JOURNAL OF AIRCRAFT
Vol. 49, No. 6, November–December 2012

Reduced-Order Modeling of Flutter and Limit-Cycle Oscillations Using the Sparse Volterra Series

Maciej Balajewicz* and Earl Dowell†
Duke University, Durham, North Carolina 27708-0300

DOI: 10.2514/1.C031637

For the past two decades, the Volterra series reduced-order modeling approach has been successfully used for the purpose of flutter prediction, aeroelastic control design, and aeroelastic design optimization. The approach has been less successful, however, when applied to other important aeroelastic phenomena, such as aerodynamically induced limit-cycle oscillations. Similar to the Taylor series, the Volterra series is a polynomial-based approach capable of progressively approximating nonlinear behavior using quadratic, cubic, and higher-order functional expansions. Unlike the Taylor series, however, kernels of the Volterra series are multidimensional convolution integrals that are computationally expensive to identify. Thus, even though it is well known that aerodynamic nonlinearities are poorly approximated by quadratic Volterra series models, cubic and higher-order Volterra series truncations cannot be identified because their identification costs are too high. In this paper, a novel, sparse representation of the Volterra series is explored for which the identification costs are significantly lower than the identification costs of the full Volterra series. It is demonstrated that sparse Volterra reduced-order models are capable of efficiently modeling aerodynamically induced limit-cycle oscillations of the prototypical NACA 0012 benchmark model. These results demonstrate for the first time that Volterra series models are capable of modeling aerodynamically induced limit-cycle oscillations.

Nomenclature

a	=	nondimensional location of airfoil elastic axis, e/b
b, c	=	semichord and chord, respectively
C_L, C_M	=	lift and moment coefficients, respectively
D_i	=	ith-order single-input sparse Volterra kernel
$\mathbb{D}_i$	=	ith-order single-input sparse Volterra operator
$D_i^{j_1 j_2}$	=	ith-order multi-input sparse Volterra kernel corresponding to inputs j_1 and j_2
$\mathbb{D}_i^m$	=	ith-order m-input sparse Volterra operator
$\mathbf{D}_p^{j_1 j_2}$	=	vectorized sparse Volterra kernel $D^{j_1 j_2}$
e	=	location of airfoil elastic axis measured positive aft of airfoil midchord
h, α	=	airfoil plunge and pitch degree of freedom, respectively
H_i	=	ith-order single-input Volterra kernel
$\mathbb{H}_i$	=	ith-order single-input Volterra operator
$H_i^{j_1 \cdots j_i}$	=	ith-order multi-input Volterra kernel corresponding to inputs $j_1, \ldots, j_i$
$\mathbb{H}_i^m$	=	ith-order m-input Volterra operator
$\mathbf{H}_p^{j_1 \cdots j_i}$	=	vectorized Volterra kernel $H_p^{j_1 \cdots j_i}$
I_α	=	second moment of inertia about elastic axis
k	=	reduced frequency based on airfoil semichord, $\omega b/U_\infty$
m	=	airfoil sectional mass
$\underline{M}_p^{j_1 \cdots j_i}$	=	matrix containing terms of a pth-degree polynomial expansion of inputs $j_1, \ldots, j_i$
n	=	discrete-time index
N_{ID}	=	length of training input sequence
N_i	=	ith time lag
r_a	=	radius of gyration of airfoil about elastic axis; r_a^2 is identical to I_α/mb^2
S_α	=	first moment of inertia about elastic axis
t	=	continuous time
U_∞	=	freestream velocity
V	=	reduced velocity, $U_\infty/\omega_\alpha b$
x	=	input of nonlinear system Ψ
x_α	=	airfoil static unbalance, S_α/mb
y	=	output of nonlinear system Ψ
$\mathbf{y}$	=	vector containing observed output samples
α_{LCO}	=	limit-cycle oscillation pitch amplitude predicted using computational fluid dynamics
$\tilde{\alpha}_{\mathrm{LCO}}$	=	limit-cycle oscillation pitch amplitude predicted using sparse Volterra reduced-order model
$\Delta\tau$	=	nondimensional time step
μ	=	mass ratio, $m/\pi\rho_\infty b^2$
τ	=	nondimensional time, $U_\infty t/2b$
Ψ	=	nonlinear input–output system
ω_α, ω_h	=	uncoupled natural frequencies of pitch and plunge degrees of freedom

Presented as Paper 2010-2725 at the 51st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Orlando, FL, 12–15 April 2010; received 22 August 2011; revision received 2 January 2012; accepted for publication 7 February 2012.

*Research Assistant, Department of Mechanical Engineering and Materials Science. Student Member AIAA.

†William Holland Hall Professor, Department of Mechanical Engineering and Materials Science; Dean Emeritus, Pratt School of Engineering. Honorary Fellow AIAA.

I. Introduction

EVERY aircraft flying today is required to meet stringent, federally enforced regulations developed specifically to prevent catastrophic aeroelastic failure. Although, ultimately, compliance with these regulations must be demonstrated through full-scale flight testing, aircraft manufacturers are keen to minimize expensive flight tests through preliminary computer simulations. Albeit cheaper than full-scale flight tests, computer simulations offer their own unique set of challenges. In particular, efficient and accurate modeling of transonic flight regimes have proved most elusive. Transonic flight is characterized by complex aerodynamic features such as shock waves and turbulent boundary layers; both of which are difficult to model efficiently using computer simulations. Even with dedicated state-of-the-art computer hardware, months of expensive simulation time is necessary to model adequately certain transonic flight regimes. Given that many modern commercial aircraft are designed specifically to operate at transonic conditions, research interest in this area has been substantial. A significant portion of this effort

has been directed toward developing computationally efficient reduced-order models (ROMs) of transonic aerodynamics and aeroelastics [1–3]. A ROM is a simplified mathematical model that captures most of the important features of the more complex system under investigation. Among the many ROM methods available, the most popular include proper orthogonal decomposition [4], balanced realization [5], harmonic balance [6], automatic/algorithmic differentiation [7], artificial neural networks [8], center manifold theory [9], trajectory piecewise linearization [10], and the Volterra series [2,11,12].

Although each of the ROM paradigms listed has its own advantages and proven capability, some benefits of the Volterra series ROM approach include the following:

1) The first benefit is that no modification to simulation software required. The Volterra ROM generation is an exercise in system identification; kernels of the Volterra series are identified from "training" input–output data sets. Since the origin of the training data set is arbitrary from the identification point of view, Volterra ROMs can be generated using any capable computational fluid dynamics (CFD) software, including proprietary (ANSYS CFX/Fluent, FLUIDYN, CFD-FASTRAN, etc.), open source (OPENfoam, OVERFLOW, etc.), or in-house codes. In addition, any fluid dynamics model (panel method, Reynolds-averaged Navier–Stokes, large-eddy simulation, detached-eddy simulation, direct numerical simulation, etc.) can be used for Volterra ROM synthesis. On the other hand, methods such as harmonic balance, center manifold theory, automatic differentiation, and trajectory piecewise linearization require various degrees of CFD code modification, restricting application to open-source CFD and limiting choice of fluid models.

2) The second benefit is the frequency and the time domain capability. Once a Volterra ROM is generated in the frequency or time domain, conversion between the two is trivial, mathematically tractable, and robust. Methods such as piecewise linearization and neural networks often lack this crucial versatility.

3) The last benefit is the intrinsic nonlinear modeling capability. The shock waves characterizing transonic flight introduce strong nonlinearities into the aeroelastic system that trigger complex aeroelastic phenomena such as limit-cycle oscillations (LCOs). As a full-fledged nonlinear modeling methodology, the Volterra series is a natural candidate for modeling in such scenarios. Primarily linear ROM methods such as balanced realization require nontrivial modifications to facilitate modeling of nonlinear aeroelastic behavior.

II. Volterra Reduced-Order Models: Progress and Opportunity

Despite the large interest in the field, the practical generation of Volterra series ROMs remains a challenge, and many important issues remain unresolved. In his recent review paper, Silva suggests that "...potential disadvantages of the Volterra theory include input amplitude limitations related to convergence issues and the need for higher-order kernels" [2]. Identification of Volterra series kernels is a resource-intensive endeavor limiting most aeroelastic applications to linear/weakly nonlinear systems. For transonic aeroelastic applications, this requirement translates to small structural perturbations in low transonic flow regimes [13].

