

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

Walter A. Silva *

NASA Langley Research Center
Hampton, Virginia 23681-0001

Abstract

A significant improvement to the development of CFD-based unsteady aerodynamic reduced-order models (ROMs) is presented. This improvement involves the simultaneous excitation of the structural modes of the CFD-based unsteady aerodynamic system that enables the computation of the unsteady aerodynamic state-space model using a single CFD execution, independent of the number of structural modes. Four different types of inputs are presented that can be used for the simultaneous excitation of the structural modes. Results are presented for a flexible, supersonic semi-span 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 flow¹ include Wagner's function² (response to a unit step variation in angle of attack), Kussner's function³ (response to a sharp-edged gust), Theodorsen's function⁴ (frequency response to sinusoidal pitching motion), and Sear's function⁵ (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

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

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 (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 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 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 approximation¹⁷ and the Minimum State technique.¹⁸ 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 al¹⁹ 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 in order to excite a specific frequency range, resulting in an iterative process. In a similar vein, Rodrigues²² developed a state-space model for an airfoil in transonic flow using a transonic small-disturbance algorithm.

In the present 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).

Silva and Bartels⁸ 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 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). Flutter results for the AGARD 445.6 Aeroelastic Wing using the CFL3Dv6 code were presented as well, including computational costs. Unsteady aerodynamic state-space models were generated and coupled with a structural model within a MATLAB/SIMULINK²⁴ environment for rapid calculation of aeroelastic responses including the prediction of flutter. Aeroelastic responses computed directly using the CFL3Dv6 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 where the number of modes can be an order of magnitude or more larger, the one-mode-at-a-time method becomes impractical. Towards the solution of this problem, Kim²⁵ and Kim et al¹⁰ developed methods that enable the simultaneous excitation of the structural modes, greatly reducing the cost of identifying the aerodynamic impulse responses from the CFD code. Kim's method consists of using staggered step inputs, one per mode, and then applying several techniques for recovering the individual responses from this type of simultaneous excitation leading to the creation of state-space models.

This paper introduces four different types of excitations that can be used to simultaneously excite the structural modes while enabling the recovery of the individual 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 to 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 four types of functions used for simultaneous excitation of the CFD unsteady aerodynamic system.

CFL3Dv6.4 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.^{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 Roe³⁰ 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 a flexible, semi-span supersonic configuration are performed using 28 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.

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 Eigensystem Realization Algorithm (ERA)²³ 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.

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 inputs³³

yields four output responses. The PULSE algorithm is used to extract the individual sixteen (four times four) 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 sixteen impulse responses are available, they are then processed via the Eigen-system Realization Algorithm (ERA) in order to transform the sixteen individual impulse responses into a four-input/four-output, discrete-time, state-space model. 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

$$x(k+1) = Ax(k) + Bu(k) \quad (1)$$

$$y(k) = Cx(k) + Du(k) \quad (2)$$

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.

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

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

with B an $(n \times m)$ matrix and C 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 (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.

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 as depicted in Figure 1. The present study focuses on the development of a linearized unsteady aerodynamic ROM (in state-space form), using the general procedure depicted in Figure 2, 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 simulation ROM.

An outline of the original (one-mode-at-a-time) ROM development process⁸ is presented as background for the new enhancements. The original ROM development process is as follows:

1. Implementation of impulse/step response technique into aeroelastic CFD code;
2. Computation of impulse/step responses 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. Evaluation/validation of the state-space models generated in Step 3 via comparison with CFD results (i.e., ROM results vs. full CFD solution results);

In the original ROM process, since 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 deconvolution⁸ but there are other methods that can be used. For the original ROM process, since 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 to 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 modeshapes;

2. Apply the generated input functions simultaneously via one CFD execution; these responses are computed directly from the restart of a steady rigid CFL3D 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., ROM results vs. full CFD solution results);

An important difference between the original ROM process and the improved ROM process is stated in steps (2) of the outlines above. 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 CFL3D 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. Moving "too far away" 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 Raveh¹¹ 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. In order 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 CFL3D solution, bypassing the need (and additional computational expense) to execute a static aeroelastic solution using CFL3D.

Simultaneous Excitation Input Functions

In the situation where the goal is the simultaneous excitation of a multiple-input multiple-output (MIMO) system, system identification techniques^{34–36} 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 since, if the excitation inputs are identical and they are applied simultaneously, it becomes practically impossible for any system identification algorithm to relate 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 towards the development of state-space models. With respect to unsteady aerodynamic MIMO systems, these individual impulse responses correspond to time-domain generalized aerodynamic forces (GAFs), critical to understanding unsteady aerodynamic behavior. The Fourier-transformed version of these GAFs are the frequency-domain GAFs which provide an important link to more traditional frequency-domain-based aeroelastic analyses.

The question is how different should these input functions be and how can we quantify a level of "difference" between each input function? Kim¹⁰ uses the familiar step function as the input function but with each step input being applied at a different point in time (lagged) in order to maintain some difference between the input signals. Kim refers to this family of lagged step functions as "almost orthogonal". As Kim points out, the greater the lag between successive step inputs, the greater the difference between input signals. This then presents a spectrum of possibilities where, 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). Since 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. Using orthogonal functions directly provides a mathematical guarantee that the input functions are as different as mathematically possible. These orthogonal input functions then can be considered optimal input functions for the identification of a MIMO system.

The four families of functions to be applied towards the efficient identification of a CFD-based unsteady aerodynamic state-space model are: lagged step, block pulse, Haar, and Walsh. Although it has been stated by Kim that the lagged-step functions are not completely orthogonal (almost orthogonal), these functions are used in the present study for comparison

purposes. The other three functions each represent an orthogonal family of functions. The block pulse functions, presented in Figure 3, are orthogonal and resemble a modified step input. An advantage of step-like 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 non-symmetric airfoil) be excited in both directions for a more complete capture of relevant dynamics.

The Haar functions, presented in Figure 4, comprise one of the simplest form of wavelet functions. These functions are orthogonal by the mathematical definition of wavelet functions. Likewise, this family of functions has a similarity to step inputs and therefore embodies the impulsive (i.e., beneficial) nature of step inputs with regards to frequency bandwidth. The final set of orthogonal functions is the Walsh function, presented in Figure 5.

