AIAA JOURNAL
Vol. 48, No. 9, September 2010

Toward Real-Time Computational-Fluid-Dynamics-Based Aeroelastic Computations Using a Database of Reduced-Order Information

David Amsallem,* Julien Cortial,† and Charbel Farhat‡
Stanford University, Stanford, California 94305-4035

DOI: 10.2514/1.J050233

This paper describes a computational-fluid-dynamics-based computational methodology for fast *on-demand* aeroelastic predictions of the behavior of a full aircraft configuration at variable flight conditions and demonstrates its feasibility. The methodology relies on the offline precomputation of a database of reduced-order bases and models associated with a discrete set of flight parameters, and its training for an interpolation method suitable for reduced-order information. The potential of this near-real-time computational methodology for assisting flutter flight testing is highlighted with the aeroelastic identification of an F-16 configuration in the subsonic, transonic, and supersonic regimes.

Nomenclature

$\mathcal{C}$	=	overlapping cluster of database points
$\mathrm{Exp}_{\mathcal{S}}$	=	exponential operator at the point $\mathcal{S}$
$\mathcal{G}(,)$	=	Grassmann manifold
$\mathcal{M}$	=	interpolation model
M_∞	=	freestream Mach number
$\mathcal{N}$	=	number of points in an interpolation model
N_f	=	dimension of the high-fidelity computational fluid dynamics model
N_m	=	number of natural modes of the dry structure
N_p	=	number of processors
N_R	=	number of precomputed reduced-order models
N_Φ	=	size of the reduced-order basis and associated reduced-order model
$\mathcal{P}$	=	set of interpolation data points
p_∞	=	freestream pressure (in $\mathrm{lb/ft^2}$)
$\mathcal{S}$	=	subspace of $\mathbb{R}^{N_f}$ of dimension N_Φ
$\mathcal{T}_{\mathcal{S}}$	=	tangent space to a Grassmann manifold at the point $\mathcal{S}$
$\mathbf{U}$	=	matrix of the left singular orthonormal basis vectors associated with a thin singular value decomposition
$\mathbf{V}$	=	matrix of the right singular orthonormal basis vectors associated with a thin singular value decomposition
α	=	angle of attack, deg
ϵ	=	L_2-norm of the discrepancy in the transient lift
$\mathbf{\Gamma}$	=	matrix representing the coordinates of a point of a tangent space to the Grassmann manifold
$\mathbf{\Sigma}$	=	matrix of singular values
ρ_∞	=	freestream pressure (in $\mathrm{slug/ft^3}$)
$\mathbf{\Phi}$	=	reduced-order orthogonal basis
χ	=	point of the tangent space to a Grassmann manifold
$\bar{}$	=	mean value

Presented as Paper 2009-0800 at the 47th AIAA Aerospace Sciences Meeting, Orlando, FL, 5–8 January 2009; received 21 September 2009; revision received 24 February 2010; accepted for publication 22 April 2010.

*Department of Aeronautics and Astronautics, Durand Building, Room 028, 496 Lomita Mall. Member AIAA.

†Institute for Computational and Mathematical Engineering, Durand Building, Room 028, 496 Lomita Mall.

‡Department of Aeronautics and Astronautics, Department of Mechanical Engineering, Institute for Computational and Mathematical Engineering, Durand Building, Room 257, 496 Lomita Mall. Fellow AIAA.

I. Introduction

IT HAS already been established in the literature that linearized computational fluid dynamics (CFD)-based aeroelastic reduced-order models (ROMs) are capable of predicting the aeroelastic behavior of an airfoil [1], cascade blade [2], three-dimensional wing [3–5], twin-engine transport aircraft [6], and even full fighter aircraft [7–9] configuration with a decent level of accuracy, and of operating in real time [7]. However, it has also been equally well established that such ROMs typically lack robustness with respect to parameter variations [2,7,10], and that their reconstruction for new parameter values is computationally intensive [11]. Consequently, such ROMs can, in general, be exploited in real time only for the parameter set for which they have been built, which limits their usefulness for practical applications. To address this issue, this paper describes a new methodology for exploiting in near-real-time linearized computational ROMs for arbitrary sets of parameter values. This methodology is based on the concept of a database of reduced-order information, and the exploitation of this database by a recently developed scheme for interpolating reduced-order bases (ROBs) [8,12]. Whereas this methodology is quite general and applicable to many computational engineering problems, it is presented in this paper in the context of the application that motivated its development: namely, the *on-demand* prediction of the aeroelastic response of a given aircraft at specified flight conditions. For this reason, the database is populated here with CFD-based ROBs and ROMs precomputed at various pairs of freestream Mach number M_∞ and angle of attack α. It is also discretized, decomposed, trained, and reduced to enable real-time accurate interpolations for the flight conditions not represented in it. Consequently, the methodology described in this paper is particularly suitable for applications such as assisting flutter flight test campaigns, in this case, and for designing active control systems for flutter suppression. Some of its key computational components have already been published in the literature. Therefore, the main objectives of this work are to present for the first time the integration of these key components and the enabling of their near-real-time operation within a database, and to demonstrate the feasibility of the resulting fast computational methodology for a realistic aeroelastic application.

To this effect, the remainder of this paper is organized as follows. In Sec. II, a partitioned and linearized aeroelastic model reduction method is first overviewed. Then the concept of a database of fluid ROBs and structural ROMs is motivated. Next, a recently developed method for the multivariate interpolation of parameterized fluid ROBs [8,12,13] is described. In Sec. III, the resulting linearized CFD-based aeroelastic computational methodology is optimized then demonstrated with on-demand aeroelastic predictions for an F-16 Block 40 configuration in the subsonic, transonic, and

supersonic regimes. A database of CFD-based fluid ROBs is first constructed. Then this database is discretized into dual cells, decomposed, and trained to enable real-time local interpolations when needed. Finally, the performance of the proposed computational methodology is assessed from both CPU and accuracy viewpoints. This performance is also contrasted with that of numerical predictions based on high-dimensional (or full-order) CFD models, which allows offering some conclusions in Sec. IV.

II. Near-Real-Time CFD-Based Aeroelastic Computational Strategy

The CFD-based computational methodology proposed for fast aeroelastic predictions in an on-demand mode is organized around four main steps:

1) A set of reduced-order bases and linearized computational models for a predetermined set of flight parameters (M_{∞_i} and α_i) is computed offline.

2) This precomputed information is compactly stored into a database, and discretization, decomposition, training, and reduction of this database occurs.

3) For a point (M_∞, α) not represented in the database, local interpolation of a subset of the stored reduced-order model information is performed in real time and a linearized ROM is rapidly constructed.