In response to these observations, recent efforts by the authors have focused on developing advanced techniques to reduce the high computational cost associated with Volterra ROM synthesis. More specifically, the concept of a sparse Volterra series ROM has been investigated. Research has been focused on prototyping the method on two-dimensional aerodynamic configurations. The results of these preliminary analyses have been promising and suggest that significant computational efficiency can be achieved through application of a sparse Volterra ROM [14]. More importantly, these results support the sparse Volterra ROM method as a candidate for computationally efficient modeling of more realistic three-dimensional aircraft configurations.

In the current paper, the sparse Volterra ROM methodology is extended to aeroelastic systems. Specifically, the flutter boundary and LCOs of the NACA 0012 benchmark model [15] are modeled using the sparse Volterra ROM methodology.

III. Volterra Theory

The Volterra theory of nonlinear systems is quite mature, and several texts are available [16,17]. It was first applied to nonlinear engineering problems by Wiener [18] and first applied to subsonic and transonic aerodynamic systems by Tromp and Jenkins [19] and Silva [20], respectively. This section provides a brief summary of the Volterra theory of single- and multiple-degree-of-freedom nonlinear systems in the discrete-time domain; formulations in continuous time are available in Balajewicz et al. [13].

A. Volterra Series

The output $y[n]$ of a uniformly sampled discrete-time representation of a causal, time-invariant, fading memory, nonlinear system Ψ, due to a single-input $x[n]$, can be modeled using the pth-order Volterra series

$$y[n] = \sum_{i=1}^{p} \mathbb{H}_i\{x[n]\} \tag{1}$$

$$\mathbb{H}_i\{x[n]\} = \sum_{k_1=n-N_i}^{n} \cdots \sum_{k_i=n-N_i}^{n} H_i[n-k_1,\ldots,n-k_i] \cdot \prod_{z=1}^{i} x[k_z] \tag{2}$$

where the ith-order Volterra operator $\mathbb{H}_i$ is defined as a i-fold discrete-time convolution between the input $x[n]$ and the ith-order discrete-time Volterra kernel $H_i[n,\ldots,n]$. Assuming the system is at rest at $n<0$, it follows that $n \in \mathbb{N}$ and the time lag $N_i \in \mathbb{N}$. To model systems featuring odd output symmetry $-\Psi\{x\} = \Psi\{-x\}$, only odd Volterra operators are required; $\mathbb{H}_1, \mathbb{H}_3, \mathbb{H}_5, \cdots$. To model systems with even symmetry, only even operators are required, while for systems with no symmetry, both even and odd Volterra operators are required. Because the summation notation for the Volterra operators in Eq. (2) can be quite complicated, for illustration purposes, an explicit expression for a particular Volterra series expansion is provided in the Appendix.

The output $y[n]$ of a nonlinear system due to m inputs can be modeled using the pth-order multi-input Volterra series [13,21]

$$y[n] = \sum_{i=1}^{p} \mathbb{H}_i^m\{x_1[n],\ldots,x_m[n]\} \tag{3}$$

where for $j_1, j_2, \ldots, j_m \in \mathbb{N}[1,m]$,

$$\mathbb{H}_i^m\{x_1[n],\ldots,x_m[n]\} = \sum_{k_1=n-N_i}^{n} \cdots \sum_{k_i=n-N_i}^{n} H_i^{j_1\cdots j_i}[n-k_1,\ldots,n-k_i] \cdot \prod_{z=1}^{i} x_{j_z}[k_z] \tag{4}$$

The ith-order multi-input Volterra operator $\mathbb{H}_i^m$ is defined as a m^i-fold summation of i-fold discrete-time convolutions between the various combinations of inputs $x_1[n], x_2[n], \ldots, x_m[n]$ and the ith-order multi-input Volterra kernel $H_i^{j_1\cdots j_i}[n,\ldots,n]$. Note the appearance of superscripts on the ith-order multi-input Volterra kernel $H_i^{j_1\cdots j_i}[n,\ldots,n]$. These superscripts identify to which inputs the kernel corresponds. For example, a third-order kernel $H_3^{512}[n,n,n]$ corresponds to inputs $x_5[n], x_1[n]$ and $x_2[n]$. Volterra kernels $H_i^{j_1\cdots j_i}[n,\ldots,n]$ where $j_1 = j_2 = \cdots = j_i$ are called Volterra direct kernels. Volterra kernels for which the superscripts do no match are called Volterra cross kernels. The presence of these Volterra cross kernels differentiates the multi-input Volterra series from the single-input Volterra series summarized earlier. For illustration purposes, an explicit expression for a particular multi-input Volterra series expansion is also provided in the Appendix.

B. Volterra Kernel Identification: Curse of Dimensionality

Identification of discrete-time Volterra kernels using training inputs from the full-order system is a linear problem [17]. As such, a standard least-squares approach such as the pseudo-inverses can be used. Traditionally in aerodynamic and aeroelastic applications, Volterra kernel identification has been performed using impulsive inputs; this method was not used in this paper. Recent studies indicate that the impulse identification method is far from optimal in terms of identification accuracy and efficiency [13,22]. Compared to the impulse identification method, identification schemes of the pseudo-inverse variety are easier to implement, and more robust and accurate [14]. Let $\underline{y} = \{y[0], y[1], \ldots, y[N_{ID}]\}^T$ be the collection of $N_{ID} \in \mathbb{N}$ observed output samples computed using the full-order system. Then, the optimal Volterra kernels of the Volterra series ROM with respect to the N_{ID} observed output samples are equal to

$$\begin{bmatrix} \mathbf{H}_1^{j_1} \\ \vdots \\ \mathbf{H}_2^{j_1 j_2} \\ \vdots \\ \mathbf{H}_p^{j_1 \cdots j_m} \end{bmatrix} = [\, \underline{M}_1^{j_1} \quad \cdots \quad \underline{M}_2^{j_1 j_2} \quad \cdots \quad \underline{M}_p^{j_1 \cdots j_m} \,]^{+} \mathbf{y} \tag{5}$$

where $j_1, j_2, \ldots, j_m \in \mathbb{N}[1, m]$ and $+$ is the Moore–Penrose pseudo-inverse. The vectors $\mathbf{H}$ contain terms of the Volterra kernels, while the matrices M contain the inputs at various times n. Explicit expressions for these parameters are provided in the Appendix. In general, the identification performance of the preceding method is proportional to the length of training input N_{ID}. As a result, the pseudo-inverse identification method is performed iteratively, terminating when the identified Volterra kernels satisfy some predefined convergence criteria.

The length of the training input N_{ID} required for identification convergence grows geometrically with the Volterra series order p. Unfortunately, since the training input must be simulated using the full-order system, the computational cost associated with Volterra series identification also grows geometrically. For example, if identification of a first-order Volterra series requires σ computational resources (CPU time, for example), the identification of a pth-order Volterra series requires σ^p resources. This rapid growth of complexity and identification time is sometimes referred to as the "curse of dimensionality" and is true regardless of the identification method used [16,17]. Fortunately, over the years, numerous methods that attempt to alleviate this cost have been developed [23–25] and several have been applied to aerodynamics applications [14,24,26].

The goal of this paper is to investigate the aeroelastic ROM capabilities of a recently proposed method [14] that uses a sparse Volterra series. In a previous publication [14], the method was demonstrated to efficiently model the nonlinear aerodynamic response of a two-dimensional airfoil forced into high-amplitude pitch oscillations. In the present publication, the method is extended to the dynamics of aeroelastic systems; specifically, the method is used to find the flutter boundary and LCOs of the NACA 0012 benchmark model [15]. The following section provides a brief summary of the sparse Volterra series.

C. Sparse Volterra Series

A sparse Volterra series, sometimes referred to as a pruned Volterra series [14], is a Volterra series composed of parametrically sparse Volterra kernels [25]. A parametrically sparse Volterra kernel is a lower-dimensional approximation of a standard Volterra kernel. Among the various sparse representations available [25], one of the most straightforward is the diagonalized representation where only the diagonal terms of the standard Volterra kernel are used:

$$H_i[n - k_1, \ldots, n - k_i] = 0 \tag{6}$$

for $k_1 \neq k_2 \neq \cdots \neq k_i$. Hence, the pure-diagonal Volterra series for a single-degree-of-freedom system is of the form

$$y[n] = \sum_{i=1}^{p} \mathbb{D}_i\{x[n]\} \tag{7}$$

$$\mathbb{D}_i\{x[n]\} = \sum_{k=n-N_i}^{n} D_i[n - k] \cdot x[k]^i \tag{8}$$

where $D_i[n]$ is the diagonal component of the ith-order Volterra kernel $H_i[n, \ldots, n]$. For illustration purposes, an explicit expression for a particular sparse Volterra series expansion is provided in the Appendix. A pure-diagonal Volterra series for a multiple-degree-of-freedom system is of the form

$$y[n] = \sum_{i=1}^{p} \mathbb{D}_i^m\{x_1[n], \ldots, x_m[n]\} \tag{9}$$

where, if $i \in 2 \cdot \mathbb{N}$,

$$\mathbb{D}_i^m\{x_1[n], \ldots, x_m[n]\} = \sum_{k=n-N_i}^{n} D_i^{j_1 j_2}[n - k] \cdot x_{j_1}[k] \cdot x_{j_2}[k]^{i/2} \tag{10}$$

otherwise,

$$\mathbb{D}_i^m\{x_1[n], \ldots, x_m[n]\} = \sum_{k=n-N_i}^{n} D_i^{j_1 j_2}[n - k] \cdot x_{j_1}[k]^{(i+1/2)} \cdot x_{j_2}[k]^{(i-1/2)} + \sum_{k=n-N_i}^{n} D_i^{j_2 j_1}[n - k] \cdot x_{j_1}[k]^{(i-1/2)} \cdot x_{j_2}[k]^{(i+1/2)} \tag{11}$$

See the Appendix for an illustrative example. For the sake of brevity, from now on we refer to the pure-diagonal sparse Volterra series as simply the sparse Volterra series.