It is important to mention that the Haar functions are generated using formulas based on a power of two algorithm. These functions have yielded excellent results when applied towards the identification of a CFD-based unsteady aerodynamic state-space model using a single CFL3D execution for a multi-mode configuration. However, due to the fact that these functions are generated based on a power of two algorithm, the number of modes drives the size and, subsequently, the record length of the input functions that are then input to CFL3D. As a result, this can severely limit the number of modes that can be excited at one time. For example, if ten modes are to be excited simultaneously, then the record length of the ten Haar functions generated (one per mode) is defined by the record length of the Haar function that corresponds to mode ten. Then taking two to the power of ten yields 1024 time steps. Twelve modes yields a record length for the Haar functions of 4096 time steps, thirteen modes yields 8192 time steps, fourteen modes yields 16384 time steps, and so on. Clearly, for a very large number of modes, this family of functions is not practical. Therefore, only those results for the lagged step (not orthogonal), the block pulse (orthogonal), and the Walsh (orthogonal) functions are presented.

Supersonic Semi-Span CFD Model

The configuration used for the present analyses is a supersonic semi-span wind-tunnel model known as the Rigid Semi-span Model (RSM). This configuration has been tested several times at the NASA Langley's Transonic Dynamics Tunnel (TDT). The actual wind-tunnel model was fabricated using graphite and is very rigid. However, as part of collaborative studies between the NASA Langley Research Center and the Boeing Company (Seattle), a "softened" computational model of the RSM was developed by Dr. Moeljo Hong (Boe-

ing).³⁷ The model was "softened" by simply reducing the four modal frequencies by a factor of four.

The results presented are for Euler (inviscid) solutions at a Mach number of 0.7 and an angle of attack of 3 degrees. 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 static aeroelastic responses is not presented in this paper but is presented in another paper by this author.²⁶ Figure 6 presents the surface grid for the CFL3D RSM configuration.

Results

As previously stated, when using the lagged-step functions as excitation inputs, it is important to consider the lag (number of time steps) between each successive step input. It is the time lag between each step input that determines the quality of the resultant model that is identified. A time lag of fifteen (15) time steps was used initially in order to explore the effect of the time lag on the resultant ROM. Presented in Figure 7 is the CFL3D response in the first generalized coordinate due to the excitation of the four lagged-step inputs. The effect of each of the lagged steps is evident in the response of the generalized coordinate as evidenced by the four pulses in the response.

Presented in Figure 8 is the CFL3D response in the first generalized coordinate due to the excitation of four lagged-step inputs but this time with a relative time lag of 50 time steps. The effect of each of the lagged steps is evident in the response of the generalized coordinate as evidenced by the four pulses in the response.

Figure 9 presents the four block pulse functions input to the CFL3D RSM model for simultaneous excitation of the four modes.

Presented in Figure 10 is the CFL3D response in the first generalized coordinate due to the excitation of the block pulse inputs. The effect of each of the block pulse inputs is evident in the response of the generalized coordinate as evidenced by the four pulses in the response. Note that the block pulse functions were defined over an interval of 2000 time steps. The lagged step inputs were defined depending on the desired lag between each step input but all solutions (lagged step, block pulse, and Walsh) were executed for the same number of time steps.

In Figure 11, Figure 12, Figure 13, and Figure 14 each of the four Walsh functions is presented separately for clarity. The CFL3D response in the first generalized coordinate due to the Walsh function inputs is presented in Figure 15. Like the block pulse functions, the Walsh functions were generated over a record length of 2000 time steps.

Processing the outputs for all four generalized coordinates for each input function through

the PULSE algorithm in SOCIT, the time-domain generalized aerodynamic forces (GAFs) can be identified. The first twenty-five time steps of the time-domain GAF in mode 1 due to an input in mode 1 (GAF 1,1) for the four input functions (lagged step (15 steps), lagged step (50 steps), block pulse, and Walsh) is presented as Figure 16. Clearly, if the focus is on the initial portion of the GAF (high frequency), then all four input functions yield identical GAFs. The results for the other GAFs (fifteen additional GAFs) is similar to the one shown in Figure 16.

However, an analysis of the time-domain GAF 1,1 from the 20th to the 200th time step, presented in Figure 17, reveals noticeable differences due to the four input functions. Note that the vertical scale has been greatly expanded from that of Figure 16. This variance can be considered an indication of the difficulty that the system identification algorithm is encountering in attempting to identify the system. It can be seen that the greatest variance is for the lagged step input with 15 time step delays, compared to the others. It is interesting to note the similarity between the lagged step (50 time steps) and the block pulse input functions.

Additional comparison of these results can be obtained via the analysis of the frequency-domain version of the GAFs. The frequency-domain GAFs from each of four input functions are compared to the frequency-domain GAFs from the use of serial (one-at-a-time) step inputs. Comparison of the simultaneous input responses with those from the one-mode-at-a-time does not imply that the one-mode-at-a-time results are the correct results. That is, it is quite possible that (and most probable) that the responses obtained from the simultaneous excitation will be different from the responses obtained from the one-mode-at-a-time excitations due to potential nonlinear modal coupling effects. The reason for comparing these two sets of responses is simply due to the fact that the one-mode-at-a-time responses represent the original ROM approach. Application of an FFT to each one of the time-domain GAFs directly yields the desired frequency-domain GAFs.⁸ Only a subset of the sixteen GAFs will be presented in the figures below. The first comparison is for the real component of GAF 1,1, due to the lagged step input with 15 time-step delays, presented in Figure 18. The effect of the variance exhibited in the time domain is manifested as a difference at low reduced frequencies.

The imaginary component of GAF 1,1 due to the lagged step with 15 time-step delays is presented in Figure 19. The comparison for the imaginary component is quite good, with slight variations at certain frequency ranges.

The real and imaginary components of GAF 1,1 due to the lagged step with 50 time-step delays are presented in Figure 20 and Figure 21. Here again, some variations from the serial inputs are noticed as well, in particular with respect to the steady (zero reduced frequency)

value.

The real and imaginary components of GAF 1,1 due to the block pulse function inputs are presented as Figure 22 and Figure 23. For both the real and imaginary components, noticeable improvement in the comparison is achieved with the use of the block pulse function.

The real and imaginary components of GAF 1,1 due to the Walsh function inputs are presented as Figure 24 and Figure 25. For both the real and imaginary components, once again, noticeable improvement in the comparison is achieved with the use of the Walsh function. In particular, for both the block pulse and the Walsh functions, the steady value (value at zero reduced frequency) is well captured.