4) This ROM is analyzed in real time for aeroelastic predictions.

Each of these four steps is summarized below.

A. Partitioned Approach for Constructing an Aeroelastic ROM

Here, the construction of a linearized CFD-based aeroelastic ROM for a given set of flight conditions is based on the partitioned fluid–structure approach presented in [8,11,14], among others. In this approach, model reduction is performed separately for the fluid and structure subsystems. Using an appropriate nondimensionalization of the fluid model, this decoupled approach reduces the number of aerodynamic flight parameters from four (freestream Mach number M_∞, angle of attack α, freestream density ρ_∞, and freestream pressure p_∞) to two (namely, M_∞ and α), thereby enabling the reusage of a computed aeroelastic ROM at many different freestream pressure and/or density values (and therefore multiple altitude levels). Consequently, this decoupled approach for constructing an aeroelastic ROM accelerates the computation of the critical flutter speed of an aircraft at specified freestream Mach number and angle of attack. The main purpose of this work, however, is to increase the range of applications of an aeroelastic ROM computed at a single pair of flight parameters (M_∞ and α) beyond that associated with the computation of a flutter speed index at a given freestream Mach number and a given angle of attack.

B. Proper-Orthogonal-Decomposition-Based Fluid ROB

Regarding the fluid subsystem, it is decided to populate the aforementioned database with fluid ROBs rather than ROMs for an important reason explained in Sec. II.D. These ROBs are precomputed using the proper orthogonal decomposition (POD) method applied to fluid snapshots computed in the frequency domain [8,11,14]. The underlying computational framework is that of the linearization around a steady-state equilibrium configuration of the three-field formulation of fluid–structure interaction problems [15]. For each set of flight conditions of the form (M_{∞_i} and α_i), the major steps of this computational framework can be summarized as follows:

1) Compute the steady-state equilibrium solution of the aeroelastic problem at (M_{∞_i}, α_i). The cost of this computational step depends on the size of the high-dimensional CFD model, but is usually relatively low, particularly when the fluid is assumed to be inviscid and the computation is performed on a parallel processor.

2) Evaluate the linearized coupled fluid–structure operators at the above steady-state solution. This step is typically computationally inexpensive. It involves linearizing the governing coupled equations of dynamic equilibrium of the fluid and structural subsystems around the steady-state equilibrium solution computed in step 1 above [11,15].

3) Generate fluid snapshots in the frequency domain and in the complex plane, for a chosen set of reduced frequencies and structural mode shapes [8,11,14]. The cost of this computational step depends on the size of the linearized high-dimensional CFD model, and the numbers of reduced frequencies and structural mode shapes of interest, respectively. These are used to harmonically perturb the steady-state equilibrium solution computed in the first step outlined above in order to generate representative solution snapshots. This third step is, by far, the most computationally intensive one. It requires the solution of a number of large-scale (high-dimensional) linear systems of equations that is proportional to the product of the chosen number of reduced frequencies and chosen number of structural mode shapes.

4) Construct a fluid ROB, $\mathbf{\Phi}(M_{\infty_i}, \alpha_i)$, using the POD method [11]. This step typically involves the computation of a singular value decomposition (SVD) of the snapshot matrix. In this work, the computational cost of the SVD process is reduced by a parallel factorization algorithm.

Further details about the above computational steps can be found in [1,6–8,11–14,16,17], among others. Here, it is emphasized that step 3 can be computationally intensive. For example, it was shown in [11] that for a fighter configuration at Mach $M_\infty = 0.9$, about 90% of the total CPU time associated with the construction of a POD-based fluid ROB is consumed by the computation in the frequency domain of snapshots of the fluid subsystem.

C. Modal-Based Structural ROM

Since in this work only the aerodynamic flight conditions are varied, a single structural ROM is precomputed for the aeroelastic system of interest. This ROM is based on the classical reduction of the finite element (FE) mass, stiffness, and damping matrices of the structural subsystem by N_m natural *dry* modes.

D. Database of ROBs and ROMs

A linearized ROM is attractive because it typically operates in real or near real time. In computational mechanics, it is often constructed by projecting a linearized high-dimensional model onto a ROB. The projection step itself is, in general, computationally inexpensive. However, as highlighted in Sec. II.B for the case of CFD and the POD method, the generation of an appropriate ROB is, in general, computationally expensive as it requires computing a large number of system responses to input excitations using a linearized high-dimensional model. Hence, a ROM is in practice effective only if the computational cost associated with its construction can be quickly amortized over multiple usages of this ROM for different reduced-order analyses. Unfortunately, different reduced-order analyses are typically performed for different parameter values, and most if not all linearized ROM technologies lack robustness with respect to parameter variations. For example, it was shown in [8] that a POD-based fluid ROM constructed at a specific Mach number does not approximate well the dynamics of the fluid for a different freestream Mach number. As a result, a fluid ROM needs to be reconstructed each time an aerodynamic parameter such as the Mach number or angle of attack is changed, which is computationally inefficient, however. Alternatively, one can imagine adapting to new values of aerodynamic parameters such as the freestream Mach number and angle of attack a fluid ROM that was computed for specific values of these parameters. In practice, such an adaptation can be performed by interpolating a given set of fluid ROMs computed for different sets of values of the aerodynamic parameters to rapidly obtain a new fluid ROM for the new desired set of values of these aerodynamic parameters. Such interpolation requires the precomputation and storage in a database of several fluid ROMs. However, to the best of the authors' knowledge, no interpolation method for CFD-based fluid ROMs has been published in the literature and demonstrated for high-speed flows, and no other form parameter adaptation method has been shown to be successful when the parameter of interest is the Mach number and the flow is in the compressible regime. On the

other hand, a promising multivariate interpolation method was recently developed for fluid ROBs [8,12,13]. This method, which is outlined in Sec. II.E below was also recently extended by the authors to the interpolation of structural (but not fluid) ROMs [18]. It is for these reasons that in this work, fluid ROBs rather than fluid ROMs are precomputed for various pairs of flight parameters (M_{∞_i}, α_i) and stored in a database, together with one or many structural ROMs. Nevertheless, once a fluid ROM is assembled by projecting a linearized high-dimensional CFD model onto a fluid ROB, this ROM is also stored in the same database. Furthermore, all fluid ROBs stored in the database are constrained to have the same dimension in order to facilitate their interpolation as explained in the following section.