D. Sparse Volterra Kernel Identification

Let $\underline{y} = \{y[0], y[1], \ldots, y[N_{ID}]\}^T$ be the collection of $N_{ID} \in \mathbb{N}$ observed output samples computed using the full-order system. Then, the optimal Volterra kernels of the sparse Volterra series ROM with respect to the N_{ID} observed output samples, where $j_1, j_2 \in \mathbb{N}[1, m]$, equal

$$\begin{bmatrix} \mathbf{D}_1^{j_1} \\ \vdots \\ \mathbf{D}_2^{j_1 j_2} \\ \vdots \\ \mathbf{D}_p^{j_1 j_2} \end{bmatrix} = [\, \underline{M}_1^{j_1} \quad \cdots \quad \underline{M}_2^{j_1 j_2} \quad \cdots \quad \underline{M}_p^{j_1 j_2} \,]^{+} \mathbf{y} \tag{12}$$

where $+$ is the Moore–Penrose pseudo-inverse. The vectors $\mathbf{D}$ contain terms of the sparse Volterra kernels, while the matrices $\underline{M}$ contain the inputs at various times n. Explicit expressions for these parameters are provided in the Appendix.

In stark contrast to the standard Volterra series case, the length of the training input N_{ID} required for identification convergence of a sparse Volterra series grows linearly with order, p. For example, if identification of a first-order sparse Volterra series requires σ computational resources (CPU time for example), the identification of a pth-order sparse Volterra series requires $\sigma \cdot p$ resources.

E. Motivation for a Sparse Representation

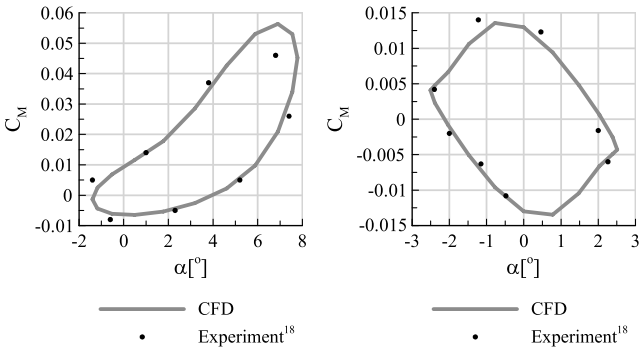
It is well known that sparse representation of the Volterra series can decrease the modeling generality of the full Volterra series. In other words, it is well known that a nth-order sparse Volterra series ROM of a nonlinear system will, in general, be less accurate than a nth-order full Volterra series ROM of the same system. The redeeming quality of the sparse Volterra series is its identification efficiency: the time required to identify a higher-order sparse Volterra series ROM can be substantially lower than the time required to identify a lower-order full Volterra series ROM. Recall that full Volterra series kernels are plagued by the curse of dimensionality: identification times of a full

Table 1 AGARD test cases

Case	$\alpha(t)$	k	M_∞
CT2	$3.16\ \text{deg} + 4.59\ \text{deg}\ \sin(\omega t)$	0.081	0.6
CT5	$0.016\ \text{deg} + 2.51\ \text{deg}\ \sin(\omega t)$	0.081	0.755

Volterra kernel grow geometrically with order, while identification times of a sparse Volterra kernel grow linearly with order. Since higher-order Volterra kernels are associated with stronger nonlinearities, identification of higher-order sparse Volterra kernels can often be more beneficial than the identification of lower-order full Volterra kernels. This benefit is well known and has been successfully exploited by researchers in other disciplines [25]. Among the various sparse representations available, some of the more popular include the diagonal Volterra series [25,27–31], the Hammerstein model [32,33], and the Uryson model [34,35].

It is also worth noting that our choice of a sparse Volterra series was not motivated by the assumption that diagonal terms of Volterra kernels are dominant and/or physically more significant. In fact, it is well known that Volterra kernels of aeroelastic systems are not diagonally dominant and/or more physically significant. The identification of a sparse Volterra ROM relies on the identification of a new set of sparse Volterra kernels that compensate for the absence of offdiagonal terms in a least-squared sense. In other words, we do not identify a sparse Volterra ROM by first identifying a full Volterra series and then retain only the diagonal terms of the full Volterra kernels. Instead, a sparse Volterra series is identified directly using the pseudo-inverse least-squares identification method summarized in Sec. III.D. Using this method, the identified sparse Volterra series ROM is an optimal sparse representation of the full Volterra series ROM with respect to the training input. In general, sparse representations are accurate so long as 1) the phase space neighborhood spanned by the training input is reasonably limited; and 2) the phase-space neighborhood spanned by the identified sparse Volterra ROM remains close to the phase space spanned by the training input. Since many important aeroelastic phenomenon are limited dynamically (i.e., narrow range of reduced frequencies, harmonic oscillation in time, and limited range of structural deflection), we believe sparse Volterra series representations offer an efficient and accurate ROM methodology.



a) CT2 test case **b) CT5 test case**

Fig. 1 Moment coefficient $C_M[n]$ loops for AGARD test cases.

IV. Modeling Aerodynamic Response Using Sparse Volterra Reduced-Order Models

A. Computational Fluid Dynamics Validation

For code validation purposes, the aerodynamic response of a pitching two-dimensional NACA 0012 airfoil are simulated. Specifically, two unsteady test cases from the popular AGARD Technical Report R-702 [36] are simulated. Summarized in Table 1, the aerodynamics of both test cases are characterized by strong shock waves experiencing Tijdeman and Seebass's [37] type B shock motion, where ω is the pitch angular frequency that is related to the reduced frequency $k = \omega c/2U_\infty$. For both test cases, the airfoil oscillates at quarter-chord. The aerodynamics of the NACA 0012 airfoil are modeled using the Carleton multiblock (CMB) CFD code. The CMB code was originally developed at the University of Glasgow, specifically tailored for transonic time-arching aeroelastic analysis. The aerodynamics of the airfoil were modeled using the inviscid Euler equations. The NACA 0012 airfoil domain was discretized using a C-type 180×33 Euler grid with 130 nodes on the airfoil. The first node perpendicular to the airfoil surface was at a distance of approximately $0.001c$. The mesh extended into the far field approximately $10c$ in all directions. The unsteady solutions were solved using a dimensionless time step $\Delta\tau = 1.96$. This choice of mesh and time step was based on several mesh refinement studies carried out by Dubuc et al. [38] and Balajewicz et al. [13,14,26].

Figure 1 compares the steady-state experimental and CFD outputs of the AGARD test cases summarized in Table 1; very good agreement is obtained. Errors are likely associated with the neglect of viscous forces and uncertainties in the experimental data [38,39]. The moment coefficient $C_M[n]$ is measured at the quarter-chord.