A comparison of the frequency-domain GAF 2,2 is presented for the lagged step (50 time-step delays) and the Walsh function inputs. Figure 26 and Figure 27 contain the comparison of the real and imaginary components of GAF 2,2 for the lagged step (50 time-step delays) and the serial (step) inputs. Some differences are noticed, in particular in the low reduced frequency range for both the real and imaginary components.

Figure 28 and Figure 29 are the comparison of the real and imaginary components of GAF 2,2 for the Walsh and the serial (step) inputs. An improvement in the comparison is once again noted.

The time-domain GAFs are then processed through the ERA (part of SOCIT) in order to generate a state-space model of the unsteady aerodynamic system. The resultant state-space models (for the lagged step with 50 time-step delays and the Walsh functions) were 12th order in dimension. To demonstrate the effectiveness of the new input functions, a representative result is presented as Figure 30, a comparison of responses in the four generalized coordinates for the CFL3D solution and the ROM solution. The CFL3D responses are from a CFL3D solution that includes both static and dynamic aeroelastic responses simultaneously. Typically, static and dynamic aeroelastic responses are computed separately in order 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 in order 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 ROM. For the present paper, the sample results presented in Figure 30 are for a full solution; that is, a solution that includes both static and dynamic responses for a given dynamic pressure. In this case, the results presented are for a dynamic pressure of 0.1 psi. Notice that the solutions start at a generalized coordinate value of zero (static, rigid solution) and converge to a non-zero 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 ROM captures the combined static and dynamic aeroelastic responses predicted by the CFL3D solution.

Concluding Remarks

An improved method for generating unsteady aerodynamic state-space reduced-order models (ROMs) was introduced that enables the simultaneous excitation of an unsteady aerodynamic CFD model, consisting of any number of modes, with a single CFD execution. This is achieved by the introduction of orthogonal functions as the input excitations to the system. Comparisons of generalized aerodynamic forces (GAFs) were presented for the lagged step input, block pulse input, and the Walsh function input. Results compared well with the results from the one-mode-at-a-time method that, for four modes, required four separate CFD executions. Additional research is underway to continue the evaluation of these functions on more complex configurations with a larger number of modes in order to define the optimal input (and input amplitudes) for the simultaneous excitation of unsteady aerodynamic systems. In addition, research will be performed in the system identification arena in order to minimize the error associated with the identified state-space models.

References

- ¹Bisplinghoff, R. L. and Ashley, H., *Principles of Aeroelasticity*, Dover Publications, Inc., New York, 1975.
- ²Wagner, H., “Über die Entstehung des dynamischen Auftriebes von Tragflügeln,” *Mathematical Mechanics*, 1925.
- ³Kussner, H. G., “Schwingungen von Flugzeugflügeln,” *Luftfahrtforschung*, Vol. 4, 1929.
- ⁴Theodorsen, T., “General Theory of Aerodynamic Instability and the Mechanism of Flutter,” Tech. rep., NACA, Report 496, 1935.
- ⁵Sears, W. R., “Some Aspects of Non-Stationary Airfoil Theory and Its Practical Applications,” *Journal of the Aeronautical Sciences*, Vol. 8, No. 3, pp. 104–108.
- ⁶Giesing, 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.
- ⁷Silva, W. A., “Identification of Nonlinear Aeroelastic Systems Based on the Volterra Theory: Progress and Opportunities,” *Journal of Nonlinear Dynamics*, Vol. 39, Jan. 2005.
- ⁸Silva, 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*, No. 19, 2004, pp. 729–745.

⁹Beran, 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.

¹⁰Kim, T., Hong, M., Bhatia, K. G., and SenGupta, G., “Aeroelastic Model Reduction for Affordable Computational Fluid Dynamics-Based Flutter Analysis,” *AIAA Journal*, Vol. 43, 2005, pp. 2487–2495.

¹¹Raveh, D. E., “Identification of Computational-Fluid-Dynamic Based Unsteady Aerodynamic Models for Aeroelastic Analysis,” *Journal of Aircraft*, Vol. 41, June 2004, pp. 620–632.

¹²Silva, 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.

¹³Silva, 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.

¹⁴Silva, W. A., *Discrete-Time Linear and Nonlinear Aerodynamic Impulse Responses for Efficient CFD Analyses*, Ph.D. thesis, College of William & Mary, December 1997.

¹⁵Silva, W. A., “Application of Nonlinear Systems Theory to Transonic Unsteady Aerodynamic Responses,” *Journal of Aircraft*, Vol. 30, 1993, pp. 660–668.

¹⁶Raveh, 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.

¹⁷Roger, K. L., “Airplane Math Modeling Methods for Active Control Design,” Agard-cp-228, Aug. 1977.

¹⁸Karpel, M., “Time Domain Aeroservoelastic Modeling Using Weighted Unsteady Aerodynamic Forces,” *Journal of Guidance, Control, and Dynamics*, Vol. 13, No. 1, 1990, pp. 30–37.

¹⁹Gupta, 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.

²⁰Cowan, 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.

²¹Cowan, 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.

²²Rodrigues, 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.

²³Juang, 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.

²⁴“Registered Product of the MathWorks, Inc.” .

²⁵Kim, T., “Efficient Reduced-Order System Identification for Linear Systems with Multiple Inputs,” *AIAA Journal*, Vol. 43, 2005, pp. 1455–1464.

²⁶Silva, 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*, No. AIAA Paper No. 2007-2051, Honolulu, HI, April 23-26 2007.

²⁷Krist, S. L., Biedron, R. T., and Rumsey, C. L., “CFL3D User’s Manual Version 5.0,” Tech. rep., NASA Langley Research Center, 1997.

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

²⁹Bartels, R. E., Rumsey, C. L., and Biedron, R. T., “CFL3D Version 6.4: General Usage and Aeroelastic Analysis,” *NASA TM 2006 214301*, April 2006.

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

³¹Juang, 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.

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

³³Silva, 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*, No. AIAA Paper No. 2007-1988, Honolulu, HI, April 23-26 2007.

³⁴Eykhoff, P., *System Identification: Parameter and State Identification*, Wiley Publishers, 1974.

³⁵Ljung, L., *System Identification: Theory for the User*, Prentice-Hall Publishers, 1999.

³⁶Zhu, Y., *Multivariable System Identification for Process Control*, Pergamon Publishers, 2001.

³⁷Hong, 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*, No. AIAA Paper No. 2003-1923, Norfolk, VA, April 7-10 2003.

