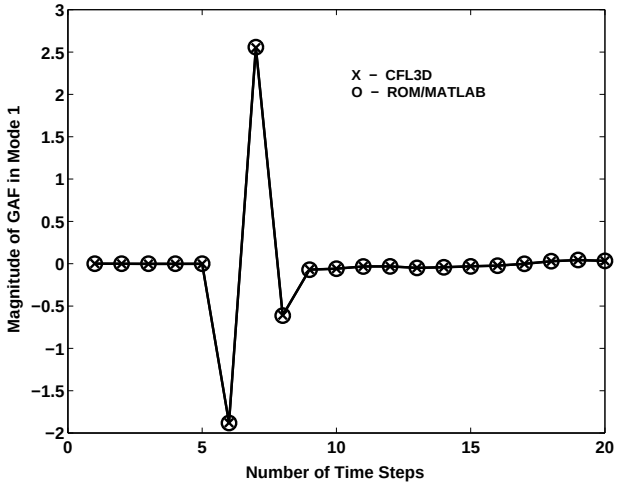

Fig. 9 Comparison of GAFs for the CFD and state-space unsteady aerodynamic model.

the other response was computed using the state-space model within MATLAB. The exact comparison verifies the accuracy of the unsteady aerodynamic state-space model. The response from the unsteady aerodynamic state-space model was generated within seconds while the response from CFL3Dv6 required approximately two hours of total elapsed computing time and 177 CPU hours.

Flutter

Coupling the state-space model of the unsteady aerodynamic system with a state-space model of the structure within MATLAB/SIMULINK results in a state-space aeroelastic system shown in Figure 10. The aeroelastic response of the system is a function of the initial conditions of the structure and the dynamic pressure.

In order to validate this state-space aeroelastic system, simulations were performed at various dynamic pressures. Figure 11 presents the generalized coordinate time histories and the corresponding generalized coordinate FFTs at zero dynamic pressure (wind-off). The zero dynamic pressure attenuates all aerodynamic effects, leaving only structural effects. With zero structural damping, the response consists of, in the time domain, the simple harmonic motion of the uncoupled vibration modes and, in the frequency domain, frequency spikes of the uncoupled vibration modes: 9.60, 38.2, 48.35, and 91.54 Hz.

At a dynamic pressure of 50 psf, Figure 12, the effect of aerodynamic damping is evident in the decaying response of the generalized coordinate time histories. The associated modal frequency spikes at this condition are no longer uncoupled as they were in Figure 11.

Finally, at a dynamic pressure of 75 psf, Figure 13, flutter is evident. A close-up of this aeroelastic transient is presented as Figure 14. This result compares

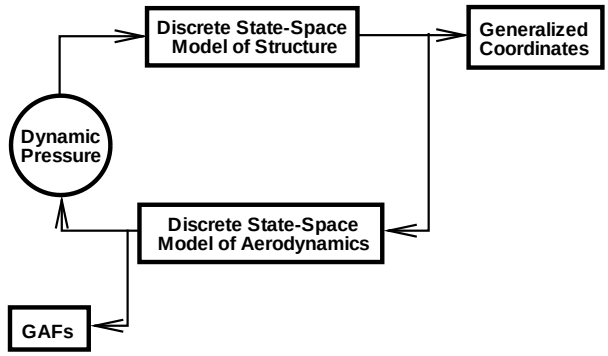

Fig. 10 SIMULINK model of the aeroelastic system.

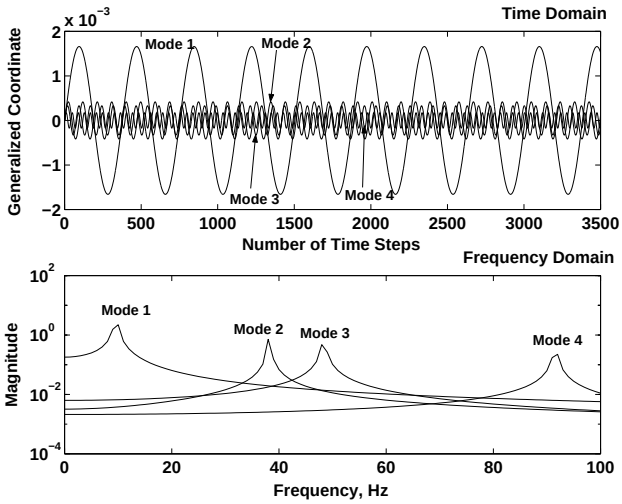

Fig. 11 Aeroelastic response of the state-space aeroelastic system at M=0.9 and Q=0 psf.

identically with that of Figure 5, which was computed using CFL3Dv6 directly. In fact, the ROM results compare identically with results using CFL3Dv6 directly at all dynamic pressures investigated.