B. Band-Limited Random Pitch Input

The sparse Volterra series ROM is used to model the unsteady aerodynamics of a NACA 0012 airfoil oscillating at a quarter-chord at Mach numbers of 0.6, 0.7, 0.755, and 0.8. To validate ROM performance over a wide range of reduced frequencies and amplitudes, a band-limited ($k = 0.04 - 0.12$) 1 deg mean, random pitch input, as illustrated in Fig. 2, is simulated. Sparse Volterra ROMs of the NACA 0012 aerodynamics are identified using the pseudo-inverse method outlined in Sec. III.D. The first 100 data points ($n = 0, 1, \ldots, 100$) are used for system identification, and the other 200 data points ($n = 101, 102, \ldots, 300$) are used for prediction using the identified sparse Volterra ROM. Figure 3 illustrates the output of a first- and fourth-order sparse Volterra ROM relative to the full-order CFD output at Mach number 0.755. Overall, excellent agreement is attained. Modeling performances of the sparse Volterra ROMs are quantified using the percent relative error:

$$\%\text{error} = \frac{\|\mathbf{y}_{\text{CFD}} - \mathbf{y}_{\text{ROM}}\|_2}{\|\mathbf{y}_{\text{CFD}}\|_2} \cdot 100 = \left(\frac{\sum_{n=101}^{300} |\mathbf{y}_{\text{CFD}}[n] - \mathbf{y}_{\text{ROM}}[n]|^2}{\sum_{n=101}^{300} |\mathbf{y}_{\text{CFD}}[n]|^2} \right)^{1/2} \cdot 100 \qquad (13)$$

Figure 4 summarizes the performance of the sparse Volterra ROMs up to the fourth order for Mach numbers 0.6, 0.7, 0.755, and 0.8. Observing Fig. 4, it is clear that good modeling performance is achieved for all Mach numbers selected.

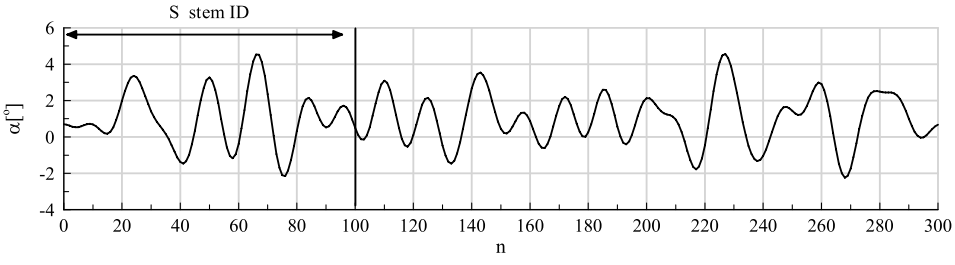


Fig. 2 Unsteady pitch input.

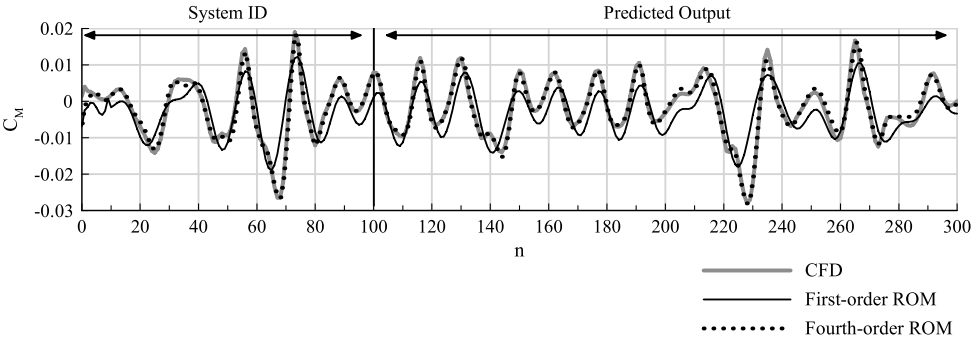


Fig. 3 Full-order CFD output vs first- and fourth-order ROM outputs at $M = 0.755$.

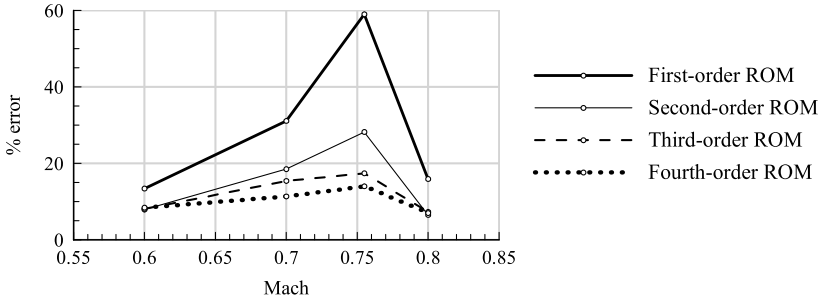


Fig. 4 Sparse Volterra ROM modeling errors.

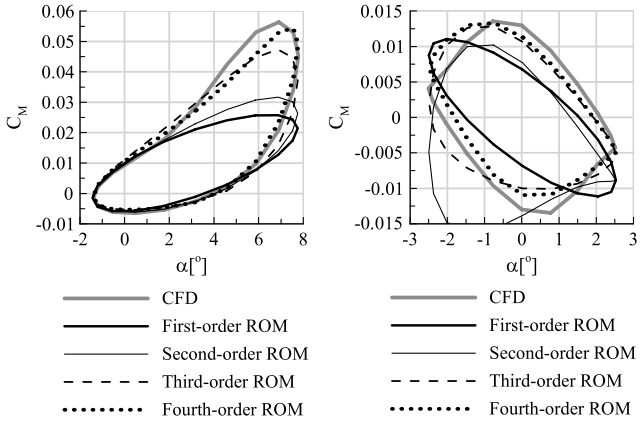
C. AGARD CT2 and CT5 Test Cases

Finally, the identified ROMs are used to model single-frequency inputs. Specifically, the identified ROMs are used to model the AGARD CT2 and CT5 test cases, as shown in Fig. 5. In both cases, good modeling performance is achieved only through higher-order (i.e., third or fourth order) Volterra ROMs.

V. Modeling Aeroelastic Response Using Sparse Volterra Reduced-Order Models

In this section, the aeroelastic dynamics of the two-degree-of-freedom NACA 0012 Benchmark Active Controls Technology (BACT) model are considered:

$$\begin{bmatrix} 1 & x_\alpha \\ x_\alpha & r_\alpha^2 \end{bmatrix} \cdot \begin{bmatrix} h_{\tau\tau}/b \\ \alpha_{\tau\tau} \end{bmatrix} + \frac{4}{V^2} \begin{bmatrix} (\omega_h/\omega_\alpha^2) & 0 \\ 0 & r_\alpha^2 \end{bmatrix} \cdot \begin{bmatrix} h/b \\ \alpha \end{bmatrix} = \frac{4}{\pi\mu} \begin{bmatrix} -C_L(\tau) \\ 2C_M(\tau) \end{bmatrix} \tag{14}$$



a) CT2 test case **b) CT5 test case**

Fig. 5 Sparse Volterra ROMs of AGARD test cases.

The NACA 0012 benchmark model was first analyzed experimentally by Rivera et al. [40] and later numerically using the harmonic balance ROM method by Kholodar et al. [41]. The experiment used a three-dimensional wing model with a constant-chord NACA 0012 section. The structural parameters are $x_\alpha = 0$, $r_\alpha^2 = 1.024$, $\omega_h/\omega_\alpha = 0.646$, $\tau = U_\infty t/2b$, and $a = 0$; where a is the nondimensional location of the airfoil elastic axis measured positive aft of the airfoil midchord.

A. Discretization of the Aeroelastic Ordinary Differential Equation

Assuming a uniformly sampled discrete-time representation of the system

$$x[n] = x(\tau|_{\tau=n\Delta\tau} = x(n\Delta\tau)) \tag{15}$$

where $x \in \{h, \alpha, C_L, C_M\}$ and $n \in \mathbb{N}$, the time derivatives of the aeroelastic ordinary differential equation system are approximated using a first-order explicit Euler scheme:

$$x_\tau \approx \frac{x[n+1] - x[n]}{\Delta\tau} \tag{16}$$

The time step $\Delta\tau$ was progressively decreased until convergent results were obtained. (The results presented in this paper have been replicated using a second-order explicit Euler scheme and a first-order implicit Euler scheme. For the sake of brevity, only the explicit first-order form is presented here.) This formulation produces an explicit expression for h and α of the form

$$\begin{bmatrix} h[n+1] \\ \alpha[n+1] \end{bmatrix} = f\left\{ \begin{bmatrix} h[n] \\ \alpha[n] \end{bmatrix}, \begin{bmatrix} h[n-1] \\ \alpha[n-1] \end{bmatrix}, \begin{bmatrix} C_L[n] \\ C_M[n] \end{bmatrix} \right\} \tag{17}$$

B. Full-Order Modeling of the Aerodynamics

The aerodynamics of the NACA 0012 airfoil are modeled using the CMB CFD code. The aerodynamics of the airfoil were modeled using the inviscid Euler equations. The Euler equations were solved using the same spatial and temporal mesh used in Sec. IV.