List of Figures

1	Coupling of structure and aerodynamics within an aeroelastic CFD code. . .	19
2	Identification of generalized aerodynamic forces (GAFs).	20
3	Block pulse functions.	21
4	Haar (wavelet) functions.	22
5	Walsh functions.	23
6	Computational grid of supersonic semi-span configuration.	24
7	CFL3D response in the first generalized coordinate due to excitation using four lagged-step inputs with a lag of 15 time steps between each input. . . .	25
8	CFL3D response in the first generalized coordinate due to excitation using four lagged-step inputs with a lag of 50 time steps between each input. . . .	26
9	Block pulse functions input to the CFL3D RSM model for modal excitation.	27
10	CFL3D response in the first generalized coordinate due to excitation using block pulse functions.	28
11	Walsh input function for the first mode (shown separately).	29
12	Walsh input function for the first and second modes.	30
13	Walsh input function for the first, second, and third modes.	31
14	Walsh input function for the first, second, third, and fourth modes.	32
15	CFL3D response in the first generalized coordinate due to excitation using Walsh functions.	33
16	Initial portion of the time-domain GAF in mode 1 due to an input in mode 1 (GAF 1,1).	34
17	Variation of GAF 1,1 for the four input functions from the 20th to the 200th time step.	35
18	Real component of GAF 1,1 for the lagged step (15 time steps) and serial step inputs.	36
19	Imaginary component of GAF 1,1 for the lagged step (15 time steps) and serial step inputs.	37
20	Real component of GAF 1,1 for the lagged step (50 time steps) and serial step inputs.	38
21	Imaginary component of GAF 1,1 for the lagged step (50 time steps) and serial step inputs.	39
22	Real component of GAF 1,1 for the block pulse and serial step inputs.	40
23	Imaginary component of GAF 1,1 for the block pulse and serial step inputs.	41
24	Real component of GAF 1,1 for the Walsh and serial step inputs.	42
25	Imaginary component of GAF 1,1 for the Walsh and serial step inputs. . . .	43
26	Real component of GAF 2,2 for the lagged step (50 time-step delays) and serial step inputs.	44
27	Imaginary component of GAF 2,2 for the lagged step (50 time-step delays) and serial step inputs.	45
28	Real component of GAF 2,2 for the Walsh and serial step inputs.	46
29	Imaginary component of GAF 2,2 for the Walsh and serial step inputs. . . .	47
30	Full solution (static plus dynamic aeroelastic responses) generalized coordinate responses at a dynamic pressure of 0.1 psi.	48

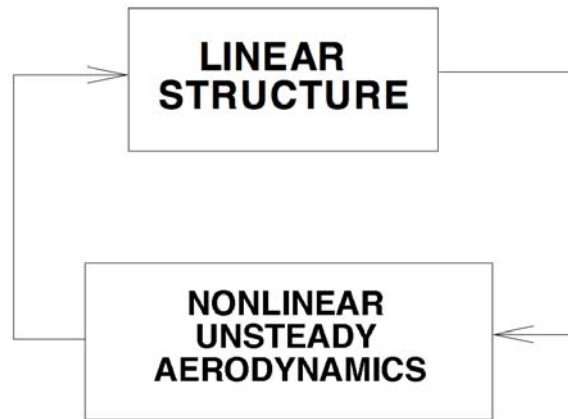


Figure 1: Coupling of structure and aerodynamics within an aeroelastic CFD code.

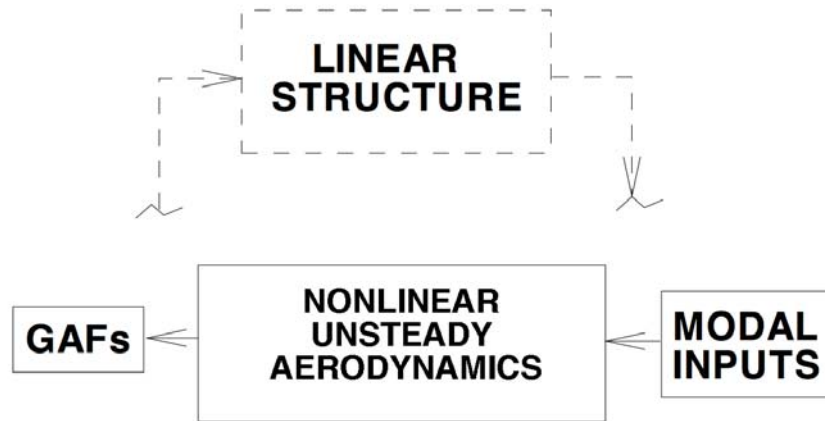


Figure 2: Identification of generalized aerodynamic forces (GAFs).

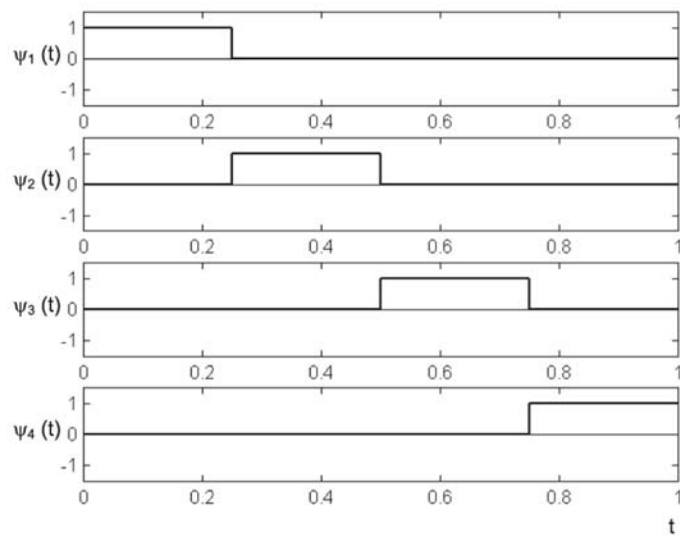


Figure 3: Block pulse functions.

