JOURNAL OF AIRCRAFT
Vol. 45, No. 4, July–August 2008

Simultaneous Excitation of Multiple-Input/Multiple-Output CFD-Based Unsteady Aerodynamic Systems

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

DOI: 10.2514/1.34328

A significant improvement to the development of computational-fluid-dynamics-based unsteady aerodynamic reduced-order models is presented. The improvement involves the simultaneous excitation of the structural modes of the computational-fluid-dynamics-based unsteady aerodynamic system. This improvement enables the computation of the unsteady aerodynamic state-space model using a single computational fluid dynamics execution, independent of the number of structural modes. Two new types of input functions are presented that can be used for the simultaneous excitation of the unsteady aerodynamic system via the excitation of the structural modes. Results are presented for a semispan configuration using the CFL3Dv6.4 code.

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 Sears's function [5] (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 [6].

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. As a result, the integration of traditional computational aeroelastic simulations into preliminary design activities involving disciplines such as aeroelasticity, aeroservoelasticity, 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 when compared with the original system, thereby addressing the first challenge.

At present, the development of computational fluid dynamics (CFD)-based ROMs is an area of active research at several government, industry, and academic institutions [7–11]. Development of ROMs based on the Volterra theory is one of several ROM methods currently under development [12–16]. 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 may be considered to be the ROMs of the particular nonlinear system under investigation. Alternatively, upon transformation of the linearized and nonlinear impulse responses into state-space form, the state-space models generated may also be considered ROMs.

Various inputs can be used in the time domain (CFD code) to generate generalized aerodynamic forces (GAFs) in the frequency domain to perform standard frequency-domain aeroelastic analyses. But if time-domain aeroservoelasticity analyses are desired, the frequency-domain GAFs are transformed back into the time domain using traditional rational function approximation techniques. These techniques include, for example, the well-known Roger approximation [17] and the minimum state technique [18]. The rational function approximation 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. [19] and Cowan et al. [20,21] applied a set of flight-testing inputs to an unsteady CFD code and used the information to create a linear autoregressive moving average (ARMA) 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 to excite a specific frequency range, resulting in an iterative process. In a similar vein, Rodrigues [22] developed a state-space model for an airfoil in transonic flow using a transonic small-disturbance algorithm.

Presented as Paper 1988 at the 2007 Structures, Structural Dynamics, and Materials Conference, Honolulu, HI, 23–26 April 2007; received 29 August 2007; revision received 6 March 2008; accepted for publication 6 March 2008.

*Senior Research Scientist, Aeroelasticity Branch. Associate Fellow AIAA.

In the present paper, a direct approach for efficiently generating linearized unsteady aerodynamic state-space models is presented. Silva and Bartels [8] introduced the development of linearized unsteady aerodynamic state-space models for prediction of flutter and aeroelastic response using the parallelized aeroelastic capability of the CFL3Dv6.4 code. The results presented provided an important validation of the various phases of the ROM development process. The Eigensystem realization algorithm (ERA) [23], which transforms an impulse response (one form of ROM) into state-space form (another form of ROM), was applied for the development of the aerodynamic state-space models. The ERA is part of the SOCIT (System/Observer/Controller Identification Toolbox) [24]. Flutter results for the AGARD 445.6 aeroelastic wing using the CFL3Dv6.4 code were presented as well, including computational costs. Unsteady aerodynamic state-space models were generated and coupled with a structural model within a MATLAB®/Simulink® environment for rapid calculation of aeroelastic responses, including the prediction of flutter. Aeroelastic responses computed directly using the CFL3Dv6.4 code showed excellent comparison with the aeroelastic responses computed using the CFD-based ROM.

Previously, the aerodynamic impulse responses that were used to generate the aerodynamic state-space models were computed using CFL3Dv6.0 via the excitation of one mode at a time. For a four-mode system, these computations are not very expensive. However, for more realistic cases in which the number of modes can be an order of magnitude or more larger, the one-mode-at-a-time method becomes impractical. Toward the solution of this problem, Kim [25] and Kim et al. [10] developed methods that enable the excitation of all the structural modes via a single CFD solution, greatly reducing the cost of identifying the aerodynamic impulse responses from the CFD code. The Kim et al. [10] method consists of using staggered step inputs, one per mode, and then applying several techniques for recovering the individual responses from this set of excitations, leading to the creation of state-space models.

This paper introduces two new types of excitations that can be used to simultaneously excite the structural modes while enabling the recovery of the individual input/output responses. This will enable the computation of aerodynamic impulse responses for any number of structural modes using a single CFD execution. Reduced-order models generated using the one-mode-at-a-time method are compared with ROMs using the simultaneous excitation inputs. Recent additional enhancements to the development of unsteady aerodynamic and aeroelastic ROMs, which includes the use of ROMs for static aeroelastic responses at matched-point atmospheric conditions, are presented in a separate paper [26].

Description of CFD and System Identification Methods

The following subsections describe the parallelized aeroelastic version of the CFL3Dv6.4 code, the phases of the original and improved ROM development processes, and a description of the functions used for simultaneous excitation of the CFD unsteady aerodynamic system.

CFL3Dv6.4 Code

The computer code used in this study is the CFL3Dv6.4 code, which solves the three-dimensional, thin-layer, Reynolds-averaged Navier–Stokes equations with an upwind finite volume formulation [27–29]. 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 and 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 CFL3Dv6.4 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 protocol can be used 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 message-passing-interface parallel aeroelastic computations for a semispan configuration are performed using 28 flowfield blocks. To achieve an optimal division of grid points, it is necessary to place flowfield 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 flowfield 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.

System Identification Method

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 ERA [23] and the observer Kalman identification [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) [24] developed at NASA Langley Research Center.

For the present paper, there are several algorithms within the SOCIT that are used for the development of unsteady aerodynamic discrete-time state-space models. The pulse algorithm is used to extract individual input/output impulse responses from simultaneous input/output responses. For a four-input/four-output system, simultaneous excitation of all four inputs [32] yields four-output responses. The pulse algorithm is used to extract the individual 16 (4 times 4) impulse responses that associate the response in one of the outputs due to one of the inputs. Details of the pulse algorithm are provided in the references. Once the individual 16 impulse responses are available, they are then processed via the ERA to transform the 16 individual impulse responses into a four-input/four-output, discrete-time, state-space model. A brief summary of the basis of this system identification algorithm follows.