C. Reduced-Order Modeling of the Aerodynamics

A sparse Volterra ROM of the NACA 0012 aerodynamics is identified using the pseudo-inverse method outlined in Sec. III.D. The training input consists of random band-limited ($k = 0.04 - 0.12$) zero-mean rigid oscillations in both pitch and heave simultaneously. The first 300 time steps of this training input are shown in Fig. 6.

For each output, $C_L[n]$ and $C_M[n]$, a sparse Volterra ROM is identified using the pseudo-inverse method. Since the NACA 0012 airfoil is a symmetric airfoil, the aerodynamic outputs have odd symmetry. In other words, if the outputs $C_L[n]$ and $C_M[n]$ are known for some input $\alpha[n]$, then the outputs for $-\alpha[n]$ are equal to $-C_L[n]$ and $-C_M[n]$. As a result, to generate a sparse Volterra ROM of the aerodynamics of the NACA 0012 benchmark model, only odd Volterra terms are required; $\mathbb{D}_1, \mathbb{D}_3, \mathbb{D}_5, \cdots$.

Finally, since a finite-order discrete-time Volterra series (both standard and sparse) is a polynomial weighted moving average [25] [see Appendix, Eq. (A1)], it can be expressed in the form

$$\begin{bmatrix} C_L[n] \\ C_M[n] \end{bmatrix} = f_2\left\{\begin{bmatrix} h[n] \\ \alpha[n] \end{bmatrix}, \begin{bmatrix} h[n-1] \\ \alpha[n-1] \end{bmatrix}, \begin{bmatrix} h[n-2] \\ \alpha[n-2] \end{bmatrix}, \cdots\right\} \tag{18}$$

D. Reduced-Order Modeling of the Aeroelastics

An aeroelastic sparse Volterra ROM of the NACA 0012 benchmark model is formed by combining Eqs. (17) and (18), which results in a nonlinear autoregressive model for pitch and heave:

$$\begin{bmatrix} h[n+1] \\ \alpha[n+1] \end{bmatrix} = f_3\left\{\begin{bmatrix} h[n] \\ \alpha[n] \end{bmatrix}, \begin{bmatrix} h[n-1] \\ \alpha[n-1] \end{bmatrix}, \begin{bmatrix} h[n-2] \\ \alpha[n-2] \end{bmatrix}, \cdots\right\} \tag{19}$$

Considering the airfoil at rest for $n < 0$, for a given set of initial structural perturbations (i.e., $h[0] = \delta_h$, $\alpha[0] = \delta_\alpha$), the aeroelastic response is calculated by marching Eq. (19) forward in time.

VI. Results and Discussion

A. NACA 0012 Benchmark Model Flutter and Limit-Cycle Oscillation

Figure 7a compares the experimentally derived flutter boundary of the NACA 0012 benchmark model with the harmonic balance ROM method [41] and the sparse Volterra ROM method. Both the harmonic balance and sparse Volterra ROMs are derived from CFD simulation of the inviscid Euler equations. The harmonic balance ROMs used an in-house Euler solver developed at Duke University, while the sparse Volterra ROMs used the CMB Euler solver. Three sparse Volterra ROMs were generated, one for each Mach number of interest: $M = 0.7, 0.8$, and 0.95 labeled A, B, and C, respectively. For Mach numbers where the inviscid Euler equations are considered good approximations ($M < 0.8$), both ROM models perform well. The large discrepancy between the experimental flutter boundary value and the sparse Volterra value at $M = 0.95$ is typical of inviscid Euler simulations of the transonic flow. It is not possible to capture flow phenomenon such as boundary-layer/shock-wave interactions and shock-wave-induced boundary-layer separation that have strong influence on the aeroelastic behavior of the airfoil at these Mach values using the inviscid Euler equations. Figure 7b illustrates the numerically derived LCO pitch amplitude using the harmonic balance ROM [41] and a third-order sparse Volterra ROM. At lower LCO amplitudes ($\alpha < 3$ deg), there is excellent agreement between the two methods.

As stated in Sec. V.D, the LCO amplitude is determined by marching Eq. (19) forward in time using some small initial perturbations. Figure 8 illustrates one of these simulations using

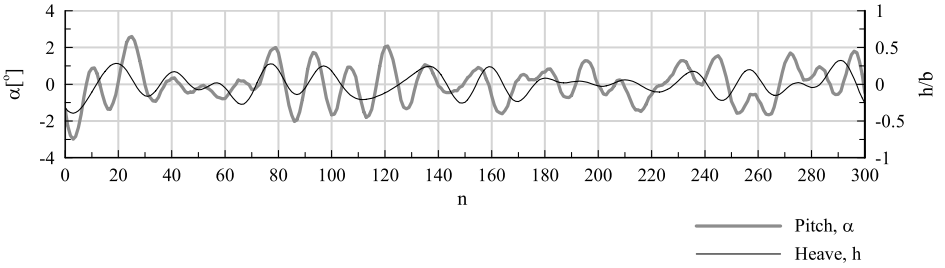
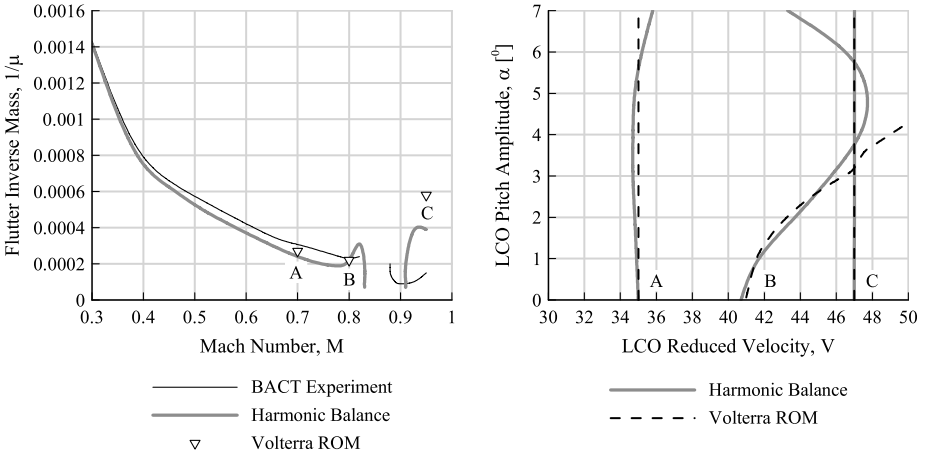


Fig. 6 Sparse Volterra ROM identification training inputs.



a) Flutter boundary **b) LCO pitch amplitude**

Fig. 7 Experimentally and numerically derived aeroelastic responses of NACA 0012 benchmark model.

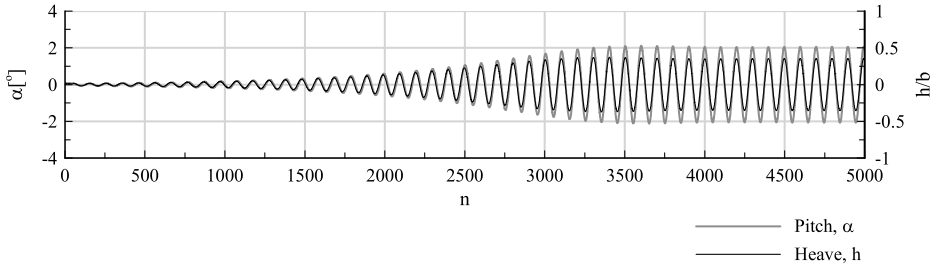


Fig. 8 Sparse Volterra ROM output due to $\delta_\alpha = 0.01$ deg and $\delta_h = 0.01b$ at $M = 0.8$ and $V = 44$.

initial perturbations $\delta_\alpha = 0.01$ deg and $\delta_h = 0.01b$ at Mach number $M = 0.8$ and reduced velocity $V = 44$.

B. Computational Savings

Each third-order sparse Volterra ROM was generated using the first 250 time steps $N_{ID} = 250$ of the training input illustrated in Fig. 6 with time lags $N_1 = 20$ and $N_3 = 10$. The computational cost associated with the generation of each sparse Volterra ROM is overwhelmingly dominated by the computational cost required to simulate the N_{ID} time steps of the training input using the full-order system (i.e., CFD). All CFD simulations presented in this paper were performed on a Intel Pentium 3.2 GHz desktop machine running Red Hat 2.6.9 with 1 GB of RAM. Using this computer, the computational cost per time step was approximately 0.07 min. In other words, the cost of generating each sparse Volterra ROM using a training input of length $N_{ID} = 250$ was approximately $0.07 \cdot 250 = 17.5$ min. Once a sparse Volterra ROM is generated, calculation of LCO amplitudes using the time-marching technique illustrated in Fig. 8 is trivial, requiring less then 1 s of CPU time. Therefore, the total cost associated with the production of Fig. 7 using the sparse Volterra ROM was approximately 53 min (17.5 min per Mach number).