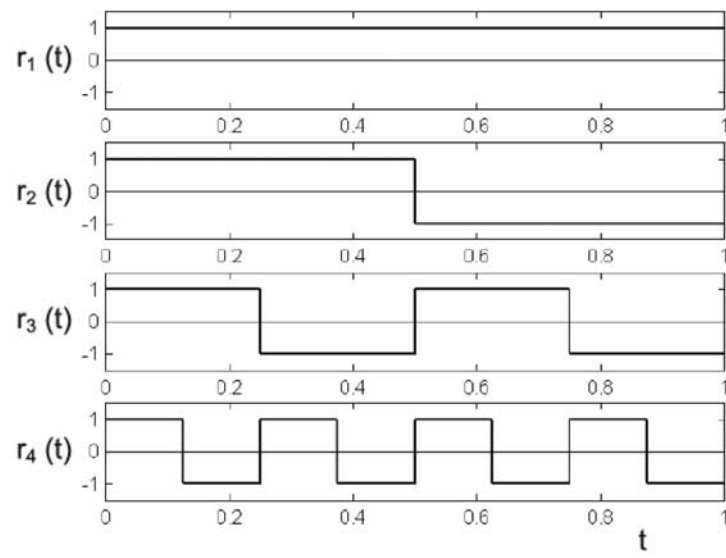


Figure 4: Haar (wavelet) functions.

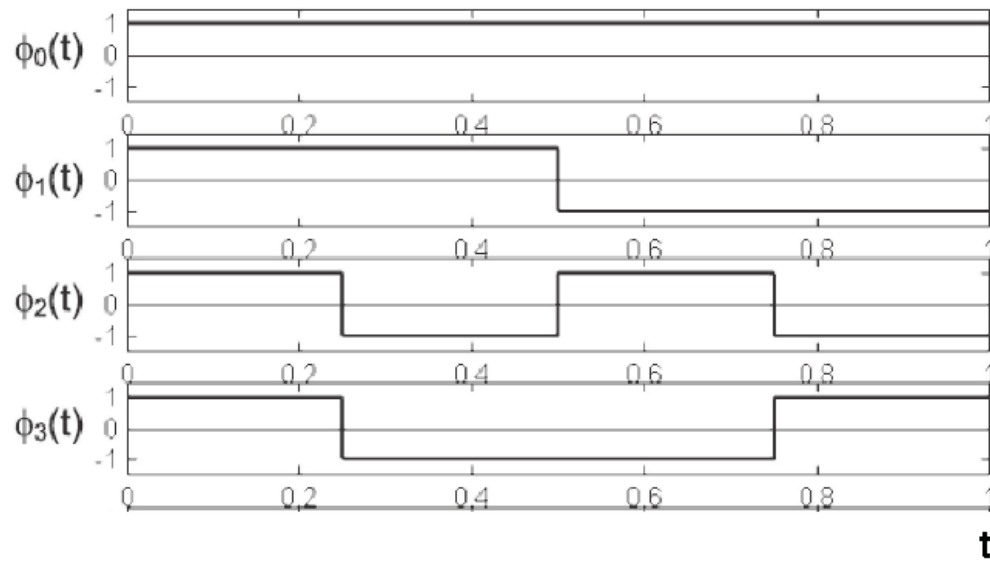


Figure 5: Walsh functions.

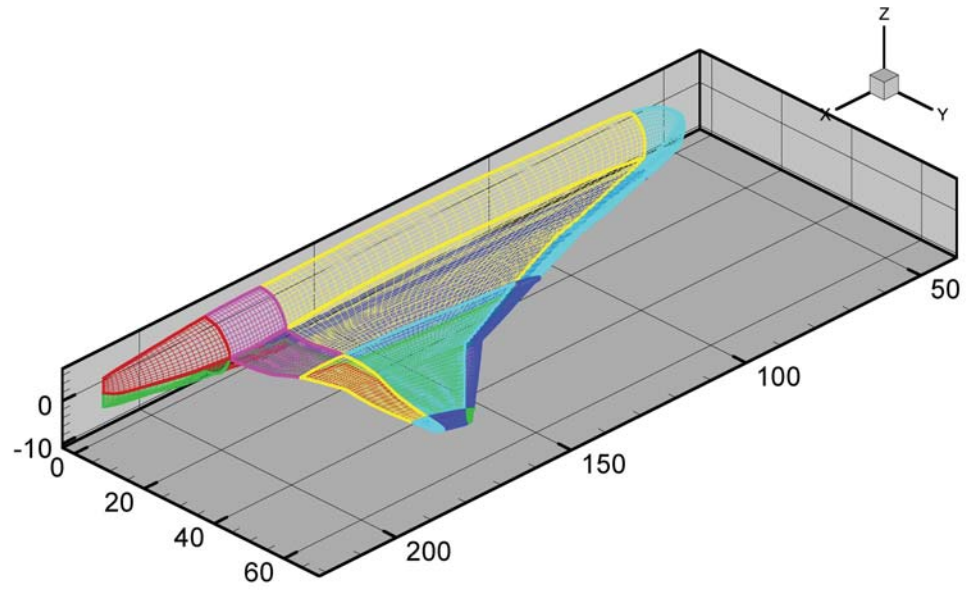


Figure 6: Computational grid of supersonic semi-span configuration.

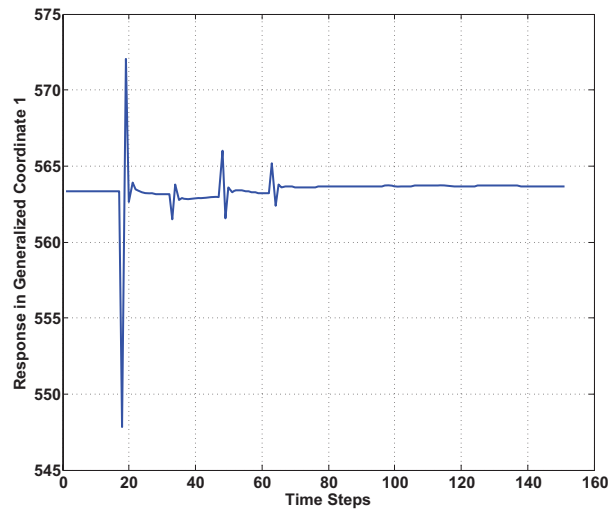


Figure 7: CFL3D response in the first generalized coordinate due to excitation using four lagged-step inputs with a lag of 15 time steps between each input.