E. Multivariate Interpolation of Fluid ROBs in a Tangent Space to a Manifold

Interpolation is a natural and attractive idea for generating a fluid ROB at a point (M_∞, α) not represented in the database. However, it is not a straightforward task. Indeed, reduced-order bases generated by the POD method are orthogonal, but the *standard* interpolation of orthogonal vectors does not necessarily produce a set of orthogonal vectors. Similarly, the standard interpolation of a set of bases does not necessarily produce a basis. A set of orthogonal bases is an example of constrained data. In general, standard interpolation does not preserve a constraint, because constrained data does not belong to a *vector space*. For this reason, a *special* interpolation method for ROBs was recently developed [8,12,13]. This interpolation method is based on concepts from differential geometry and guarantees that the outcome of the interpolation of a set of ROBs is a genuine ROB. It is summarized below to keep this paper as self-contained as possible.

Let N_f and N_Φ denote the size of the high-dimensional CFD model and the dimension of the ROB $\mathbf{\Phi}(M_\infty, \alpha)$ (and associated reduced-order model), respectively. Here, N_Φ is fixed independently from the flight parameters M_∞ and α. The subspace $\mathcal{S}(M_\infty, \alpha)$ spanned by $\mathbf{\Phi}(M_\infty, \alpha)$ belongs to the Grassmann manifold $\mathcal{G}(N_\Phi, N_f)$. Hence, $\mathcal{S}$ belongs to a manifold and not to a vector space: therefore, its representative ROB $\mathbf{\Phi}$ cannot be interpolated directly.

As the manifold $\mathcal{G}$ is Riemannian, it admits at each of its points $\mathcal{S}$ a tangent space $\mathcal{T}_\mathcal{S}$ that *is a vector space*: therefore, the representative matrix Γ of a point in $\mathcal{T}_\mathcal{S}$ can be interpolated directly. Now, consider the subspace point $\mathcal{S}_0 = \mathcal{S}(M_{\infty_0}, \alpha_0)$ spanned by the ROB $\mathbf{\Phi}_0 = \mathbf{\Phi}(M_{\infty_0}, \alpha_0)$. At this point, two different mappings can be defined between $\mathcal{G}$ and its tangent space $\mathcal{T}_{\mathcal{S}_0}$ at $\mathcal{S}_0$.

1) The *logarithm* maps points of $\mathcal{G}$ in the neighborhood of $\mathcal{S}_0$ onto the tangent space at this point [see Eqs. (1)].

2) The *exponential* maps the tangent space $\mathcal{T}_{\mathcal{S}_0}$ to the manifold $\mathcal{G}$ itself [see Eqs. (2)].

Both logarithm and exponential objects can be represented by $N_f \times N_\Phi$ real matrices. Therefore, no tangent space needs to be explicitly computed. For example, if $\mathcal{S}_1 = \mathcal{S}(M_{\infty_1}, \alpha_1) \in \mathcal{G}$ is in the neighborhood of $\mathcal{S}_0 \in \mathcal{G}$ and is spanned by the matrix $\mathbf{\Phi}_1 = \mathbf{\Phi}(M_{\infty_1}, \alpha_1)$, the logarithm of $\mathcal{S}_1$ is represented by the matrix given by the second of the following equations:

$$(\mathbf{I} - \mathbf{\Phi}_0\mathbf{\Phi}_0^T)\mathbf{\Phi}_1(\mathbf{\Phi}_0^T\mathbf{\Phi}_1)^{-1} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \quad \text{(thin SVD)}$$
$$\Gamma = \mathbf{U}\tan^{-1}(\mathbf{\Sigma})\mathbf{V}^T \tag{1}$$

If $\chi \in \mathcal{T}_{\mathcal{S}_0}$ is represented by the matrix Γ, the exponential of χ is given by the second of the following equations:

$$\Gamma = \tilde{\mathbf{U}}\,\tilde{\mathbf{\Sigma}}\,\tilde{\mathbf{V}}^T \quad \text{(thin SVD)}$$
$$\mathrm{Exp}_{\mathcal{S}_0}(\chi) = \mathrm{span}\{\mathbf{\Phi}_0\tilde{\mathbf{V}}\cos(\tilde{\mathbf{\Sigma}}) + \tilde{\mathbf{U}}\sin(\tilde{\mathbf{\Sigma}})\} \tag{2}$$

Now consider a precomputed set of N_R ROBs

$$\{\mathbf{\Phi}_i\}_{i=0}^{N_R-1} = \{\mathbf{\Phi}(M_{\infty_i}, \alpha_i)\}_{i=0}^{N_R-1}$$

which have the *same dimension* N_Φ and whose columns span the set of N_R *neighboring* subspaces

$$\{\mathcal{S}_i\}_{i=0}^{N_R-1} = \{\mathcal{S}(M_{\infty_i}, \alpha_i)\}_{i=0}^{N_R-1}$$

of the *same* manifold $\mathcal{G}(N_\Phi, N_f)$. Here, the idea is to generate a new ROB $\mathbf{\Phi}_{N_R}$ for a new pair of flight parameters $(M_{\infty_{N_R}}, \alpha_{N_R})$ by first mapping $\{\mathcal{S}_i\}_{i=0}^{N_R-1}$ onto the tangent space $\mathcal{T}_{\mathcal{S}_i}$ at one of the points $\mathcal{S}_i$ (and therefore computing the logarithms of the ROBs $\{\mathbf{\Phi}_i\}_{i=0}^{N_R-1}$), then interpolating these logarithms to obtain the logarithm of the desired ROB and, finally, computing the exponential of the interpolated logarithm to obtain the desired $\mathbf{\Phi}_{N_R} = \mathbf{\Phi}(M_{\infty_{N_R}}, \alpha_{N_R})$. More specifically, this process can be described by the following four steps:

1) Choose a reference point $\mathcal{S}_{i_0} \in \mathcal{G}(N_\Phi, N_f)$, $i_0 \in \{0, \ldots, N_R - 1\}$.

2) Compute the matrix representations of the logarithms $\{\Gamma_i\}_{i=0}^{N_R-1}$. (Each $\Gamma_i \in \mathbb{R}^{N_f \times N_\Phi}$.)

3) Compute each entry of a matrix $\Gamma_{N_R} \in \mathbb{R}^{N_f \times N_\Phi}$ by interpolating the corresponding entries of the matrices $\Gamma_i \in \mathbb{R}^{N_f \times N_\Phi}$ associated with the data points (M_{∞_i}, α_i). For this purpose, use a standard bivariate interpolation scheme.

4) Compute the exponential of χ_{N_R} represented by Γ_{N_R} to obtain the desired ROB at $(M_{\infty_{N_R}}, \alpha_{N_R})$, $\mathbf{\Phi}_{N_R}$, which verifies $\mathrm{Exp}_{\mathcal{S}_{i_0}}(\chi_{N_R}) = \mathrm{span}\{\mathbf{\Phi}_{N_R}\}$.