A finite dimensional, discrete-time, linear, time-invariant dynamic 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 is an m-dimensional control input, y is a p-dimensional output or measurement vector, and k is 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, and C) such that the output responses of a given system due to a particular set of inputs is reproduced by the preceding discrete-time state-space system.

For the system of equations (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

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

where B is an $n \times m$ matrix and C is a $p \times n$ 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 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.

Original ROM Development Processes

A CFD-based aeroelastic system can be viewed as the coupling of a nonlinear unsteady aerodynamic system (flow solver) with a structural system. The present study focuses on the development of a linearized unsteady aerodynamic ROM (in state-space form) that is then coupled to a structural model (also in state-space form) for aeroelastic analyses. For the discussions that follow, the term ROM will refer to the unsteady aerodynamic state-space model. When the unsteady aerodynamic state-space model (ROM) is connected to a state-space model of the structure within the Simulink environment, this system is often also referred to as a ROM. However, to avoid confusion, the Simulink aeroelastic system will be referred to as the aeroelastic ROM.

An outline of the original (one-mode-at-a-time) ROM development process [8] is presented as background for the new enhancements. The original ROM development process is as follows:

1) The impulse/step response technique is implemented into aeroelastic CFD code.

2) Impulse/step responses are computed for each mode, one mode at a time, of an aeroelastic system using the aeroelastic CFD code; these responses are computed about a static aeroelastic solution (or a given dynamic pressure).

3) Impulse responses generated in step 2 are transformed into an unsteady aerodynamic state-space system using the ERA (within SOCIT).

4) The state-space models generated in step 3 are evaluated/validated via comparison with CFD results (i.e., aeroelastic ROM results vs full aeroelastic CFD solution results).

In the original ROM process, because each mode is being excited individually, the response in each output due to a particular input (e.g., the impulse response in output 2 due to input 1) is generated almost directly. If the function being used to excite the system is an impulse (or a unit pulse for discrete-time systems), then the output from the CFD solution will consist of the impulse response of each output due to that one input. If the function being used to excite the system is not an impulse function (such as a step input or a random input), then the impulse response needs to be extracted from the input/output data. One method for extraction is deconvolution [8], but there are other methods that can be used. For the original ROM process, because each input to a system is being excited individually, the exact nature of the input function is defined, to a certain extent, based on user preference (frequency range of interest, ease of implementation into a CFD solver, etc.). The step input has emerged as a convenient input for the original ROM process due to its ease of implementation into a CFD solver and the wide range of frequency excitation that it can generate.

The primary issue, then, with the original ROM process was the identification of the unsteady aerodynamic impulse responses one mode at a time (step 2). Clearly, for a large number of modes, this procedure becomes impractical.

Improved ROM Development Processes

An outline of the improved simultaneous modal excitation ROM development process with the recent enhancements is as follows:

1) Generate the number of functions (from a selected family) that corresponds to the number of structural mode shapes.

2) Apply the generated input functions simultaneously via one CFD execution; these responses are computed directly from the restart of a steady rigid CFL3Dv6.4 solution (not about a particular dynamic pressure).

3) Using the simultaneous input/output responses, identify the individual impulse responses using the pulse algorithm (within SOCIT);

4) Transform the individual impulse responses generated in step 3 into an unsteady aerodynamic state-space system using the ERA (within SOCIT).

5) Evaluate/validate the state-space models generated in step 4 via comparison with CFD results (i.e., aeroelastic ROM results vs full aeroelastic CFD solution results);

An important difference between the original ROM process and the improved ROM process is stated in steps 2 of the preceding outlines. For the original ROM process, if a static aeroelastic condition existed, then a ROM was generated about a selected static aeroelastic condition. So a static aeroelastic condition of interest was defined (typically a dynamic pressure) and that static aeroelastic condition was computed using CFL3Dv6.4 as a restart from a converged steady rigid solution. Once a converged static aeroelastic solution was obtained, the ROM process was applied about that condition. This implies that the resultant ROM is, of course, limited in some sense to the neighborhood of that static aeroelastic condition. Solutions far removed from that condition could result in loss of accuracy.

The reason for generating ROMs in this fashion was because no method had been defined to enable the computation of a static aeroelastic solution using a ROM. Any ROMs generated in this fashion were therefore limited to the prediction of dynamic responses about a static aeroelastic solution, including the methods by Raveh [11] and by Kim et al. [10]. The improved ROM method, however, includes a method for generating a ROM directly from a steady rigid solution. As a result, these improved ROMs can then be used to predict both static aeroelastic and dynamic solutions for any dynamic pressure. To capture a specific range of aeroelastic effects (previously obtained by selecting a particular dynamic pressure), the improved ROM method relies on the excitation amplitude to excite aeroelastic effects of interest. The details of the method for using a ROM for computing both static aeroelastic and dynamic solutions is presented in another reference by the present author [26]. For the present results, all responses were computed from the restart of a steady rigid CFL3Dv6.4 solution, bypassing the need (and additional computational expense) to execute a static aeroelastic solution using CFL3Dv6.4.

In the situation in which the goal is the simultaneous excitation of a multiple-input/multiple-output (MIMO) system, system identification techniques [33–35] dictate that the nature of the input functions used to excite the system must be properly defined if accurate input/output models of the system are to be generated. The most important point to keep in mind when defining these input functions is that these functions need to be different, in some sense, from each other. This makes sense because if the excitation inputs are identical and they are applied simultaneously, it becomes practically impossible for any system identification algorithm to isolate the effects of one input on a given output. This, in turn, makes it practically impossible for that algorithm to extract the individual impulse responses for each input/output pair. As has already been well established, the individual impulse responses for each input/output pair are necessary ingredients toward the development of state-space models. With respect to unsteady aerodynamic MIMO systems, these individual impulse responses correspond to time-domain GAFs, critical to understanding unsteady aerodynamic behavior. The Fourier-transformed version of these GAFs are the frequency-domain GAFs that provide an important link to more traditional frequency-domain-based aeroelastic analyses.