Reproducing Fig. 7 using the full-order system (i.e., coupling the CFD solver with a structural solver) would require substantially more computational time. For example, approximately 350 min of CPU time would be required to reproduce Fig. 8 alone[‡]. To reproduce Fig. 7, several time-domain simulations would be required for various reduced velocities. Although the exact number of simulations would depend on the particular aeroelastic system of interest, approximately 10 simulations per reduced velocity are typical. Therefore, the total cost associated with the reproduction of Fig. 7 using the full-order system would be approximately 10,500 min (3500 min per Mach number).

C. Volterra Reduced-Order Model Identification Convergence

The optimal Volterra order p, training input length N_{ID}, and time lags N_i are not known a priori; these values must determined from an offline convergence study. Figure 9 illustrates the percent relative error

$$\%\text{Error} = \frac{|\alpha_{\text{LCO}} - \tilde{\alpha}_{\text{LCO}}|}{|\alpha_{\text{LCO}}|} \cdot 100 \tag{20}$$

of the LCO amplitude $\tilde{\alpha}_{\text{LCO}}$, calculated using the sparse Volterra ROM for $M = 0.8$ and $V = 44$. The assumed exact LCO amplitude calculated using the harmonic balance method is α_{LCO}. In Fig. 9, the percent relative error is plotted as a function of both the training input length N_{ID} and CPU time required to simulate the training input using CFD. For this convergence study, all other terms including the Volterra order and time lags have been kept constant, i.e., $p = 3$, $N_1 = 20$, and $N_3 = 10$. As expected, the ability of the sparse Volterra ROM to accurately predict the LCO amplitude is proportional to the length of the training input used for identification. The benefits of

[‡]This value was calculated assuming a full-order aeroelastic solver can achieve similar cost per time-step performance as the aerodynamic solver; approximately 0.07 min /n.

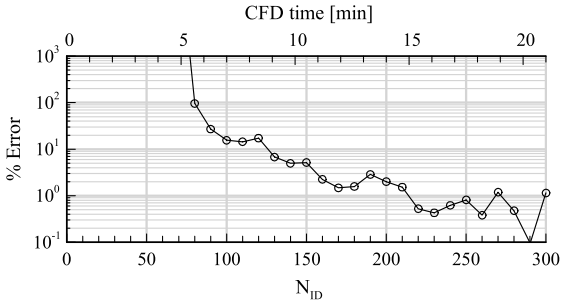


Fig. 9 Sparse Volterra ROM LCO amplitude calculation error vs training input length.

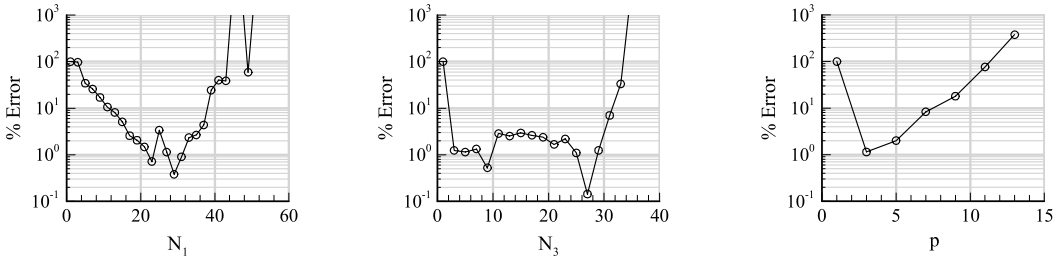
increased accuracy must ultimately be balanced with the cost of training input simulations.

Figures 10a–10c illustrate the percent relative error of the sparse Volterra ROM as a function of time lags N_1 and N_3 and Volterra order p. As in the previous study, all other parameters except the one identified on the abscissa are kept constant. For all three convergence studies summarized in Figs. 10a–10c, the training input length is also kept constant at $N_{ID} = 250$.

Unlike for the training input length N_{ID}, increasing the time lags and Volterra order does not necessarily lead to an increase in accuracy. Increasing the complexity of the model (i.e., Volterra order and time lags) leads to overfitting: a classic phenomenon in system identification not reserved to the sparse Volterra series [42]. The optimal choice of parameters, such as Volterra order and time lags, is always a balance between model flexibility and model complexity. Observing Figs. 10a–10c, broad zones of high accuracy are clearly visible. It is important to note that these convergence studies do not add significant time and/or computational costs to the ROM identification procedure. This is due to the fact that the sparse Volterra ROM identification procedure is overwhelmingly dominated by the computational costs associated with the initial CFD simulation of the aerodynamic response. Once the aerodynamic response is calculated using CFD, the time and computational costs associated with offline convergence studies are relatively not significant.

VII. Conclusions

The most attractive feature of a pure-diagonal sparse Volterra series reduced-order model (ROM) is the fact that computational costs associated with its identification grow linearly with ROM order. This is in stark contrast to the standard Volterra series case, where computational costs required for its identification grow exponentially with order. In other words, identification of stronger nonlinear aerodynamics using higher-order Volterra ROMs becomes a feasible undertaking. The applicability of the sparse Volterra series ROM to nonlinear aeroelastic systems was demonstrated using the NACA 0012 benchmark model. For the specific transonic Mach numbers analyzed, the identified sparse Volterra ROMs were demonstrated to efficiently and accurately estimate the flutter boundaries and limit-cycle oscillation amplitudes of the NACA 0012 benchmark model.



a) Sparse volterra ROM error vs N_1 **b) Sparse volterra ROM error vs N_3** **c) Sparse volterra ROM error vs order**

Fig. 10 Sparse Volterra ROM LCO amplitude calculation error vs time lags and order.

Appendix

I. Appendix Material for Section III.A

Equation (A1) is an example of a third-order single-input Volterra series with $N_1 = 2$, $N_2 = 1$, and $N_3 = 0$:

$$\begin{aligned} y[n] &= H_1[0]x[n] + H_1[1]x[n-1] + H_1[2]x[n-2] + H_2[0,0]x[n]^2 + H_2[1,0]x[n-1]x[n] + H_2[0,1]x[n]x[n-1] \\ &\quad + H_2[1,1]x[n-1]^2 + H_3[0,0,0]x[n]^3 \end{aligned} \tag{A1}$$

As clearly evident from Eq. (A1), a finite-order discrete-time Volterra series is a polynomial-weighted moving average [25]. Equation (A2) is an example of a third-order two-input Volterra series with $N_1 = 1$, $N_2 = 0$, and $N_3 = 0$:

$$\begin{aligned} y[n] &= H_1^1[0]x_1[n] + H_1^1[1]x_1[n-1] + H_1^2[0]x_2[n] + H_1^2[1]x_2[n-1] + H_2^{11}[0,0]x_1[n]^2 + H_2^{12}[0,0]x_1[n]x_2[n] + H_2^{21}[0,0]x_2[n]x_1[n] \\ &\quad + H_2^{22}[0,0]x_2[n]^2 + H_3^{111}[0,0,0]x_1[n]^3 + H_3^{112}[0,0,0]x_1[n]^2x_2[n] + H_3^{121}[0,0,0]x_1[n]^2x_2[n] + H_3^{211}[0,0,0]x_1[n]^2x_2[n] \\ &\quad + H_3^{122}[0,0,0]x_1[n]x_2[n]^2 + H_3^{212}[0,0,0]x_1[n]x_2[n]^2 + H_3^{221}[0,0,0]x_1[n]x_2[n]^2 + H_3^{222}[0,0,0]x_2[n]^3 \end{aligned} \tag{A2}$$

It is important to note that, in general, for nonzero time lags, cross kernels such as H_2^{12} and H_2^{21} are not equivalent.