The interpolation method in a tangent space to a Grassmann manifold outlined above is graphically depicted in Fig. 1. It was numerically shown in [8] that this method is robust with respect to the choice of the reference point $\mathcal{S}_{i_0}$. For any chosen subset of interpolation data, the logarithm maps can be precomputed and stored in the same database of reduced-order information in order to enable the execution of this interpolation method in real time.

F. Construction of a Fluid ROM for a Partitioned Aeroelastic Analysis

For a specified pair of flight parameters (M_{∞_i}, α_i), an appropriate fluid ROB is either selected from the database if found there, or computed in real time by interpolating an appropriate subset of the fluid ROBs stored in the database using the algorithm outlined in Sec. II.E. Next, the corresponding fluid ROM is rapidly constructed by performing the following steps:

1) Compute the steady-state equilibrium solution of the aeroelastic problem of interest at (M_{∞_i}, α_i). As stated earlier, the cost of this computational step depends on the size of the high-dimensional CFD model, but is usually low, particularly on a parallel processor and for inviscid flows.

2) Evaluate the linearized coupled fluid–structure operators at the above steady-state solution. Again, as stated earlier, this step is typically computationally inexpensive.

3) Project (in the Galerkin sense) the linearized coupled fluid–structure operators computed above onto the ROB $\mathbf{\Phi}(M_{\infty_i}, \alpha_i)$. Typically, this is a computationally inexpensive step that completes the construction of a linearized fluid ROM.

Here, it is noted that a linearized fluid ROM constructed as outlined above includes not only a reduced-order model of the fluid subsystem but also its coupling with the modal-based structural

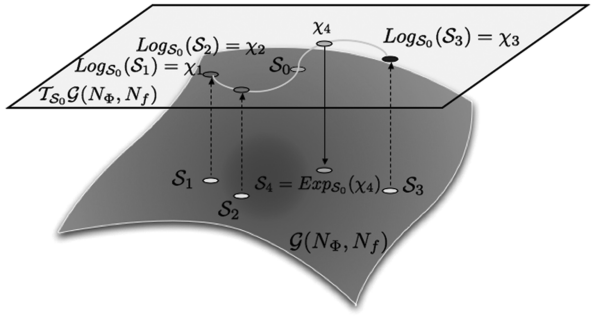


Fig. 1 Schematic description of the interpolation of four subspaces in a tangent space to a Grassmann manifold.

ROM outlined in Sec. II.C (this is the reason why step 2 and step 3 above emphasize the linearized coupled fluid–structure operators instead of the linearized fluid operator only).

G. Assembly and Postprocessing of an Aeroelastic ROM

Once a fluid ROM is constructed for specified values of the flight parameters (M_∞, α) as explained above, the corresponding linearized aeroelastic ROM becomes readily available in partitioned form, because the constructed fluid ROM incorporates the reduced-order form of the coupling operators with the structural ROM. This partitioned form of the linearized aeroelastic ROM can be processed in real time for flutter analysis: 1) either in the frequency domain using a standard eigenvalue-based solution approach or 2) in the time domain using the same state-of-the-art loosely coupled fluid–structure solution procedures [19,20] that were developed for high-dimensional computational models, except that in this case, these algorithms perform in real time because of the lower dimensionality of the aeroelastic model.

III. Application to the Aeroelastic Identification of an F-16 Configuration

The CFD-based aeroelastic reduced-order modeling methodology described in Sec. II has been implemented in the nonlinear aeroelastic code AERO [21,22]. Here, it is demonstrated with the *on-demand* aeroelastic parametric identification of an F-16 Block 40 configuration with clean wings.

The starting points of the structural and fluid computational models are 1) a detailed and undamped FE structural model of the F-16 aircraft built with bar, beam, solid, plate, and shell, metallic, and composite elements with a total of 168,799 degrees of freedom and 2) an unstructured inviscid CFD grid with 403,919 vertices and 2,019,595 unknowns, respectively. These two computational models, which are shown in Fig. 2, are used for the following:

1) Generate the linearized fluid snapshots in the frequency domain, precompute a set of fluid ROBs, and compute the steady-state solutions needed for constructing fluid ROMs.

2) Construct the modal-based structural ROM.

3) Generate reference linearized solutions to the various aeroelastic problems considered in the remainder of this section.

In all cases, the fluid subsystem is assumed to be inviscid and is discretized in space and time by second-order-accurate methods.

To support the desired on-demand mode of computations, a database of reduced-order computational information is first constructed for the chosen F-16 configuration. Next, the domain of this database is discretized into cells, and a local optimal fluid ROB interpolation scheme is assigned to each cell. The emphasis on locality is not only to accelerate computations but also to avoid mixing different flow regimes (for example, subsonic and supersonic). Then any query about the aeroelastic behavior of the aircraft at an operating point $(M_\infty^\star, \alpha^\star)$ not represented in the database is addressed as follows:

1) First, the cell enclosing this operating point is identified.

2) Then its assigned interpolation scheme is applied to generate in real time the fluid ROB at the specified operating point.

3) The corresponding linearized fluid ROM is constructed.

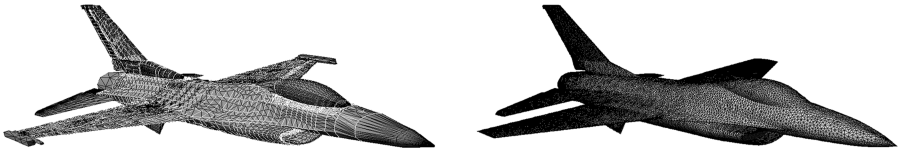
4) The corresponding linearized partitioned aeroelastic ROM is assembled.

5) Finally, this ROM is exploited to respond in real time to the query.