These aeroelastic transients are computed in seconds within MATLAB/SIMULINK, thus allowing a larger number of cycles to be computed for improved frequency resolution. In addition, if parametric variations that involve the structure are desired (structural damping, updated frequencies, etc), the analyses can be performed using this approach since the unsteady aerodynamic system is unaffected by these variations.

These results validate the ROM methodology presented and are examples of a new and powerful tool available to the aeroelastician. Most importantly, the state-space models developed are suitable for use

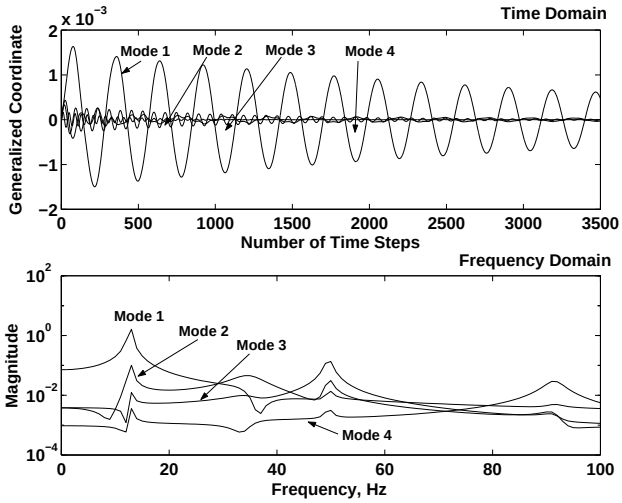

Fig. 12 Aeroelastic transients in terms of generalized coordinates for the state-space system at M=0.9, Q=50 psf, and g=0.0.

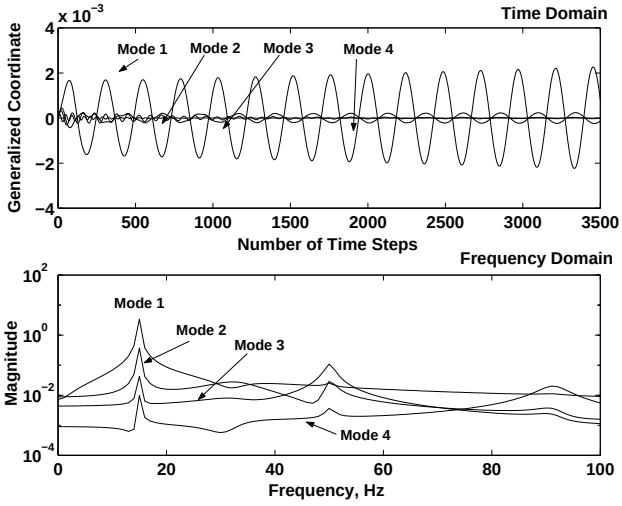

Fig. 13 Aeroelastic transients in terms of generalized coordinates for the state-space system at M=0.9, Q=75 psf, and g=0.0.

within a mutidisciplinary design environment, including ASE analysis and design.

Concluding Remarks

A reduced-order model (ROM) was developed for aeroelastic analysis using the recently-developed, parallelized CFL3D version 6.0 computational fluid dynamics (CFD) code. Flutter results for the AGARD 445.6 Wing, computed using CFL3D directly, were presented, including a discussion of the associated computational costs. The ROM of the unsteady aerodynamic system, in state-space form, was developed using modal impulse responses. Important numerical issues associated with the computation of the impulse responses including time/frequency resolution

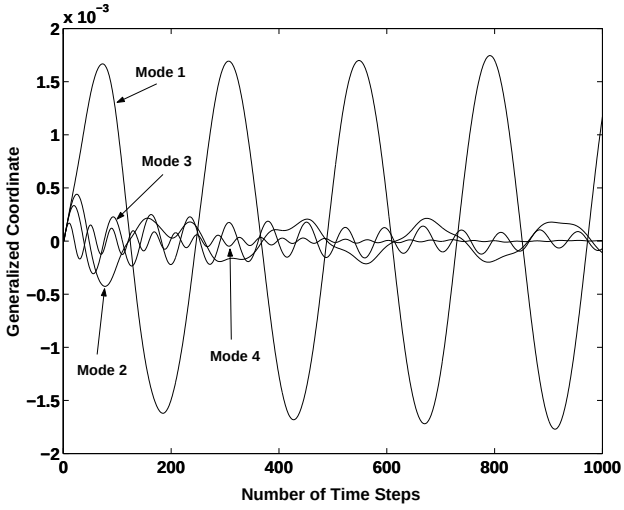

Fig. 14 Close-up of the aeroelastic transients for the state-space system at M=0.9, Q=75 psf, and g=0.0.