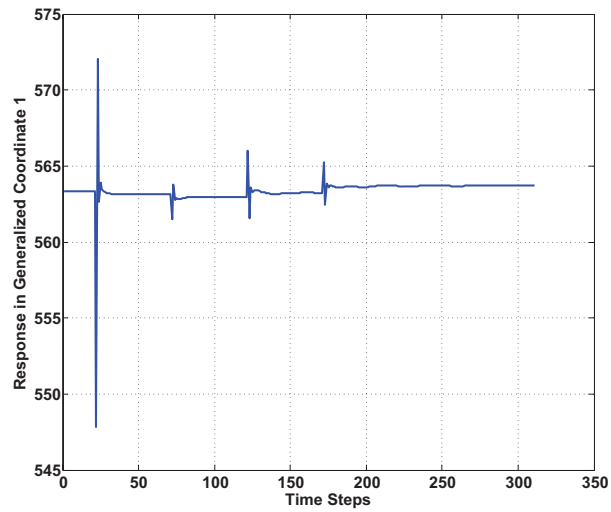


Figure 8: CFL3D response in the first generalized coordinate due to excitation using four lagged-step inputs with a lag of 50 time steps between each input.

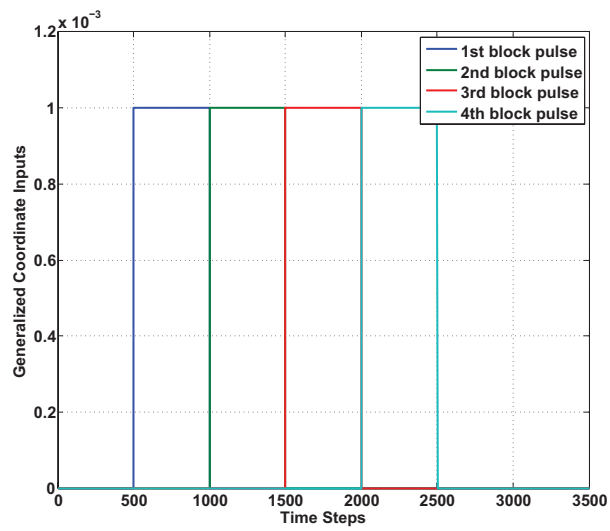


Figure 9: Block pulse functions input to the CFL3D RSM model for modal excitation.

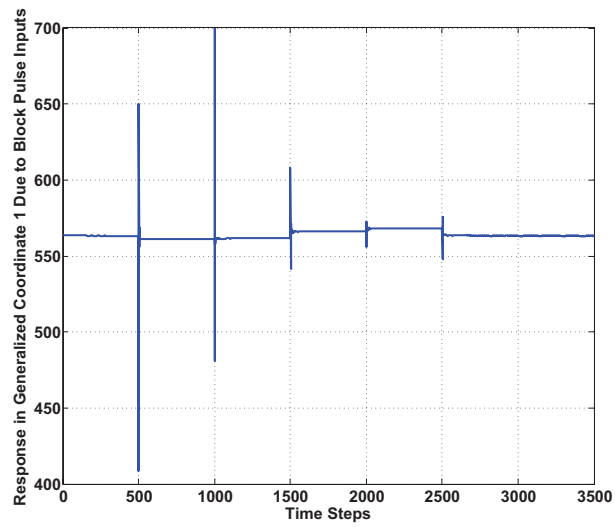


Figure 10: CFL3D response in the first generalized coordinate due to excitation using block pulse functions.

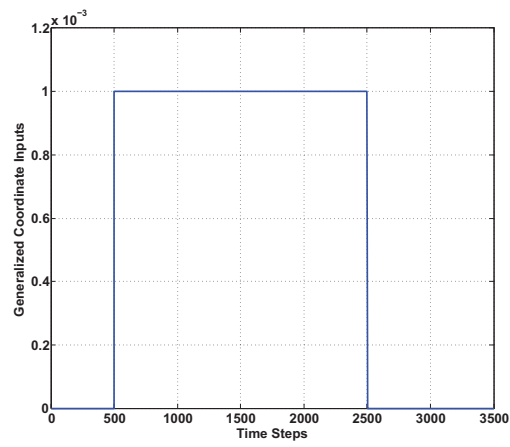


Figure 11: Walsh input function for the first mode (shown separately).

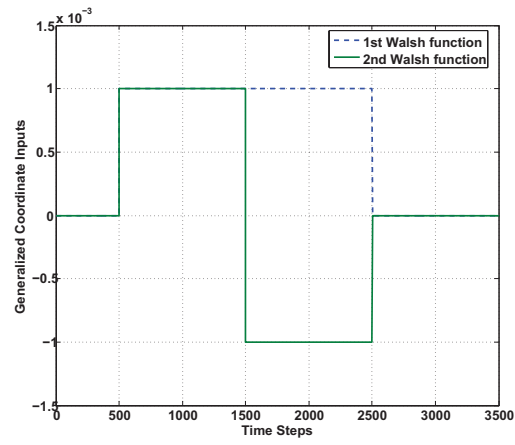


Figure 12: Walsh input function for the first and second modes.

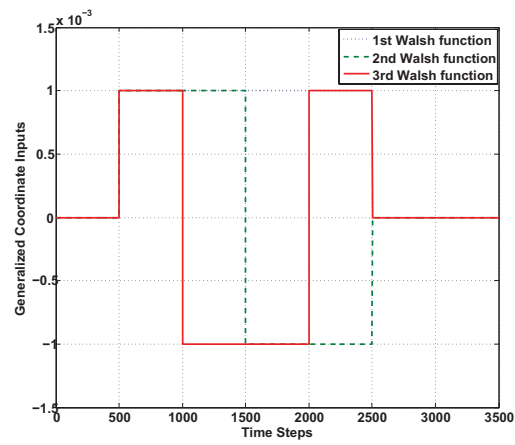


Figure 13: Walsh input function for the first, second, and third modes.

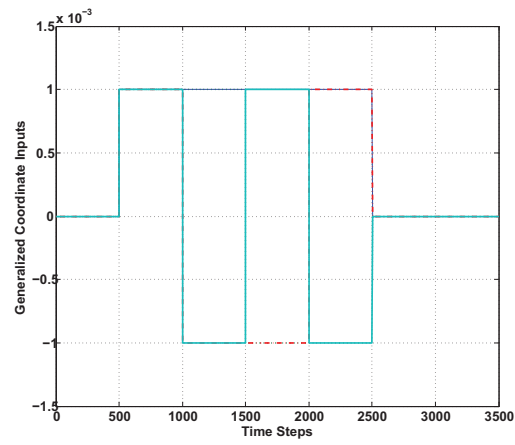


Figure 14: Walsh input function for the first, second, third, and fourth modes.