A. Database Setup

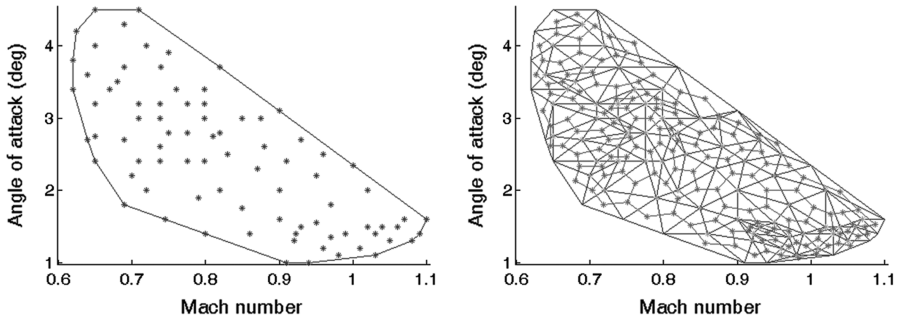
The space freestream Mach number/angle of attack is sampled by 83 pairs of flight conditions (M_{∞_i}, α_i) that are chosen more or less randomly, but with the objective of covering the subsonic, transonic, and supersonic flight regimes and a range of angles of attack (Fig. 3a). It is found that, in general, 99 fluid snapshots [generated as explained in Sec. II.B using nine structural mode shapes of the structure each forced at six different frequencies (including the zero frequency), which generates 54 real parts and 45 imaginary parts of the computed fluid responses] are needed to construct for a given pair of flight conditions a POD-based fluid ROM of any dimension $N_\Phi \leq 99$. Given that for $N_\Phi \leq 99$, the cost of constructing a POD-based fluid ROM is dominated by that of computing the 99 fluid snapshots, it is decided to set $N_\Phi = 90$ for all chosen pairs of flight conditions in order to maximize the accuracy of the anticipated ROB interpolations. The computed fluid ROBs are stored in a database together with a structural ROM constructed by projecting the mass and stiffness matrices of the detailed FE structural model of the F-16 aircraft on nine natural dry modes. (These modes are the nine lowest frequency modes contributing the most to the transient response of the considered aircraft to a suddenly applied gravity load.) Hence, all linearized aeroelastic ROMs based on this database are of dimension 99 $(90 + 9.)$

To demonstrate the on-demand aeroelastic parametric identification of the chosen F-16 configuration using this database, the



a) Detailed FE structural model **b) CFD surface grid**

Fig. 2 High-dimensional aeroelastic computational model of an F-16 Block 40 aircraft.



a) Parameter sampling and parametric domain of interest for aeroelastic predictions (interior of region delimited by a line) **b) Triangulation of the data points and construction of the dual cells**

Fig. 3 F-16 database of fluid ROBs and dual-cell representation.

parametric domain of interest is set to the interior of the convex hull of the database points, as shown in Fig. 3a.

B. Discretization, Decomposition, Training, and Reduction

Next, the space freestream Mach number/angle of attack is triangulated using the data points introduced above (Fig. 3b). A set of dual cells is constructed from the triangles and decomposed into *overlapping* clusters of cells. Within each cluster, two sets of data points are distinguished (Fig. 4): a set of interior data points whose enclosing cells belong to one cluster only, and a set of boundary data points that are shared by at least two clusters, except at the boundary of the convex hull of the database points. The latter set of data points results from the overlapping nature of the decomposition. Its role is to ensure that any interpolation within a cell never leads in practice to an extrapolation (because extrapolations can lead to numerical instabilities and/or inaccuracies). Finally, using concepts from machine learning [23], the database is trained to identify within each dual cell an appropriate local interpolation scheme.

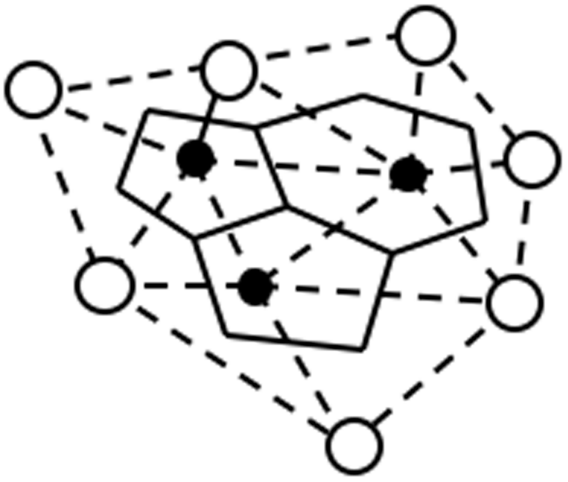


Fig. 4 Schematic description of a cluster of dual cells of the database: the black and white circles denote the interior and boundary data points, respectively. The triangles resulting from the Delaunay triangulation are drawn using dashed lines, and the corresponding dual cells are delimited by full lines.

1. *Triangulation and Dual-Cell Construction*

The discretization of the parametric domain of interest into dual cells containing each one and only one data point is performed in two steps as follows:

1) Perform a Delaunay triangulation for the entire set of data points. The resulting triangles are shown in Fig. 3b.

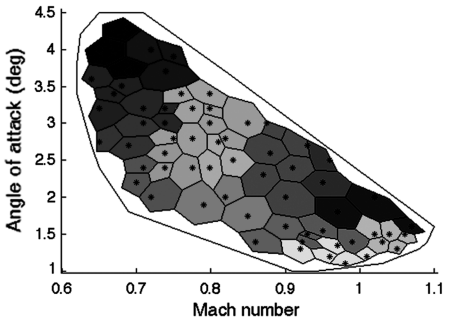
2) Generate the associated dual cells. For each interior data point, the corresponding dual cell is defined as the polygon whose vertices are the centers of gravity of the triangles connected to this data point, as shown in Fig. 4.

It is emphasized here that the triangulation and dual-cell construction are performed to enable the selection of the ROMs to be interpolated but are not part of the interpolation method itself.

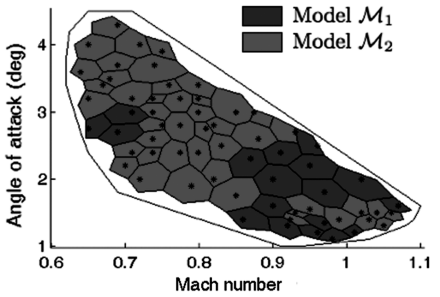
The dual cells associated with the triangulation performed in step 1 above are highlighted in Fig. 3b. Their union covers the interior of the regions of interest in the database.

2. *Clustering and Cross-Validation*

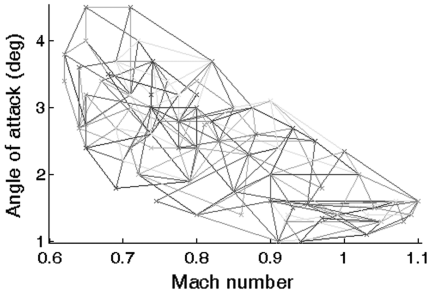
Next, the k-means algorithm [23] is applied to decompose the database (more specifically, its representation by dual cells) into overlapping clusters $\mathcal{C}_j$ with two to five dual cells and nine to 13 data points each (Fig. 5a). Within each cluster $\mathcal{C}_j$, a cross-validation process [23] is applied to select, among a few considered choices, the best local interpolation model. To this effect, and in order to minimize preprocessing time, two different $\mathcal{N}_k$-point interpolation model $\mathcal{M}_k$, $k = 1, 2$, are considered: model $\mathcal{M}_1$ using $\mathcal{N}_1 = 4$ interpolation points of the cluster $\mathcal{C}_j$, and model $\mathcal{M}_2$ using $\mathcal{N}_2 = 5$. Each interpolation model is tested as follows. The interior data points of each cluster $\mathcal{C}_j$ are treated as test points. At each test point, a dozen surrogate fluid ROBs are computed in real time by interpolating fluid ROBs precomputed at $\mathcal{N}_k$ randomly chosen *other* data points of the



a) Clustering of the dual cells (each shade of gray represents a cluster and each dot represents a test point)



b) Retained local interpolation models (each dot represents a test point)



c) Retained data points for local interpolations (for each dual cell, the corresponding set of interpolating data points is shown as a polygon enclosing it)