and amplitude-dependent convergence issues were presented. The unsteady aerodynamic state-space ROM was then combined with a state-space model of the structure to create an aeroelastic simulation using the MATLAB/SIMULINK environment. The MATLAB/SIMULINK ROM was used to rapidly compute aeroelastic transients including flutter. The ROM shows excellent agreement with the aeroelastic analyses computed using the CFL3Dv6.0 code directly but at significantly lower computational costs. The aeroelastic state-space models generated are then suitable for use in a multidisciplinary, design environment including computational aeroservoelasticity (ASE).

Appendix

Time Step/Frequency Issues

Successful development of a state-space model of an unsteady aerodynamic system requires an understanding of the relationship between the time step used for the numerical discretization and the frequency content associated with that particular discretization. It is well known from signal processing theory that the frequency resolution of a given discretization, ΔF, is inversely proportional to the product of the number of time steps, N, and the discretizing time step, ΔT, or

$$\Delta F = \frac{1}{N * \Delta T}$$

Standard numerical analysis states that a smaller time step is more accurate due to a reduction of the error terms associated with most numerical discretizations. A smaller value of frequency resolution is also preferred so that appropriate frequency-domain phenomena can be captured accurately. This is important, for example, when an aeroelastic system contains modes that are closely-spaced in frequency. Therefore, a large

number of time steps is needed to satisfy both of these requirements.

This observation leads to an important consideration when impulse and step responses are generated for subsequent use in a convolution or state-space framework. If the time step is reduced (in an effort to reduce numerical error) while the number of time steps is kept constant, the frequency resolution is increased. This increased frequency resolution may lead to inaccurate representation of frequency content. Therefore, in an attempt to improve the accuracy of the step or impulse response, by decreasing the time step (without regard to the number of time steps), the overall predictive capability of the step or impulse response may in fact be compromised. Guendel[19] and Raveh[18] indicate that decreasing the time step resulted in decreased predictive accuracy of the impulse response. The preceding observation may explain this counterintuitive and unexpected result.

A small nondimensional time step size (0.001) can reduce the numerical error, but it places a limit on the modal amplitude allowed since it affects the discretized modal velocity. In addition, a small time step requires a large number of time steps in order to achieve a small frequency resolution. However, using the subiteration capability available within the CFL3Dv6 code, larger time steps can be used while controlling the level of the numerical error. The ability to safely use larger time steps provides a significant benefit with respect to the time/frequency resolution issue. In particular, the use of a larger nondimensional time step permits the use of a larger input amplitude (modal velocity) to excite nonlinear terms. At the same time, a larger nondimensional time step yields a smaller frequency resolution for a given number of time steps. Therefore, the use of a larger nondimensional time step allows larger input amplitudes and a smaller frequency resolution for less time steps than would be required for a smaller nondimensional time step. This provides valuable computational efficiency.

Amplitude/Convergence

The subiteration capability must also be used for controlling the numerical error that is influenced by the amplitude of the excitation input. Figure 15 is a comparison of the residual (measure of error) for two modal step responses with different input amplitudes at a Mach number of 0.9. In the figure, three regions are presented: Steady Solution, Transient Unsteady Solution, and Final Unsteady Solution. The Steady Solution consists of the steady-state solution of the Euler equations. The steady-state solution is then used as the starting point for the unsteady solution. The steady-state solution does not contain the time-derivative terms needed for the unsteady solution and, as a result, the introduction of the unsteady terms at the start of the unsteady solution induces the

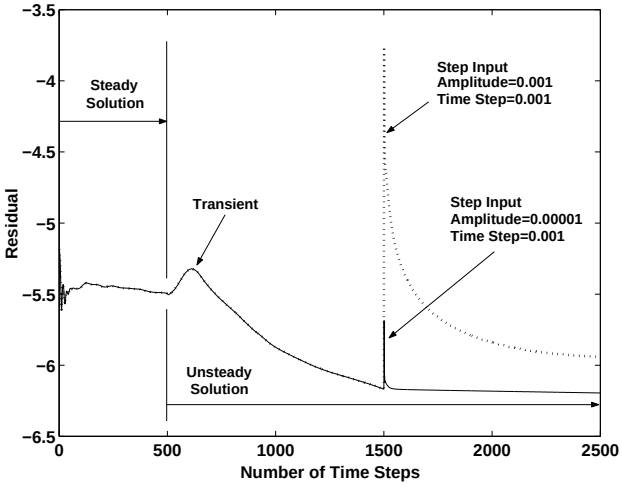

Fig. 15 Comparison of the residual errors for two step responses at different amplitudes.

numerical transient shown in the figure (Transient Unsteady Solution). The step input is therefore delayed so that the step response will not be contaminated by the transient at the start of the unsteady solution.