II. Appendix Material for Section III.B

$$\begin{aligned} \mathbf{H}_1^{j_1} &= \{H_1^{j_1}[0], H_1^{j_1}[1], \cdots\}^T \\ &\vdots \\ \mathbf{H}_2^{j_1 j_2} &= \{H_2^{j_1 j_2}[0,0], H_2^{j_1 j_2}[1,0], \cdots\}^T \\ &\vdots \\ \mathbf{H}_p^{j_1 \cdots j_p} &= \{H_p^{j_1 \cdots j_p}[0,0,\cdots,0], H_p^{j_1 \cdots j_p}[1,0,\cdots,0], \cdots\}^T \end{aligned}$$

$$\begin{aligned} \mathbf{H}_1^{j_1} &= \{H_1^{j_1}[0], H_1^{j_1}[1], \cdots\}^T \\ &\vdots \\ \mathbf{H}_2^{j_1 j_2} &= \{H_2^{j_1 j_2}[0,0], H_2^{j_1 j_2}[1,0], \cdots\}^T \\ &\vdots \\ \mathbf{H}_p^{j_1,\ldots,j_p} &= \{H_p^{j_1 \cdots j_p}[0,0,\cdots,0], H_p^{j_1 \cdots j_p}[1,0,\cdots,0], \cdots\}^T \end{aligned} \tag{A3}$$

$$\underline{M}_1^{j_1} = \begin{bmatrix} x_{j_1}[0] & 0 & 0 & \\ x_{j_1}[1] & x_{j_1}[0] & 0 & \\ x_{j_1}[2] & x_{j_1}[1] & x_{j_1}[0] & \\ & & & \ddots \end{bmatrix}$$

$$\vdots$$

$$\underline{M}_2^{j_1 j_2} = \begin{bmatrix} x_{j_1}[0]\cdot x_{j_2}[0] & 0 & 0 & 0 & \\ x_{j_1}[1]\cdot x_{j_2}[1] & x_{j_1}[1]\cdot x_{j_2}[0] & x_{j_1}[0]\cdot x_{j_2}[0] & 0 & \\ x_{j_1}[2]\cdot x_{j_2}[2] & x_{j_1}[2]\cdot x_{j_2}[1] & x_{j_1}[1]\cdot x_{j_2}[1] & x_{j_1}[1]\cdot x_{j_2}[0] & \\ & & & & \ddots \end{bmatrix}$$

$$\vdots$$

$$\underline{M}_p^{j_1 \cdots j_p} = \begin{bmatrix} \prod_{n=1}^{p}\{x_{j_n}[0]\} & 0 & 0 & \\ \prod_{n=1}^{p}\{x_{j_n}[1]\} & \prod_{n=1}^{p-1}\{x_{j_n}[1]\}\cdot x_{j_p}[0] & \prod_{n=1}^{p-2}\{x_{j_n}[1]\}\cdot\prod_{n=p-1}^{p}\{x_{j_n}[1]\} & \\ \prod_{n=1}^{p}\{x_{j_n}[2]\} & \prod_{n=1}^{p-1}\{x_{j_n}[2]\}\cdot x_{j_p}[1] & \prod_{n=1}^{p-2}\{x_{j_n}[2]\}\cdot\prod_{n=p-1}^{p}\{x_{j_n}[1]\} & \\ & & & \ddots \end{bmatrix}$$

$$\begin{aligned} \mathbf{H}_1^{j_1} &= \{H_1^{j_1}[0], H_1^{j_1}[1], \cdots\}^T \\ &\vdots \\ \mathbf{H}_2^{j_1 j_2} &= \{H_2^{j_1 j_2}[0,0], H_2^{j_1 j_2}[1,0], \cdots\}^T \\ &\vdots \\ \mathbf{H}_p^{j_1 \cdots j_p} &= \{H_p^{j_1 \cdots j_p}[0,0,\cdots,0], H_p^{j_1 \cdots j_p}[1,0,\cdots,0], \cdots\}^T \end{aligned} \tag{A4}$$

III. Appendix Material for Section III.C

Equation (A5) is an example of a third-order single-input sparse Volterra series with $N_1 = 2$, $N_2 = 1$, and $N_3 = 0$:

$$y[n] = D_1[0]x[n] + D_1[1]x[n-1] + D_1[2]x[n-2] + D_2[0]x[n]^2 + D_2[1]x[n-1]^2 + D_3[0]x[n]^3 \tag{A5}$$

Equation (A6) is an example of a third-order two-input sparse Volterra series with $N_1 = 1$, $N_2 = 0$, and $N_3 = 0$:

$$\begin{aligned} y[n] = {} & D_1^1[0]x_1[n] + D_1^1[1]x_1[n-1] + D_1^2[0]x_2[n] + D_1^2[1]x_2[n-1] \\ & + D_2^{11}[0]x_1[n]^2 + D_2^{12}[0]x_1[n]x_2[n] + D_2^{22}[0]x_2[n]^2 + D_3^{11}[0]x_1[n]^3 \\ & + D_3^{12}[0]x_1[n]^2x_2[n] + D_3^{21}[0]x_1[n]x_2[n]^2 + D_3^{22}[0]x_2[n]^3 \end{aligned} \tag{A6}$$

IV. Appendix Material for Section III.D

$$\begin{aligned} \mathbf{D}_1^{j_1} &= \{D_1^{j_1}[0], D_1^{j_1}[1], \cdots\}^T \\ &\vdots \\ \mathbf{D}_2^{j_1 j_2} &= \{D_2^{j_1 j_2}[0], D_2^{j_1 j_2}[1], \cdots\}^T \\ &\vdots \\ \mathbf{D}_p^{j_1 j_2} &= \{D_p^{j_1 j_2}[0], D_p^{j_1 j_2}[1], \cdots\}^T \end{aligned} \tag{A7}$$

$$\begin{aligned} \underline{M}_1^{j_1} &= \begin{bmatrix} x_{j_1}[0] & 0 & 0 & \\ x_{j_1}[1] & x_{j_1}[0] & 0 & \\ x_{j_1}[2] & x_{j_1}[1] & x_{j_1}[0] & \\ & & & \ddots \end{bmatrix} \\ &\vdots \\ \underline{M}_2^{j_1 j_2} &= \begin{bmatrix} x_{j_1}[0]\cdot x_{j_2}[0] & 0 & 0 & \\ x_{j_1}[1]\cdot x_{j_2}[1] & x_{j_1}[1]\cdot x_{j_2}[0] & 0 & \\ x_{j_1}[2]\cdot x_{j_2}[2] & x_{j_1}[1]\cdot x_{j_2}[1] & x_{j_1}[0]\cdot x_{j_2}[0] & \\ & & & \ddots \end{bmatrix} \\ &\vdots \\ \underline{M}_p^{j_1 j_2} &= \begin{bmatrix} x_{j_1}[0]^{p/2}\cdot x_{j_2}[0]^{p/2} & 0 & 0 & \\ x_{j_1}[1]^{p/2}\cdot x_{j_2}[1]^{p/2} & x_{j_1}[0]^{p/2}\cdot x_{j_2}[0]^{p/2} & 0 & \\ x_{j_1}[2]^{p/2}\cdot x_{j_2}[2]^{p/2} & x_{j_1}[1]^{p/2}\cdot x_{j_2}[1]^{p/2} & x_{j_1}[1]^{p/2}\cdot x_{j_2}[0]^{p/2} & \\ & & & \ddots \end{bmatrix} \end{aligned} \tag{A8}$$

References

[1] Dowell, E., Edwards, J., and Strganac, T., "Nonlinear Aeroelasticity," *Journal of Aircraft*, Vol. 40, No. 5, 2003, pp. 857–874. doi:10.2514/2.6876

[2] Silva, W., "Identification of Nonlinear Aeroelastic Systems Based on the Volterra Theory: Progress and Opportunity," *Nonlinear Dynamics*, Vol. 39, Nos. 1–2, 2005, pp. 25–62. doi:10.1007/s11071-005-1907-z

[3] Lucia, D. J., Beran, P. S., and King, P. I., "Reduced Order Modeling of an Elastic Panel in Transonic Flow," *Journal of Aircraft*, Vol. 40, No. 2, 2003, pp. 338–347. doi:10.2514/2.3098

[4] Thomas, J. P., Dowell, E. H., and Hall, K. C., "Three-Dimensional Aeroelasticity Using Proper Orthogonal Decomposition Based Reduced Order Models," *Journal of Aircraft*, Vol. 40, 2003, pp. 544–551. doi:10.2514/2.3128

[5] Willcox, K., and Peraire, J., "Balanced Model Reduction Via the Proper Orthogonal Decomposition," *AIAA Journal*, Vol. 40, No. 11, 2002, pp. 2323–2330. doi:10.2514/2.1570

[6] Thomas, J. P., Custer, C. H., Dowell, E. H., and Hall, K. C., "Unsteady Flow Computation Using a Harmonic Balance Approach Implemented About the OVERFLOW 2 Flow Solver," *19th AIAA Computational Fluid Dynamics Conference*, American Institute of Aeronautics and Astronautics, San Antonio, Texas, 2009;also AIAA Paper 2009-4270.