The question is how different these input functions should be and how we can quantify a level of difference between each input function. Kim et al. [10] used the familiar step function as the input

function, but with each step input being applied at a different point in time (lagged) to maintain some difference between the input signals. This family of lagged-step functions is not orthogonal. The greater the lag between successive step inputs, the greater the difference between input signals. This then presents a spectrum of possibilities in which, at one end, one has step input functions with no lags (identical functions, not orthogonal), and at the other end, the step functions are separated by an extremely large lag (in the limit, these are the individual step inputs from the original ROM process). Because orthogonality (linear independence) is the most precise mathematical method for guaranteeing the difference between signals, the present research focuses on the application of families of orthogonal functions as candidate input functions.

The three families of functions to be applied toward the efficient identification of a CFD-based unsteady aerodynamic state-space model are lagged-step, block-pulse, and Walsh. Although the lagged-step functions (not shown) are not orthogonal, these functions are used in the present study for comparison purposes. The other two functions each represent an orthogonal family of functions and are presented in Figs. 1 and 2. The block-pulse functions (Fig. 1) can be considered to be modified step inputs. An advantage of steplike functions is the broad frequency bandwidth that is excited by the impulsive nature of these functions. An added benefit of the block-pulse functions over traditional step functions is that block-pulse inputs provide excitation to the system in both positive and negative directions. From an unsteady aerodynamics point of view, it is important that modes of realistic wing configurations (e.g., with a nonsymmetric airfoil) be excited in both directions for a more complete capture of relevant dynamics.

The second set of orthogonal functions are the Walsh functions (Fig. 2). Likewise, this family of functions has a similarity to step inputs and therefore embodies the impulsive (i.e., beneficial) nature of step inputs with regard to frequency bandwidth.

In addition to the block-pulse and Walsh functions, a third family of orthogonal functions, the Haar functions (not shown), were also investigated. The Haar functions comprise one of the simplest form of wavelet functions [36]. Haar functions are generated using a power-of-2 algorithm. These functions yielded excellent results when applied toward the identification of a CFD-based unsteady aerodynamic state-space model using a single CFL3Dv6.4 execution for a multimode configuration. However, due to the fact that these functions are generated using a power-of-2 algorithm, the number of modes for the problem drives the size and, subsequently, the record length of the Haar input functions that are then input to CFL3Dv6.4. For example, if ten modes are to be excited simultaneously, then the record length of the ten Haar functions generated (one per mode) is

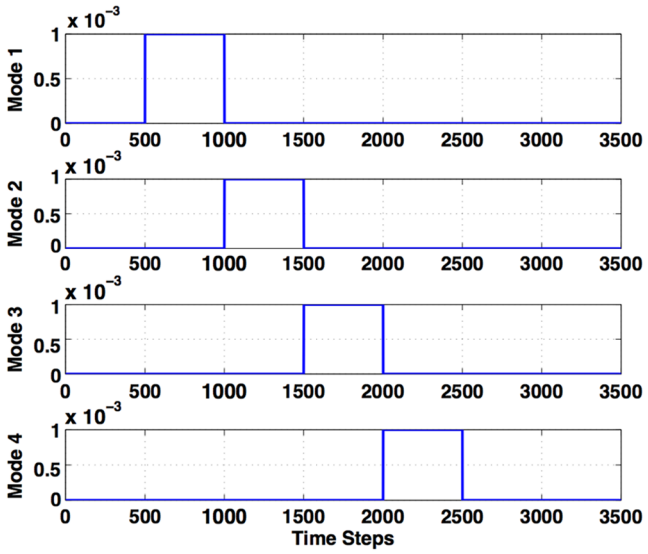

Fig. 1 Orthogonal block-pulse functions used as input for modal excitation.

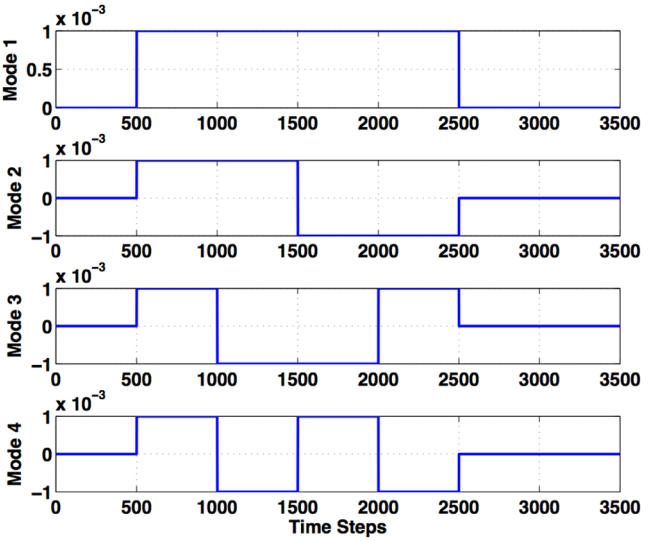

Fig. 2 Orthogonal Walsh functions used as input for modal excitation.

defined by the record length of the Haar function that corresponds to mode 10. Then taking 2 to the power of 10 yields 1024 time steps. Twelve modes yields a record length for the Haar functions of 4096 time steps, and 20 modes yields 1,048,576 time steps. Clearly, for a large number of modes, this family of functions becomes impractical. Therefore, only results for the lagged-step (not orthogonal), the block-pulse (orthogonal), and the Walsh (orthogonal) functions are presented.

Semispan CFD Model

The configuration used for the present analyses is a semispan wind-tunnel model known as the rigid semispan model (RSM). This configuration has been tested several times at NASA Langley's Transonic Dynamics Tunnel. The actual wind-tunnel model was fabricated using graphite and is very rigid. However, as part of a collaborative effort between the NASA Langley Research Center and the Boeing Company, a softened computational model of the RSM was developed by Hong [37]. The model was softened by simply reducing the first four modal frequencies by a factor of 4 and limiting the aeroelastic analysis to these first four modes. This four-mode softened version of the RSM was then used as a simple aeroelastic model for validation of ROM techniques. Figure 3 presents the surface grid for the CFL3Dv6.4 RSM configuration. Figure 4 presents the four softened mode shapes for the CFL3Dv6.4 RSM configuration.