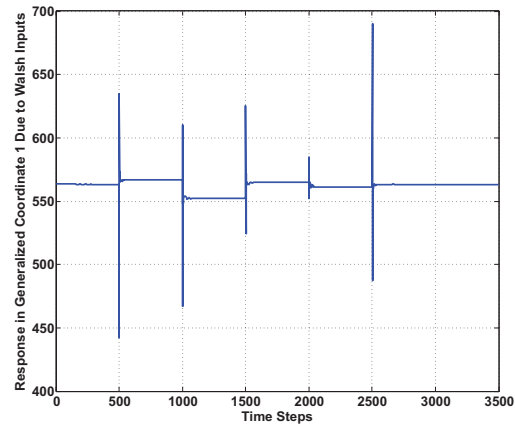


Figure 15: CFL3D response in the first generalized coordinate due to excitation using Walsh functions.

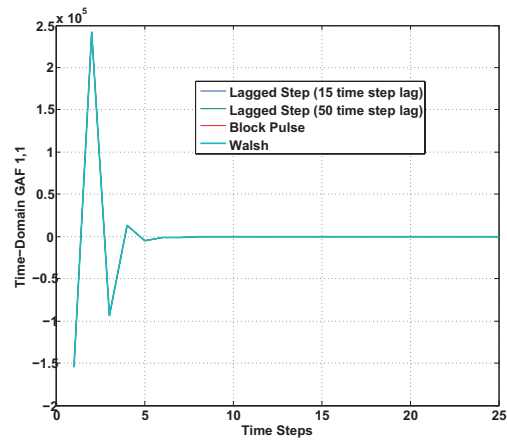


Figure 16: Initial portion of the time-domain GAF in mode 1 due to an input in mode 1 (GAF 1,1).

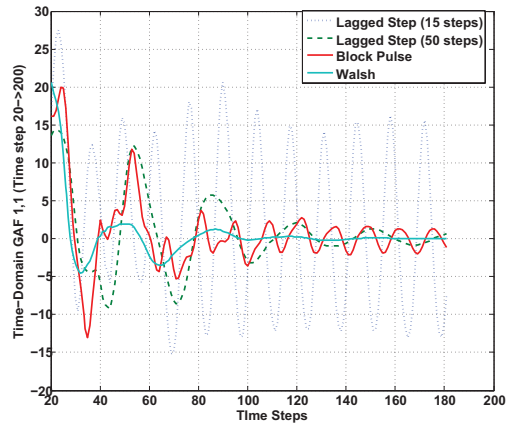


Figure 17: Variation of GAF 1,1 for the four input functions from the 20th to the 200th time step.

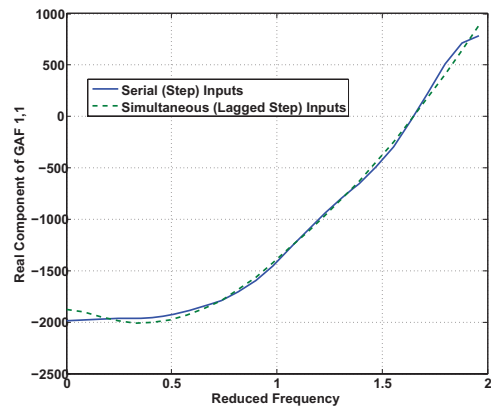


Figure 18: Real component of GAF 1,1 for the lagged step (15 time steps) and serial step inputs.

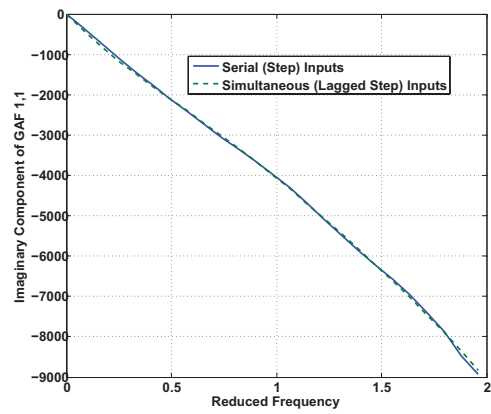


Figure 19: Imaginary component of GAF 1,1 for the lagged step (15 time steps) and serial step inputs.

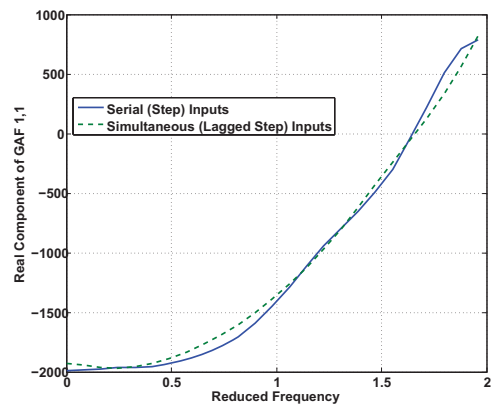


Figure 20: Real component of GAF 1,1 for the lagged step (50 time steps) and serial step inputs.

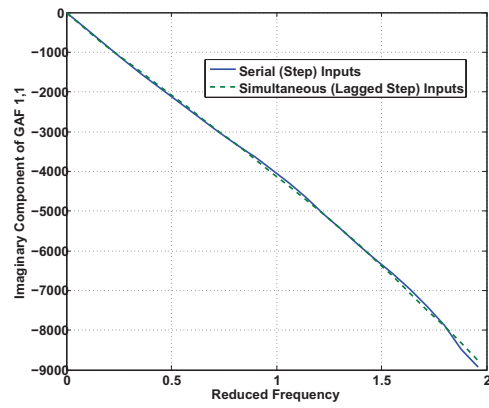


Figure 21: Imaginary component of GAF 1,1 for the lagged step (50 time steps) and serial step inputs.

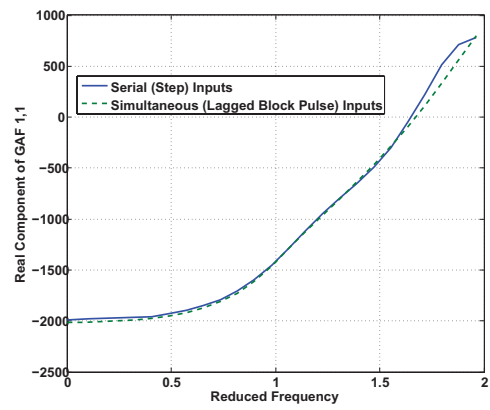


Figure 22: Real component of GAF 1,1 for the block pulse and serial step inputs.