An additional point to be made is that the two step responses converge to different error levels even though 10 subiterations per time step are being applied to each solution. The larger amplitude therefore needs an increased number of subiterations to reduce its error level to the level of the smaller amplitude response. Even though the difference in amplitudes is large for this comparison (two orders of magnitude), this result emphasizes the importance of tracking the numerical error as a function of amplitude and applying the subiteration procedure appropriately.

Proper development of a CFD-based ROM requires careful attention to the creation and growth of numerical error so that relevant physical characteristics of a system are not clouded by nonphysical noise. It is also strongly recommended that linearity tests be performed at the conditions of interest. A simple linearity test consists of applying inputs at various amplitudes to determine the range of amplitudes over which linear conditions are satisfied. A second linearity test consists of validating the assumption of modal superposition by comparing the response to an excitation of all the modes with the sum of the responses for individual modes. These types of tests were performed for the present analysis but are not included in the paper. The point to be made is that, for the conditions at which analyses were performed, and the range of amplitudes investigated, the assumption to linearize the aeroelastic system was validated.

References

[1]Bisplinghoff, R. L. and Ashley, H., *Principles of Aeroelasticity*, Dover Publications, Inc., New York, 1975.

[2]Wagner, H., "Uber die Entstehung des dynamischen Auftriebes von Tragflugeln," *Mathematical Mechanics*, 1925.

[3]Kussner, H. G., "Schwingungen von Flugzeugflugeln," *Luftfahrtforschung*, Vol. 4, 1929.

[4]Theodorsen, T., "General Theory of Aerodynamic Instability and the Mechanism of Flutter," Tech. rep., NACA, Report 496, 1935.

[5]Giesing, J. P., Kalman, T. P., and Rodden, W. P., "Subsonic Unsteady Aerodynamics for General Configurations, Part I. Direct Application of the Nonplanar Doublet Lattice Method," Report AFFDL-TR-71-5, Vol. I, November 1971.

[6]Beran, P. S. and Silva, W. A., "Reduced-Order Modeling: New Approaches for Computational Physics," *Presented at the 39th AIAA Aerospace Sciences Meeting, 8-11 January 2001, Reno,NV*, January 2001.

[7]Silva, W. A., Beran, P. S., Cesnik, C. E. S., Guendel, R. E., Kurdila, A., Prazenica, R. J., Librescu, L., Marzocca, P., and Raveh, D., "Reduced-Order Modeling: Cooperative Research and Development at the NASA Langley Research Center," *CEAS/AIAA/ICASE/NASA International Forum on Aeroelasticity and Structural Dynamics*, June 2001.

[8]Silva, W. A., "Reduced-Order Models Based on Linear and Nonlinear Aerodynamic Impulse Responses," *CEAS/AIAA/ICASE/NASA International Forum on Aeroelasticity and Structural Dynamics*, June 1999.

[9]Silva, W. A., *Discrete-Time Linear and Nonlinear Aerodynamic Impulse Responses for Efficient CFD Analyses*, Ph.D. thesis, College of William & Mary, December 1997.

[10]Silva, W. A., "Application of Nonlinear Systems Theory to Transonic Unsteady Aerodynamic Responses," *Journal of Aircraft*, Vol. 30, 1993, pp. 660–668.

[11]Raveh, D. E., Levy, Y., and Karpel, M., "Aircraft Aeroelastic Analysis and Design Using CFD-Based Unsteady Loads," *41st Structures, Structural Dynamics, and Materials Conference*, No. 2000-1325, Atlanta, GA, April 2000.

[12]Dowell, E. H. and Hall, K. C., "Modeling of Fluid-Structure Interaction," *Journal of Annual Review of Fluid Mechanics*, Vol. 33, 2001, pp. 445–490.

[13]Dowell, E. H., Hall, K. C., and Romanowski, M. C., "Eigenmode Analysis in Unsteady Aerodynamics: Reduced Order Models," *Applied Mechanical Review*, Vol. 50, No. 6, June 1997, pp. 371–385.

[14]Baker, M. L., *Model Reduction of Large, Sparse, Discrete Time Systems with Application to Unsteady Aerodynamics*, Ph.D. thesis, University of California at Los Angeles, 1996.

[15]Seidel, D. A., Bennett, R. M., and Ricketts, R. H., "Some Recent Applications of XTRAN3S," AIAA Paper 83-1811.