The results presented are for Euler (inviscid) solutions at a Mach number of 0.7 and an angle of attack of 3 deg. This configuration does not have a symmetric airfoil and will therefore generate a static aeroelastic response in addition to the dynamic aeroelastic response. The method for using an unsteady aerodynamic state-space ROM for

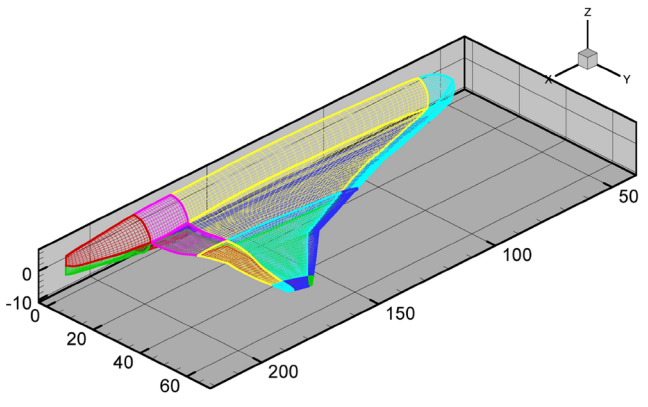

Fig. 3 Computational grid of semispan configuration.

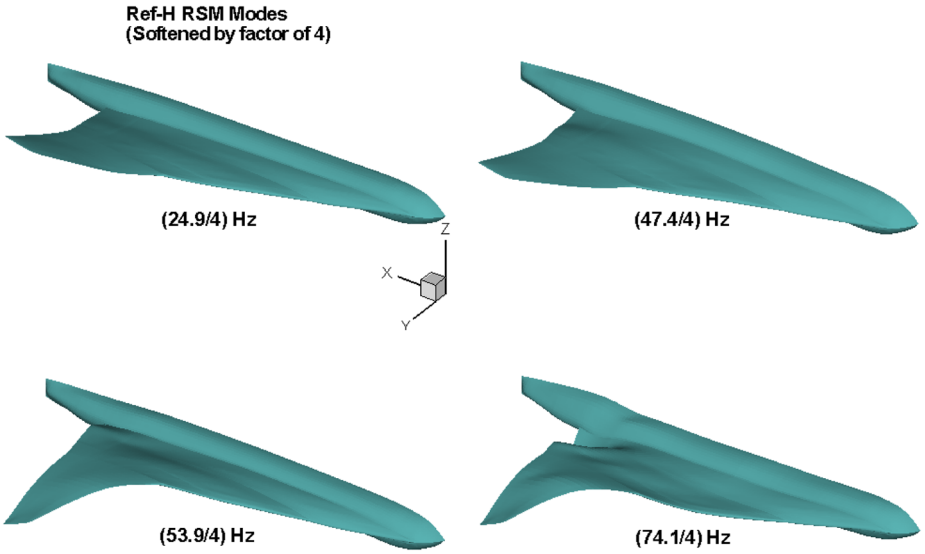

Fig. 4 Softened mode shapes of semispan configuration.

static aeroelastic responses is not presented in this paper, but is presented in another paper by this author [26].

Results

In this section, results from the lagged-step, block-pulse, and Walsh [38] input functions are compared with results from the one-mode-at-a-time approach (serial responses). Figures 1 and 2 present the block-pulse and Walsh functions, respectively, that were input to the CFL3Dv6.4 RSM model for simultaneous excitation of the four modes. Although not shown, four lagged-step inputs, one per mode, with a lag of 50 time steps between each step input, were applied as well.

All CFL3Dv6.4 solutions were 3500 time steps long, including 500 initial time steps with zero input values to prevent any startup transients from corrupting the response data. Therefore, the first lagged-step input (applied to the first mode) was applied at time step 501, the second lagged-step input (applied to the second mode) was applied at time step 551, and so on for the remaining two modes. As shown in Fig. 1, each block pulse was 500 time steps long (2000 steps total for all four block pulses), and an additional 1000 time steps were added to capture free-decay transients. As shown in Fig. 2, the Walsh inputs were applied over a 2000-time-step range and an additional 1000 time steps were added here as well. For the present study, these time-step variations were selected based on maintaining an equal number of time steps (3500 time steps) for all solutions (serial and simultaneous) and there was no attempt to optimize these time-step variations for specific input functions.

There are four one-mode-at-a-time (serial) CFL3Dv6.4 solutions, one per mode. The first serial CFL3Dv6.4 solution provides the generalized coordinate responses for modes 1–4 due to a step input in mode 1. The second serial CFL3Dv6.4 solution provides the generalized coordinate responses for modes 1–4 due to a step input in mode 2, and so on for the remaining modes. These 16 generalized coordinate responses are used to compute the desired individual impulse responses for each input/output pair. Clearly, for the serial solutions, the pulse algorithm does not need to be applied. Each one of these serial solutions was computed for 3500 time steps, costing 1.33 h each using 14 nodes from a 138-node Dell 3.4-Ghz dual-core Xeons cluster, for a total cost of 5.32 h.

For the other three input functions (50-step lagged-step, block-pulse, and Walsh), because the inputs are applied simultaneously, there is only one CFL3Dv6.4 solution per input function. The computational cost for each one of these CFL3Dv6.4 solutions was 1.33 h on the same cluster. The output from these CFL3Dv6.4 solutions consists of the generalized coordinate response in all four modes due to an input (lagged-step, block-pulse, or Walsh) in all four modes. For these CFL3Dv6.4 solutions, the pulse algorithm is necessary to extract the 16 impulse responses from the four-input/four-output sets of data. The MATLAB-based pulse algorithm computes these 16 GAF impulse responses within a few minutes on a desktop computer and is of minimal computational cost when compared with the full CFD solutions.