[7] Thomas, J. P., Dowell, E. H., and Hall, K. C., "Using Automatic Differentiation to Create a Nonlinear Reduced Order-Model-Aeroelastic Solver," *AIAA Journal*, Vol. 48, No. 1, 2009, pp. 19–24. doi:10.2514/1.36414

[8] Johnson, M. R., and Denegri, C. M., "Comparison of Static and Dynamic Neural Networks for Limit Cycle Oscillation Prediction," *Journal of Aircraft*, Vol. 40, 2003, pp. 194–203. doi:10.2514/2.3075

[9] Woodgate, M. A., and Badcock, K. J., "Fast Prediction of Transonic Aeroelastic Stability and Limit Cycles," *AIAA Journal*, Vol. 45, 2007, pp. 1370–1381. doi:10.2514/1.25604

[10] Gratton, D., and Wilcox, K., "Reduced-Order, Trajectory Piecewise-Linear Models for Nonlinear Computational Fluid Dynamics," Ph.D. Dissertation, Aeronautics and Astronautics Dept., Massachusetts Institute of Technology, 2004.

[11] Raveh, D. E., "Reduced-Order Models for Nonlinear Unsteady Aerodynamics," *AIAA Journal*, Vol. 39, No. 8, 2001, pp. 1417–1429. doi:10.2514/2.1473

[12] Marzocca, P., Librescu, L., and Silva, W., "Aeroelastic Response of Nonlinear Wing Sections Using a Functional Series Technique," *AIAA Journal*, Vol. 40, No. 5, 2002, pp. 813–824. doi:10.2514/2.1735

[13] Balajewicz, M., Nitzsche, F., and Feszty, D., "Application of Multi-Input Volterra Theory to Nonlinear Multi-Degree-of-Freedom Aerodynamic Systems," *AIAA Journal*, Vol. 48, No. 1, Jan. 2010, pp. 56–62. doi:10.2514/1.38964

[14] Balajewicz, M., Nitzsche, F., and Feszty, D., "Reduced Order Modeling of Nonlinear Transonic Aerodynamic Using a Pruned Volterra Series," *50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, American Institute of Aeronautics and Astronautics, Palm Springs, California, May 2009;also AIAA Paper 2009-2319.

[15] Abel, I., "Research and Application in Aeroelasticity and Structural Dynamics at the NASA Langley Research Center," NASA Langley Research Center TM 112852, Hampton, VA, 1997.

[16] Schetzen, M., *The Volterra and Wiener Theories of Nonlinear Systems*, Wiley, New York, 1980.

[17] Rugh, W. J., *Nonlinear System Theory. The Volterra/Wiener Approach*, John Hopkins Univ. Press, Baltimore, MD, 1981.
[18] Wiener, N., "Response of a Nonlinear Device to Noise," Massachusetts Inst. of Technology, Radiation Lab. Rept. 129, Cambridge, MA, 1942.
[19] Tromp, J. C., and Jenkins, J. E., "A Volterra Kernel Identifcation Scheme Applied to Aerodynamic Reactions," Atmospheric Flight Mechanics Conference, AIAA Paper 1990-2803, Aug. 1990.
[20] Silva, W., "Application of Nonlinear Systems Theory to Transonic Unsteady Aerodynamic Responses," *Journal of Aircraft*, Vol. 30, No. 5, 1993, pp. 660–668. doi:10.2514/3.46395
[21] Dimoka, A., Courellis, S. H., Marmarelis, V. Z., and Berger, T. W., "Modeling the Nonlinear Properties of the In Vitro Hippocampal Perforant Path-Dentate System Using Multielectrode Array Technology," *IEEE Transactions on Biomedical Engineering*, Vol. 55, No. 2, 2008, pp. 693–702.
[22] Grewal, A., and Zimcik, D., "Application of Reduced Order Modelling to Fluid-Structure Interaction Analysis," 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, AIAA Paper 2008-5961, Sept. 2008.
[23] Doyle, F. J., Pearson, R., and Ogunnaike, B., *Identification and Control Using Volterra Models*, Springer–Verlag, London, 2002.
[24] Kurdila, A. J., Prazenica, R., Rediniotis, O., and Strganac, T., "Multiresolution Methods for Reduced-Order Models for Dynamical Systems," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 2, 2001, pp. 193–200. doi:10.2514/2.4707
[25] Pearson, R. K., *Discrete-Time Dynamic Models*, Oxford Univ. Press, Oxford, England, U.K., 1999.
[26] Balajewicz, M., Nitzsche, F., and Feszty, D., "A Modified Volterra ROM Method for Nonlinear Aerodynamic Systems," *NATO RTO Specialists Meeting on Advanced Methods in Aeroelasticity*, NATO, Loen, Norway, May 2008, p. 18.
[27] Cadenas, C., Tosina, J., Ayora, M., and Cruzado, J., "A New Approach to Pruning Volterra Models for Power Amplifiers," *IEEE Transactions on Signal Processing*, Vol. 58, No. 4, 2010, pp. 2113–2120. doi: 10.1109/TSP.2009.2039815
[28] Zhu, A., and Brazil, T., "Behavior Modeling of RF Power Amplifiers Based on Pruned Volterra Series," *IEEE Microwave and Wireless Components Letters*, Vol. 14, No. 12, 2004, pp. 563–565. doi:10.1109/LMWC.2004.837380
[29] Zhu, A., Dooley, J., and Brazil, T., "Simplified Volterra Series Based Behavioral Modeling of RF Power Amplifiers Using Deviation-Reduction," *IEEE Microwave and Wireless Components Letters*, Vol. 14, No. 3, 2006, pp. 1113–1116. doi:10.1109/MWSYM.2006.249958
[30] Balajewicz, M., "Reduced Order Modeling of Transonic Flutter and Limit Cycle Oscillations Using the Pruned Volterra Series," 51st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, AIAA Paper 2010-2725, April 2010.
[31] Kibangou, A. Y., and Favier, G., "Wiener-Hammerstein Systems Modeling Using Diagonal Volterra Kernels Coefficients," *IEEE Signal Processing Letters*, Vol. 13, No. 6, June 2006, pp. 381–384. doi:10.1109/LSP.2006.871705
[32] Eskinat, E., Johnson, S., and Luyben, W., "Use of Hammerstein Models in Identification of Nonlinear Systems," *AIChE Journal*, Vol. 37, No. 2, 1991, pp. 255–268. doi:10.1002/aic.690370211
[33] Fruzzetti, K. P., Palazo, A., and McDonald, K. A., "Nonlinear Model Predictive Control Using Hammerstein Models," *Journal of Process Control*, Vol. 7, No. 1, 1997, pp. 31–41. doi:10.1016/S0959-1524(97)80001-B
[34] Gallman, P., "An Iterative Method for the Identification of Nonlinear Systems Using a Uryson Model," *IEEE Transactions on Automatic Control*, Vol. 20, 1975, pp. 771–775. doi:10.1109/TAC.1975.1101087
[35] Billings, S. A., "Identification of Nonlinear Systems A Survey," *IEE Proceedings D, Control Theory and Applications*, Vol. 127, Nov. 1980, pp. 272–285. doi:10.1049/ip-d.1980.0047
[36] "Compendium of Unsteady Aerodynamic Measurements," AGARD TR R-702, 1982.
[37] Tijdeman, H., and Seebass, R., "Transonic Flow Past Oscillating Airfoils," *Annual Review of Fluid Mechanics*, Vol. 12, 1980, pp. 181–222. doi:10.1146/annurev.fl.12.010180.001145
[38] Dubuc, L., Cantariti, F., Woodgate, M., Gribben, B., Badcock, K. J., and Richards, B. E., "Solution of the Euler Unsteady Equations Using Deforming Grids," Univ. of Glasgow, Aeronautical Rept. 9704, Glasgow, 1997.
[39] Badcock, K. J., Richards, B. E., and Woodgate, M. A., "Elements of Computational Fluid Dynamics on Block Structured Grids Using Implicit Solvers," *Progress in Aerospace Sciences*, Vol. 36, No. 5, 2000, pp. 351–392. doi:10.1016/S0376-0421(00)00005-1
[40] Rivera, J., Dansberry, B., Bennett, R., Durham, M., and Silva, W., "NACA0012 Benchmark Model Experimental Flutter ResultsWith Unsteady Pressure Distributions," NASA TM 107581, 1992.
[41] Kholodar, D., Dowell, E., Thomas, J., and Hall, K., "Limit-Cycle Oscillations of a Typical Airfoil in Transonic Flow," *Journal of Aircraft*, Vol. 41, No. 5, 2004, pp. 1067–1072. doi:10.2514/1.618
[42] Hawkins, D. M., "The Problem of Overfitting," *Journal of Chemical Information and Computer Sciences*, Vol. 44, No. 1, 2004, pp. 1–12. doi:10.1021/ci0342472