Fig. 5 Decomposition, training, and reduction of the F-16 database of fluid ROBs.

cluster $\mathcal{C}_j$ containing the test point, and the corresponding dozen of linearized aeroelastic ROMs are constructed. Then the numerical stability and accuracy of each constructed surrogate aeroelastic ROM is assessed by comparing the solution it delivers for a test aeroelastic response problem to that produced by the linearized aeroelastic ROM constructed directly from the fluid ROB available at this data (or test) point. Next, for each test point (M_{∞_i}, α_i) located within $\mathcal{C}_j$, the set of interpolation data points $\mathcal{P}_{ijk}$ (also located within $\mathcal{C}_j$) leading for each interpolation model $\mathcal{M}_k$ to the smallest L_2-norm of the discrepancy in the transient lift, ε_{ijk}, is identified. For each model $\mathcal{M}_k$, the generalized error at the test point (M_{∞_i}, α_i) is estimated as the mean value $\bar{\varepsilon}_{jk}$ of all of the discrepancy values ε_{ijk}. Finally, in each dual cell identified by its center data point (M_{∞_i}, α_i), the interpolation model $\mathcal{M}_k$ with the lowest estimated generalized error $\bar{\varepsilon}_{jk}$ is chosen (Fig. 5b) and assigned the set of interpolation data points $\mathcal{P}_{ijk}$ (Fig. 5c).

It is noted that a byproduct of the training described above is the effective reduction of the original database to the sets of points $\mathcal{P}_{ijk}$ chosen for the local interpolations.

C. On-Demand Aeroelastic Predictions

On-demand predictions of the transient aeroelastic responses of the considered aircraft to initial perturbations of its steady-state aeroelastic equilibrium states are illustrated here for five flight points $\{(M^\star_{\infty_l}, \alpha^\star_l)\}_{l=1}^5$ that are shown in Fig. 6. The $\star$ superscript emphasizes that no information is available in the database at these pairs of flight conditions. Using the aeroelastic computational methodology proposed in this paper, a fast numerical prediction is performed for each of them as follows:

1) Compute the steady-state equilibrium solution of the aeroelastic problem at $(M^\star_{\infty_l}, \alpha^\star_l)$.

2) Evaluate the linearized coupled fluid–structure operators at the above steady-state solution.

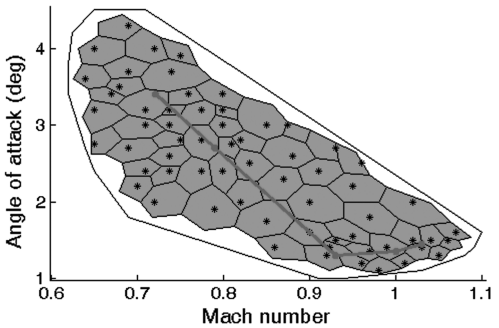


Fig. 6 Flight points not represented in the database for which on-demand aeroelastic predictions are requested.

3) Determine to which dual cell the point $(M^\star_{\infty_l}, \alpha^\star_l)$ belongs and select its local interpolation model.

4) Interpolate the corresponding precomputed logarithm mappings in the tangent space to the Grassmann manifold at the cell center to obtain $\Gamma(M^\star_{\infty_l}, \alpha^\star_l)$.

5) Map $\Gamma(M^\star_{\infty_l}, \alpha^\star_l)$ to $\mathbf{\Phi}^\star_l = \mathbf{\Phi}(M^\star_{\infty_l}, \alpha^\star_l)$ using the exponential map. $\mathbf{\Phi}^\star_l$ is now the interpolated fluid ROB at $(M^\star_{\infty_i}, \alpha^\star_i)$.

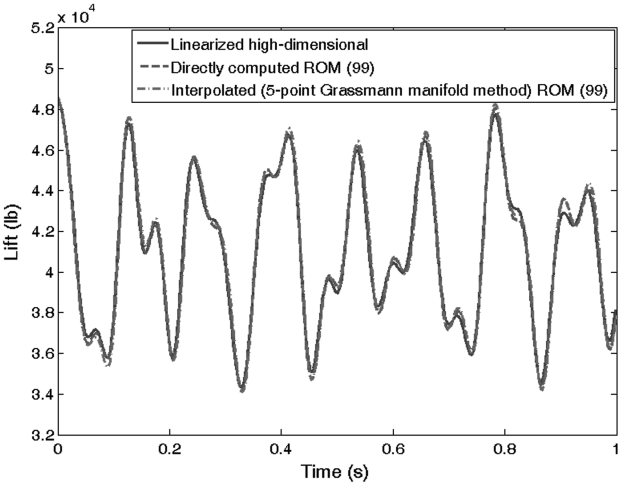


Fig. 8 Aeroelastic predictions at $M^\star_{\infty_2} = 0.790$ and $\alpha^\star_2 = 2.7°$.

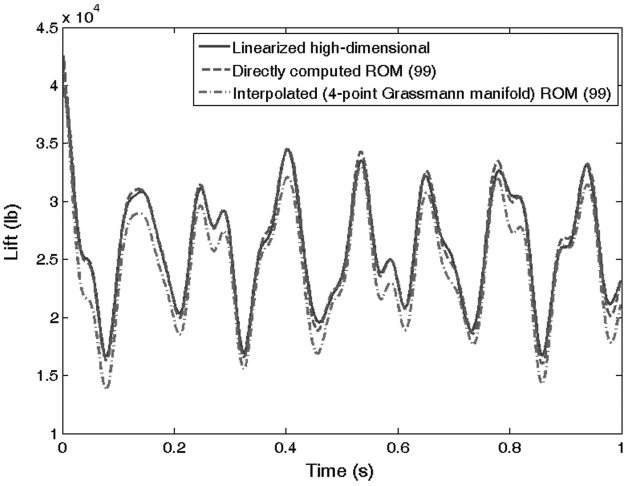


Fig. 9 Aeroelastic predictions at $M^\star_{\infty_3} = 0.930$ and $\alpha^\star_3 = 1.3°$.