Processing the outputs for the four generalized coordinates for each set of input functions through the pulse algorithm, the 16 time-domain GAF impulse responses were identified. A more direct comparison of these GAF impulse responses can be obtained via the analysis of the frequency-domain version of these GAFs. The frequency-domain GAFs due to the three input functions (50-step lagged steps, block-pulse, and Walsh) can then be compared with the frequency-domain GAFs due to the one-mode-at-a-time (serial) step inputs to determine the effectiveness and accuracy of the simultaneous excitations.

Application of an FFT to each one of the 16 time-domain GAF impulse responses directly yields the desired frequency-domain GAFs [8]. For the sake of brevity, a subset of the 16 frequency-domain GAFs will be presented and compared. The frequency-domain GAFs for the first, second, and third modes due to the simultaneous input functions are compared with the same GAFs due to the serial inputs. The diagonal (1, 1), (2, 2), and (3, 3) GAF components and the offdiagonal (2, 1), (2, 3), and (3, 2) GAF components are presented as a function of real-vs-imaginary values for the four sets of input functions (serial, lagged-step, block-pulse, and Walsh) ranging in reduced frequency from 0 (at an imaginary value of zero) to 2.0.

Presented in Fig. 5 are the frequency-domain (1, 1)-component GAFs for the serial step, 50-step lagged-step, block-pulse, and Walsh input functions. The (1, 1)-component GAF for the three simultaneous input functions compares very well with the (1, 1)-component GAF for the serial inputs. For this case, the (1, 1)-component GAF due to the block-pulse input function exhibits the best correlation with the (1, 1)-component GAF due to the serial inputs. Overall, however, it appears that all three simultaneous inputs perform well at capturing this GAF component.

Presented in Fig. 6 are the frequency-domain (2, 1)-component GAFs for the serial step, 50-step lagged-step, block-pulse, and Walsh input functions. The (2, 1)-component GAF due to the lagged-step input function differs significantly from the (2, 1)-component GAF due to the serial inputs, especially around the zero-to-low reduced frequencies. The (2, 1)-component GAFs due to the orthogonal input functions (block-pulse and Walsh) capture the overall trend when compared with the (2, 1)-component GAF due to the serial inputs. As can be seen, the (2, 1)-component GAF due to the Walsh input function exhibits the best correlation for this case.

Presented in Fig. 7 are the frequency-domain (2, 2)-component GAFs for the serial step, 50-step lagged-step, block-pulse, and Walsh

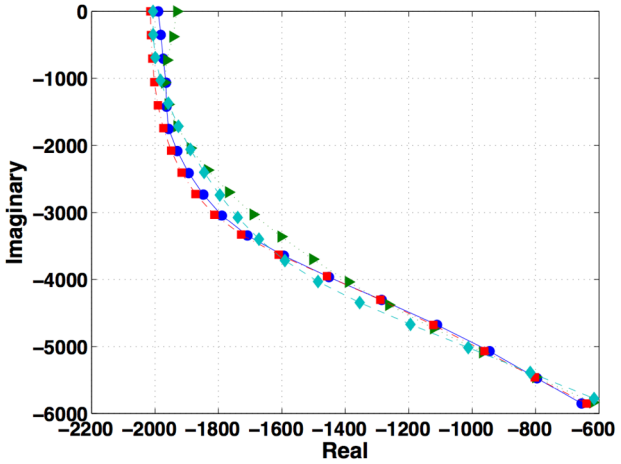

Fig. 5 GAF 1, 1 for the serial step inputs (circles), 50-step lagged-step (triangles), block-pulse (squares), and Walsh (diamonds) input functions.

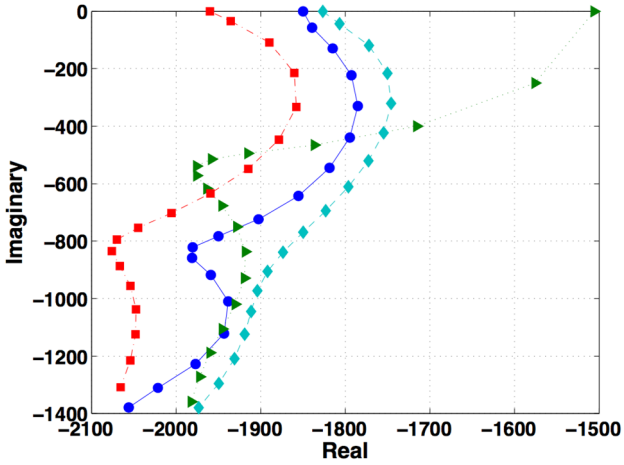

Fig. 6 GAF 2, 1 for the serial step inputs (circles), 50-step lagged-step (triangles), block-pulse (squares), and Walsh (diamonds) input functions.

input functions. The (2, 2)-component GAF for the three simultaneous input functions compares well with the (2, 2)-component GAF for the serial inputs. Closer inspection of Fig. 7 indicates that the steady-to-low reduced-frequency range for this GAF due to the Walsh input functions are better correlated with the same range for the GAF due to the serial inputs than are the GAFs due to the other two input functions (block-pulse and lagged-step). Once again, for this diagonal component GAF, all three simultaneous input functions perform well when compared with the same component GAF due to the serial inputs.

Figure 8 presents the frequency-domain (3, 3)-component GAFs for the serial step, 50-step lagged-step, block-pulse, and Walsh input functions. It can be seen that the (3, 3)-component GAF due to the block-pulse input function correlates best with the (3, 3)-component GAF due to the serial step inputs over most of the reduced-frequency range. At steady conditions (zero reduced frequency), the (3, 3)-component GAF due to the Walsh input function once again compares the best with the (3, 3)-component GAF due to the serial step inputs.

Figure 9 presents the frequency-domain (2, 3)-component GAFs for the serial step, 50-step lagged-step, block-pulse, and Walsh input functions. For this case, the (2, 3)-component GAF due to the Walsh input function correlates best with the (2, 3)-component GAF due to the serial step inputs over most of the reduced-frequency range. At steady conditions (zero reduced frequency), the (2, 3)-component GAF due to the Walsh input function compares the best with the (2, 3)-component GAF due to the serial step inputs.