[16]Lee-Rausch, E. M. and Batina, J. T., "Wing Flutter Computations Using an Aerodynamic Model Based on the Navier-Stokes Equations," *Journal of Aircraft*, Vol. 33, 1993, pp. 1139–1148.

[17]Raveh, D., Levy, Y., and Karpel, M., "Aircraft Aeroelastic Analysis and Design Using CFD-Based Unsteady Loads," AIAA Paper 2000-1325, April 2000.

[18]Raveh, D. and Mavris, D., "Reduced-Order Models Based on CFD Impulse and Step Responses," *Proceedings of the 42nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference and Exhibit*, No. 2001-1527, Seattle, WA, April 2001, AIAA-01-1527.

[19]Guendel, R. E. and Cesnik, C. E. S., "Aerodynamic Impulse Response of a Panel Method," AIAA Paper No. 2001-1210, to be presented at the 42nd Structures, Structural Dynamics, and Materials Conference, Seattle, WA.

[20]Roger, K. L., "Airplane Math Modeling Methods for Active Control Design," Agard-cp-228, Aug. 1977.

[21]Karpel, M., "Time Domain Aeroservoelastic Modeling Using Weighted Unsteady Aerodynamic Forces," *Journal of Guidance, Control, and Dynamics*, Vol. 13, No. 1, 1990, pp. 30–37.

[22]Gupta, K. K., Voelker, L. S., Bach, C., Doyle, T., and Hahn, E., "CFD-Based Aeroelastic Analysis of the X-43 Hypersonic Flight Vehicle," *Proceedings of the 39th Aerospace Sciences Meeting and Exhibit*, No. 2001-0712, Reno, CA, Jan. 2001.

[23]Cowan, T. J., Jr., A. S. A., and Gupta, K. K., "Accelerating CFD-Based Aeroelastic Predictions Using System Identification," *36th AIAA Atmospheric Flight Mechanics Conference and Exhibit*, No. 2001-0712, Boston, MA, Aug. 1998, pp. 85–93, AIAA-98-4152.

[24]Cowan, T. J., Jr., A. S. A., and Gupta, K. K., "Development of Discrete-Time Aerodynamic Model for CFD-Based Aeroelastic Analysis," *Proceedings of the 37th Aerospace Sciences Meeting and Exhibit*, No. 1999-0765, Reno, CA, Jan. 1999, AIAA-99-0765.

[25]Rodrigues, E. A., "Linear/Nonlinear Behavior of the Typical Section Immersed in Subsonic Flow," *Proceedings of the 42nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference and Exhibit*, No. 2001-1584, Seattle, WA, April 2001, AIAA-01-1584.

[26]Juang, J.-N. and Pappa, R. S., "An Eigensystem Realization Algorithm for Modal Parameter Identification and Model Reduction," *Journal of Guidance, Control, and Dynamics*, Vol. 8, 1985, pp. 620–627.

[27]"Registered Product of the MathWorks, Inc." .

[28]Krist, S. L., Biedron, R. T., and Rumsey, C. L., "CFL3D User's Manual Version 5.0," Nasa langley research center, 1997.

[29]Bartels, R. E., "Mesh Strategies for Accurate Computations of Unsteady Spoiler and Aeroelastic Problems," *AIAA Journal of Aircraft*, Vol. 37, 2000, pp. 521–525.

[30]Roe, P. L., "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes," *Journal of Computational Physics*, Vol. 43, 1981, pp. 357–372.

[31]Juang, J.-N., Phan, M., Horta, L. G., and Longman, R. W., "Identification of Observer/Kalman Filter Markov Parameters: Theory and Experiments," *Journal of Guidance, Control, and Dynamics*, Vol. 16, 1993, pp. 320–329.

[32]Juang, J.-N., *Applied System Identification*, Prentice-Hall PTR, 1994.

[33]Gordnier, R. E. and Melville, R. B., "Transonic Flutter Simulations Using an Implicit Aeroelastic Solver," *AIAA Journal of Aircraft*, Vol. 37, 2000, pp. 872–879.

[34]E. C. Yates, J., Land, N. S., and J. T. Foughner, J., "Measured and Calculated Subsonic and Transonic Flutter Characteristics of a 45-degree Swept-Back Wing Planform in Air and in Freon-12 in the Langley Transonic Dynamics Tunnel," Tech. rep., NASA, TN D-1616, 1963.

[35]Silva, W. A. and Raveh, D. E., "Development of Unsteady Aerodynamic State-Space Models from CFD-Based Pulse Responses," AIAA Paper No. 2001-1213, presented at the 42nd Structures, Structural Dynamics, and Materials Conference, Seattle, WA.