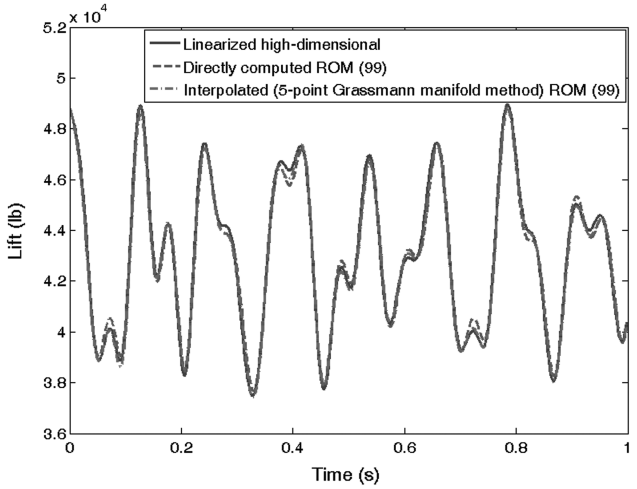


Fig. 7 Aeroelastic predictions at $M^\star_{\infty_1} = 0.721$ and $\alpha^\star_1 = 3.4°$.

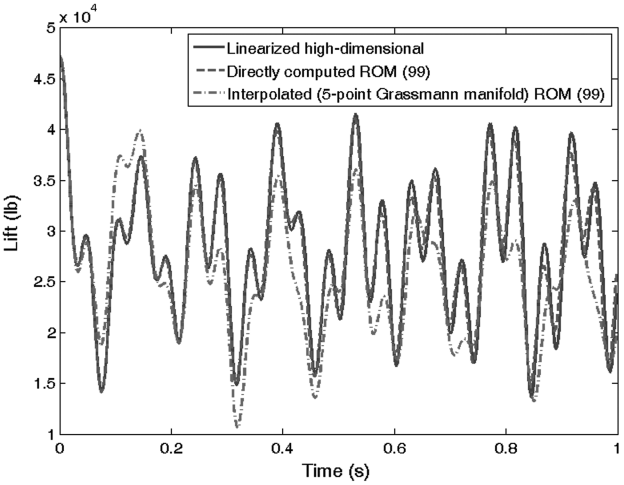


Fig. 10 Aeroelastic predictions at $M^\star_{\infty_4} = 1.000$ and $\alpha^\star_4 = 1.35°$.

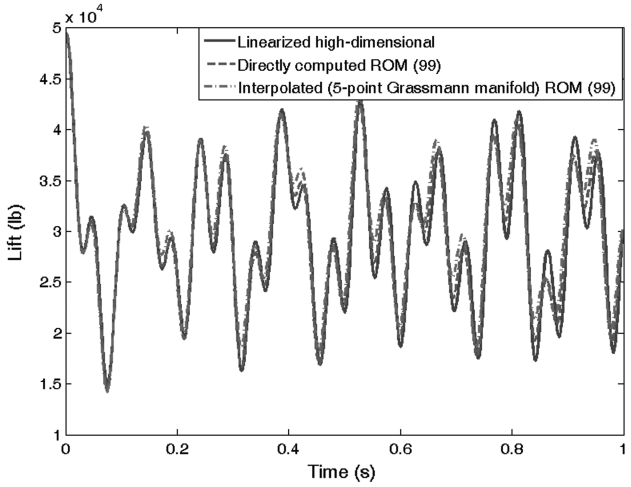


Fig. 11 Aeroelastic predictions at $M^\star_{\infty_5} = 1.030$ and $\alpha^\star_5 = 1.45°$.

6) Construct the fluid ROM at $(M^\star_{\infty_l}, \alpha^\star_l)$ by projecting the linearized fluid–structure operators on the fluid ROB $\mathbf{\Phi}^\star_l$.

7) Assemble the partitioned aeroelastic ROM at $(M^\star_{\infty_l}, \alpha^\star_l)$ and postprocess it in the time domain.

1. Numerical Results

Figures 7–11 compare lift–time histories predicted by the on-demand computational methodology proposed in this paper at the flight points $(M^\star_{\infty_1}, \alpha^\star_1)$, $(M^\star_{\infty_2}, \alpha^\star_2)$, $(M^\star_{\infty_3}, \alpha^\star_3)$, $(M^\star_{\infty_4}, \alpha^\star_4)$ and $(M^\star_{\infty_5}, \alpha^\star_5)$, respectively, to their counterparts obtained using the linearized aeroelastic ROMs directly computed at these flight points and the associated linearized high-dimensional computational models (see the beginning of Sec. III). These time histories correspond to the predicted responses of the aeroelastic system to a perturbation of its equilibrium state by an initial displacement of the structural subsystem. Similar comparisons can be found in [24] for other flight points. (At this point, it is emphasized that it was already shown in [7,11] that for the same F-16 configuration chosen here, a linearized aeroelastic ROM constructed using the POD-based approach adopted in this work reproduces accurately the lift–time histories predicted by its corresponding fully nonlinear high-dimensional model.)

Table 1 Assessment of the numerical accuracy of the proposed on-demand computational methodology

Flight point l	1	2	3	4	5
ε_l	0.41%	0.55%	6.94%	14.8%	3.59%

Table 1 reports for each of the requested flight points $(M^\star_{\infty_l}, \alpha^\star_l)$ the L_2-norm of the relative difference between the transient lift predicted using the on-demand computational model and its counterpart associated with the corresponding high-dimensional linearized computational model, ε_l. All of Figs. 7–11 and Table 1 demonstrate that, except for the flight condition $(M^\star_{\infty_4}, \alpha^\star_4)$, the proposed on-demand computational methodology delivers excellent results. At the sonic point $(M^\star_{\infty_4}, \alpha^\star_4)$, it delivers acceptable results.

2. Computational Costs

Table 2 reports the CPU timings associated with populating one data point of the F-16 database described above. About 89% of the total CPU time (102 mns) is consumed by the generation of 99 fluid snapshots in the frequency domain and the construction of the fluid ROB of dimension 90 using the POD method. All reported computations are shown to scale well when the number of processors is doubled. From these results, the reader can infer that the total CPU time associated with setting up the entire F-16 database is 141.1 h on 32 processors (83×102 mns), and 70.6 h on 64 processors.

Tables 3 and 4 report the computational cost associated with the prediction of the aeroelastic response of the considered F-16 configuration to some initial perturbation of its steady-state aeroelastic equilibrium at a flight point that is not available in the database, for two different computational modeling approaches:

1) A linearized aeroelastic ROM of dimension 99 based on a fluid ROB of dimension 90 that is directly computed at the desired flight point by the POD method.