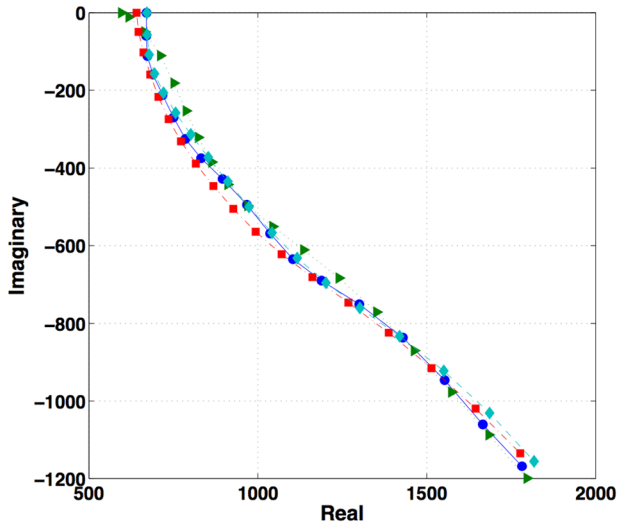

Fig. 7 GAF 2, 2 for the serial step inputs (circles), 50-step lagged-step (triangles), block-pulse (squares), and Walsh (diamonds) input functions.

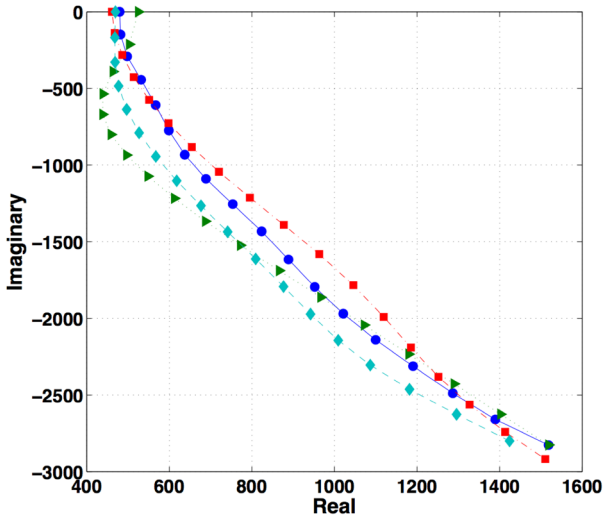

Fig. 8 GAF 3, 3 for the serial step inputs (circles), 50-step lagged-step (triangles), block-pulse (squares), and Walsh (diamonds) input functions.

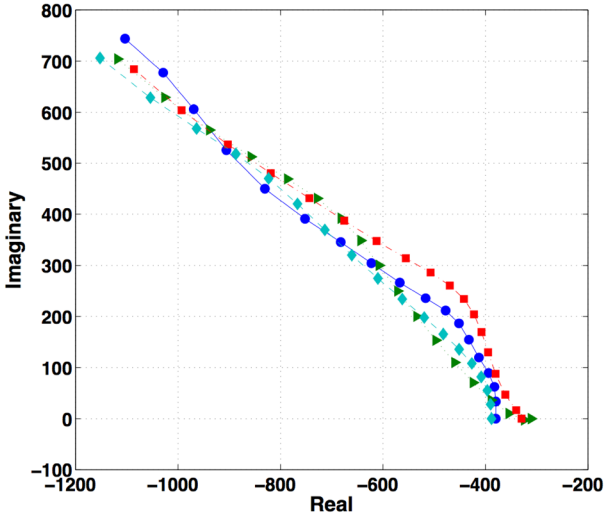

Fig. 9 GAF 2, 3 for the serial step inputs (circles), 50-step lagged-step (triangles), block-pulse (squares), and Walsh (diamonds) input functions.

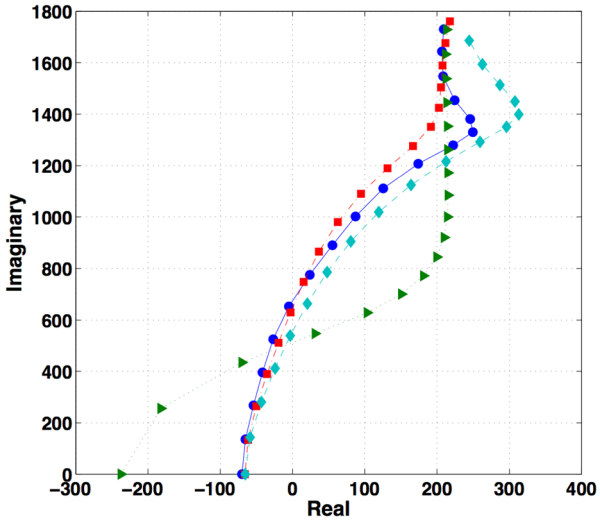

Fig. 10 GAF 3, 2 for the serial step inputs (circles), 50-step lagged-step (triangles), block-pulse (squares), and Walsh (diamonds) input functions.

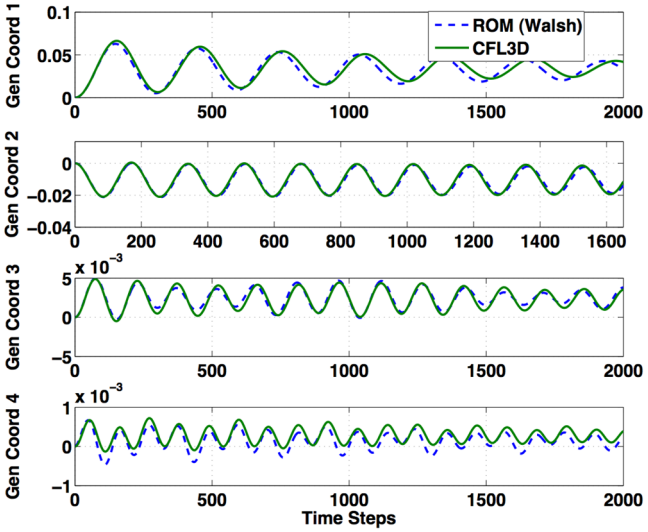

Fig. 11 Full solution (static plus dynamic aeroelastic responses) generalized coordinate responses at a Mach number of 0.7 and a dynamic pressure of 0.1 psi.