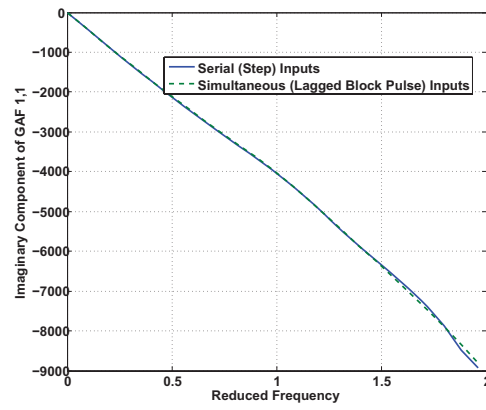


Figure 23: Imaginary component of GAF 1,1 for the block pulse and serial step inputs.

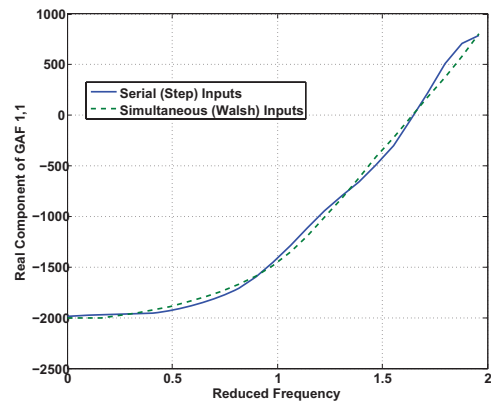


Figure 24: Real component of GAF 1,1 for the Walsh and serial step inputs.

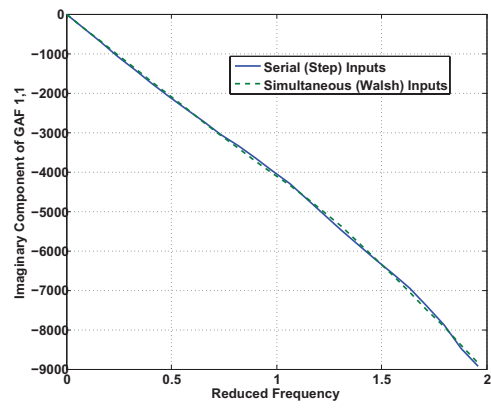


Figure 25: Imaginary component of GAF 1,1 for the Walsh and serial step inputs.

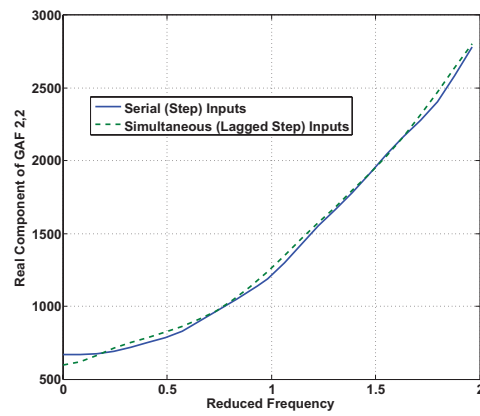


Figure 26: Real component of GAF 2,2 for the lagged step (50 time-step delays) and serial step inputs.

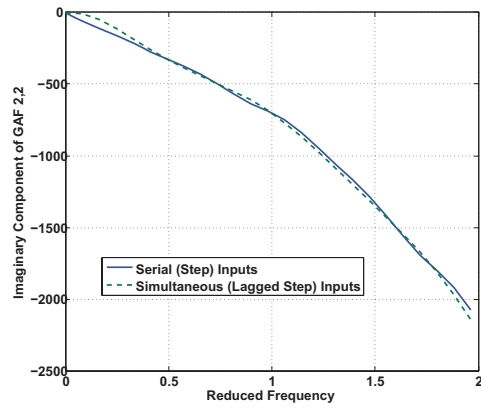


Figure 27: Imaginary component of GAF 2,2 for the lagged step (50 time-step delays) and serial step inputs.

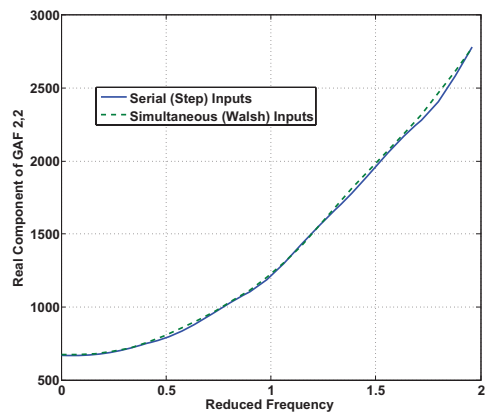


Figure 28: Real component of GAF 2,2 for the Walsh and serial step inputs.

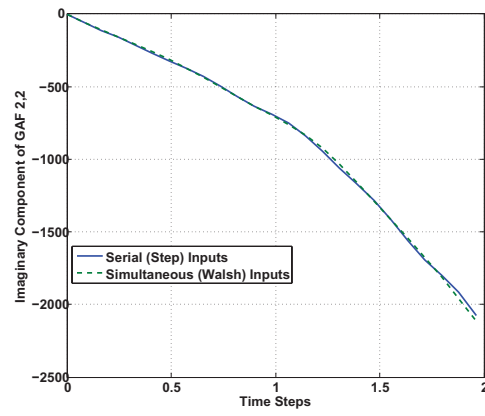


Figure 29: Imaginary component of GAF 2,2 for the Walsh and serial step inputs.

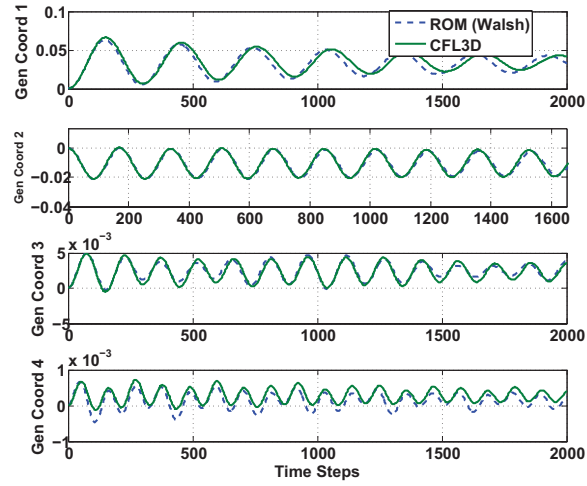


Figure 30: Full solution (static plus dynamic aeroelastic responses) generalized coordinate responses at a dynamic pressure of 0.1 psi.