Figure 10 presents the frequency-domain (3, 2)-component GAFs for the serial step, 50-step lagged-step, block-pulse, and Walsh input functions. The (3, 2)-component GAFs due to the block-pulse and Walsh input functions compare very well with the (3, 2)-component GAF due to the serial step inputs over most of the reduced-frequency range. The (3, 2)-component GAF due to the lagged-step exhibits large variations throughout most of the reduced-frequency range, improving toward the higher range of reduced frequencies. At steady conditions (zero reduced frequency), the (3, 2)-component GAF due to the block-pulse and the Walsh input functions compare very well with the (3, 2)-component GAF due to the serial step inputs.

For the results presented, the orthogonal functions (block-pulse and Walsh) demonstrate improved performance over the lagged-step functions in terms of comparison with the results obtained via the serial approach. In particular, for most of the results presented, the Walsh input function performed the best in terms of capturing the steady (zero reduced frequency) to low reduced-frequency range. The ability to accurately capture steady-to-low reduced-frequency effects with a minimal number of time steps is clearly an important metric toward identifying an optimal input function for ROM identification. Additional parametric studies are required to identify the optimal input function out of the candidate input functions presented. However, for the present results, it appears that the Walsh input function exhibited the best overall performance.

To generate a state-space model of the unsteady aerodynamic system, the 16 time-domain GAF impulse responses were processed through the ERA (part of SOCIT), which executes within a matter of minutes on a desktop computer. A state-space model of the structure was created using the modal information (modal frequencies and generalized masses) of the softened RSM configuration. The unsteady aerodynamic state-space model was then connected to the structural state-space model within the Simulink environment to create an aeroelastic system representative of the CFL3Dv6.4 aeroelastic model of the softened RSM configuration.

Presented in Fig. 11 is a comparison of aeroelastic responses in the four generalized coordinates for the CFL3Dv6.4 solution and the aeroelastic ROM solution at a Mach number of 0.7 and a dynamic pressure of 0.1 psi. The unsteady aerodynamic state-space model used in this aeroelastic ROM was the state-space model developed from the GAFs generated using the Walsh input functions. The CFL3Dv6.4 responses include both static and dynamic aeroelastic responses simultaneously. Although it is well known that accurate static aeroelastic solutions require a large number of modes (certainly larger than 4), the goal of the present comparison is to determine the accuracy of the unsteady aerodynamic state-space ROM and the aeroelastic ROM to reproduce the CFL3Dv6.4 results for a given configuration. Multiplication of each generalized coordinate with its respective mode shape and summing of these products yields the physical deformation of the configuration as a function of space and time.

Typically, static and dynamic aeroelastic responses are computed separately to reduce computational cost. A static aeroelastic solution is computed as the restart solution from a steady rigid solution using an artificially high structural damping value to accelerate convergence to a static aeroelastic solution. Then, a dynamic aeroelastic solution is computed as the restart solution from the converged static aeroelastic solution. In the paper by the present author [26], a method is introduced that enables the prediction of static and dynamic aeroelastic solutions from the same aeroelastic ROM. For the present paper, the sample results presented in Fig. 11 are for a full solution: that is, a solution that includes both static and dynamic responses for a given dynamic pressure. Notice that the solutions start at a generalized coordinate value of zero (static, rigid solution) and converge to a nonzero mean value (static aeroelastic solution), with the dynamic solution superimposed on the static solution. As can be seen, the comparison is very good and the aeroelastic ROM captures the combined static and dynamic aeroelastic responses predicted by the CFL3Dv6.4 solution.

Conclusions

Previous methods for identifying an unsteady aerodynamic state-space reduced-order model (ROM) from an aeroelastic CFD model consisted of one-mode-at-a-time (serial) inputs, resulting in as many CFD executions as there are modes. An improved method for generating these ROMs was introduced that enables the simultaneous excitation of an aeroelastic CFD model, consisting of any number of modes, with a single CFD execution. This is achieved by the introduction of orthogonal functions, such as block-pulse and Walsh functions, as the input excitations to the aeroelastic CFD system. The aeroelastic CFD system consisted of a CFL3Dv6.4 model of a semispan configuration with four structural modes. Frequency-domain generalized aerodynamic forces (GAFs) due to these orthogonal functions were compared with frequency-domain GAFs due to serial inputs. In addition, frequency-domain GAFs due to the simultaneous application of four nonorthogonal lagged-step inputs (one per mode) were presented as well. The results indicated that for the diagonal components of the GAFs presented, all three simultaneous inputs (lagged-step, block-pulse, and Walsh) were suitable for capturing relevant unsteady aerodynamic effects when compared with the GAFS due to serial inputs. For the offdiagonal components of the GAFs, the orthogonal functions (in particular, the Walsh functions) demonstrated improved performance, in particular, in the steady (zero reduced frequency) to low reduced-frequency

range. For the present four-mode CFD model, the use of simultaneous (and, in particular, orthogonal) input functions resulted in a factor-of-4 savings in computational expense because the serial input approach required four CFD executions, whereas the simultaneous input approach required only one CFD execution.

References

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

[2] Wagner, H., "Dynamische Auftrieb von Tragflugeln," *ZAMM*, Vol. 5, No. 1, 1925, pp. 17–35.

[3] Kussner, H. G., "Schwingungen von Flugzeugflugeln," *Luftfahrtforschung*, Vol. 4, June 1929, pp. 41–62.

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

[5] Sears, W. R., "Some Aspects of Non-Stationary Airfoil Theory and Its Practical Applications," *Journal of the Aeronautical Sciences*, Vol. 8, No. 3, 1941, pp. 104–108.

[6] Giesing, J. P., Kalman, T. P., and Rodden, W. P., "Subsonic Unsteady Aerodynamics for General Configurations, Part 1: Direct Application of the Nonplanar Doublet Lattice Method," U.S. Air Force Flight Dynamics Lab., Rept. TR-71-5, Vol. 1, Wright-Patterson AFB, OH, Nov. 1971.

[7] Silva, W. A., "Identification of Nonlinear Aeroelastic Systems Based on the Volterra Theory: Progress and Opportunities," *Nonlinear Dynamics*, Vol. 39, Jan. 2005, pp. 25–62.

[8] Silva, W. A., and Bartels, R. E., "Development of Reduced-Order Models for Aeroelastic Analysis and Flutter Prediction Using the CFL3Dv6.0 Code," *Journal of Fluids and Structures*, Vol. 19, No. 6, 2004, pp. 729–745.
doi:10.1016/j.jfluidstructs.2004.03.004

[9] Beran, P. S., and Silva, W. A., "Reduced-Order Modeling: New Approaches for Computational Physics," 39th Aerospace Sciences Meeting, Reno, NV, AIAA Paper 2001-0853, Jan. 2001.

[10] Kim, T., Hong, M., Bhatia, K. G., and SenGupta, G., "Aeroelastic Model Reduction for Affordable Computational Fluid Dynamics-Based Flutter Analysis," *AIAA Journal*, Vol. 43, No. 12, 2005, pp. 2487–2495.
doi:10.2514/1.11246

[11] Raveh, D. E., "Identification of Computational-Fluid-Dynamic Based Unsteady Aerodynamic Models for Aeroelastic Analysis," *Journal of Aircraft*, Vol. 41, No. 3, June 2004, pp. 620–632.
doi:10.2514/1.3149

[12] 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*, AIAA, Reston, VA, June 2001, pp. 159–174.

[13] 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, AIAA Paper 99-1262, June 1999.

[14] Silva, W. A., "Discrete-Time Linear and Nonlinear Aerodynamic Impulse Responses for Efficient CFD Analyses," Ph.D. Thesis, College of William & Mary, Williamsburg, VA, Dec. 1997.

[15] Silva, W. A., "Application of Nonlinear Systems Theory to Transonic Unsteady Aerodynamic Responses," *Journal of Aircraft*, Vol. 30, No. 5, 1993, pp. 660–668.

[16] 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, Atlanta, GA, AIAA, Paper 2000-1325, Apr. 2000.

[17] Roger, K. L., "Airplane Math Modeling Methods for Active Control Design," AGARD CP-228, AGARD, Neuilly-sur-Seine, France, Aug. 1977.

[18] 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.

[19] Gupta, K. K., Voelker, L. S., Bach, C., Doyle, T., and Hahn, E., "CFD-Based Aeroelastic Analysis of the X-43 Hypersonic Flight Vehicle," 39th Aerospace Sciences Meeting and Exhibit, Reno, CA, AIAA Paper 2001-0712, Jan. 2001.

[20] Cowan, T. J., A. S. A. Jr., and Gupta, K. K., "Accelerating CFD-Based Aeroelastic Predictions Using System Identification," 36th AIAA Atmospheric Flight Mechanics Conference and Exhibit, Boston, MA, AIAA Paper 1998-4152, Aug. 1998, pp. 85–93.

[21] Cowan, T. J., A. S. A. Jr., and Gupta, K. K., "Development of Discrete-Time Aerodynamic Model for CFD-Based Aeroelastic Analysis," 37th Aerospace Sciences Meeting and Exhibit, Reno, NV, AIAA Paper 1999-0765, Jan. 1999.

[22] Rodrigues, E. A., "Linear/Nonlinear Behavior of the Typical Section Immersed in Subsonic Flow," 42nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference and Exhibit, Seattle, WA, AIAA Paper 2001-1584, Apr. 2001.

[23] 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, No. 5, 1985, pp. 620–627.

[24] Juang, J.-N., *Applied System Identification*, Prentice–Hall, Upper Saddle River, NJ, 1994.

[25] Kim, T., "Efficient Reduced-Order System Identification for Linear Systems with Multiple Inputs," *AIAA Journal*, Vol. 43, 2005, pp. 1455–1464.
doi:10.2514/1.11225

[26] Silva, W. A., "Recent Enhancements to the Development of CFD-Based Aeroelastic Reduced Order Models," 48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Honolulu, HI, AIAA Paper 2007-2051, Apr. 2007.

[27] Krist, S. L., Biedron, R. T., and Rumsey, C. L., *CFL3D User's Manual Version 5.0*, NASA Langley Research Center, Hampton, VA, 1997.

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

[29] Bartels, R. E., Rumsey, C. L., and Biedron, R. T., "CFL3D Version 6.4: General Usage and Aeroelastic Analysis," NASA TM 2006 214301, Apr. 2006.

[30] Roe, P. L., "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes," *Journal of Computational Physics*, Vol. 43, 1981, pp. 357–372.
doi:10.1016/0021-9991(81)90128-5

[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, No. 2, 1993, pp. 320–329.

[32] Silva, W. A., "Simultaneous Excitation of Multiple-Input Multiple-Output CFD-Based Unsteady Aerodynamic Systems," 48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Honolulu, HI, AIAA Paper 2007-1988, Apr. 2007.

[33] Eykhoff, P., *System Identification: Parameter and State Identification*, Wiley, New York, 1974.

[34] Ljung, L., *System Identification: Theory for the User*, Prentice–Hall, Upper Saddle River, NJ, 1999.

[35] Zhu, Y., *Multivariable System Identification for Process Control*, Pergamon, New York, 2001.

[36] Strang, G., and Nguyen, T., *Wavelets and Filter Banks*, Wellesley–Cambridge Press, Wellesley, MA, 1996.

[37] Hong, M., Kuruvila, G., Bhatia, K., SenGupta, G., and Kim, T., "Evaluation of CFL3D for Unsteady Pressure and Flutter Predictions," 44th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Norfolk, VA, AIAA Paper 2003-1923, Apr. 2003.

[38] Pacheco, R. P., and V. Steffen, J., "Using Orthogonal Functions for Identification and Sensitivity Analysis of Mechanical Systems," *Journal of Vibration and Control*, Vol. 8, June 2002, pp. 993–1021.