2) The on-demand computational methodology proposed in this paper that is based on a database of fluid ROBs of dimension 90 and suitable local models of the interpolation method in a tangent space to the Grassmann manifold.

In both cases, after the linearized aeroelastic ROM of dimension 99 is constructed, the aeroelastic response is predicted by time-integrating the initial boundary value problem associated with this ROM for the first 1 s using 500 time steps and a single processor. As indicated by Figs. 7–11, the 1 s time interval is sufficient for identifying the aeroelastic parameters of the considered F-16 configuration at each requested flight point. The reader can observe that the proposed interpolation method accelerates the generation of a fluid ROB by a factor greater than 51 [$(12 + 90 + 0.5)$ mns$/2$ mns]. The reader can also observe that the total CPU time consumed by the proposed computational methodology for this on-demand aeroelastic prediction (not including the initial cost associated with setting up the database) is 15.5 mns. This cost is distributed as follows: 12 mns for computing a steady-state flow at the desired flight point that is not represented in the database, 2.5 mns

Table 2 Computational cost associated with populating one data point of the database using $N_p = 32$ or $N_p = 64$ processors

Computational step	N_p	CPU time	N_p	CPU time
Computation of steady-state equilibrium	32	12 mns	64	6 mns
Generation of 99 fluid snapshots and a fluid ROB (90)	32	90 mns	64	50 mns
Total CPU time	——	102 mns	——	56 mns

Table 3 Computational cost associated with the prediction of the first second of aeroelastic response of an F-16 configuration using an aeroelastic ROM based on a fluid ROB computed directly at the desired flight point and $N_p = 32$ or $N_p = 64$ processors

Computational step	N_p	CPU time	N_p	CPU time
Computation of steady-state equilibrium	32	12 mns	64	6 mns
Generation of 99 fluid snapshots and a fluid ROB (90)	32	90 mns	64	50 mns
Construction of fluid ROM	32	0.5 mn	64	0.25 mn
Assembly and postprocessing of aeroelastic ROM	1	1 mn	1	1 mn
Total CPU time	——	103.5 mns	——	57.25 mns

Table 4 Computational cost associated with the prediction of the first second of aeroelastic response of an F-16 configuration using the on-demand computational methodology proposed in this paper and $N_p = 32$ or N_p=64 processors

Computational step	N_p	CPU time	N_p	CPU time
Computation of steady-state equilibrium	32	12 mns	64	6 mns
Interpolation of fluid ROBs (5 points)	32	2 mns	64	1 mn
Construction of fluid ROM	32	0.5 mn	64	0.25 mn
Assembly and analysis of aeroelastic ROM	1	1 mn	1	1 mn
Total CPU time	——	15.5 mns	——	8.25 mns

only for constructing a fluid ROM at the desired flight point by interpolation, and 1 min only for assembling and analyzing the corresponding aeroelastic ROM for performing the desired aeroelastic prediction. Hence, this total CPU cost is dominated by the cost associated with computing a steady-state flow at the desired flight point, which is not represented in the database.

Furthermore, consider the following:

1) Predicting the aeroelastic response of the same F-16 configuration to the same initial perturbation at the same flight conditions for the same time interval and also using 500 time steps but the linearized high-dimensional computational model produces nearly the same result, but requires 44 min CPU on 32 processors, and 21.5 min on 64 processors.

2) Solving the same aeroelastic problem using the nonlinear high-dimensional aeroelastic computational model produces also almost the same result, but requires sampling the same response time interval into 2000 time steps and consumes 9 h and 32 min on 32 processors and consumes 5 h and 12 min on 64 processors. The computational advantage of the proposed on-demand computational methodology based on a precomputed database of reduced-order information is impressive, and its potential for assisting flutter flight testing is promising.

IV. Conclusions

A new CFD-based computational methodology for fast aeroelastic predictions was presented and demonstrated for a full F-16 Block 40 aircraft configuration. This methodology, which is based on a database of reduced-order fluid bases (ROBs) and reduced-order structural models (ROMs) and on a suitable method of interpolation on a manifold, was shown to greatly reduce computational cost while retaining good accuracy. It enables test operation calls for new last-minute flight configurations and thus paves the way for routine online usage of reduced-order modeling, including during flight testing. Currently, its full potential is limited by the need to recompute a steady-state flow for each desired set of flight conditions that is not represented in the database. Future work will focus on removing this limitation by developing a method for interpolating directly the fluid ROMs instead of the underlying fluid ROBs.

Acknowledgments

This material is based upon work supported partially by the U.S. Air Force Office of Scientific Research under grant F49620-01-1-0129, and partially by the National Science Foundation under grant CNS-0540419. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the U.S. Air Force Office of Scientific Research or the National Science Foundation.

References

[1] Hall, K. C., Thomas, J. P., and Dowell, E. H., "Proper Orthogonal Decomposition Technique for Transonic Unsteady Aerodynamic Flows," *AIAA Journal*, Vol. 38, No. 10, Oct. 2000, pp. 1853–1862. doi:10.2514/2.867

[2] Epureanu, B. I., "A Parametric Analysis of Reduced Order Models of Viscous Flows in Turbomachinery," *Journal of Fluids and Structures*, Vol. 17, 2003, pp. 971–982. doi:10.1016/S0889-9746(03)00044-6

[3] Lieu, T., and Lesoinne, M., "Parameter Adaptation of Reduced Order Models for Three-Dimensional Flutter Analysis," AIAA Paper 2004-0888, 2004.

[4] Thomas, J. P., Dowell, E. H., and Hall, K. C., "Static/Dynamic Correction Approach for Reduced-Order Modeling of Unsteady Dynamics," *Journal of Aircraft*, Vol. 43, No. 4, 2006, pp. 865–878. doi:10.2514/1.12349

[5] Silva, W. A., Vatsa, V. N., and Biedron, R. T., "Development of Unsteady Aerodynamic and Aeroelastic Reduced-Order Models Using the FUN3D Code," International Forum on Aeroelasticity and Structural Dynamics, Seattle, WA, Paper 2009-030, June 22–25 2009.

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

[7] Lieu, T., and Farhat, C., "Adaptation of Aeroelastic Reduced-Order Models and Application to an F-16 Configuration," *AIAA Journal*, Vol. 45, No. 6, 2007, pp. 1244–1257. doi:10.2514/1.24512

[8] Amsallem, D., and Farhat, C., "Interpolation Method for Adapting Reduced-Order Models and Application to Aeroelasticity," *AIAA Journal*, Vol. 46, No. 7, 2008, pp. 1803–1813. doi:10.2514/1.35374

[9] Hu, P., Bodson, M., and Brenner, M., "Towards Real-Time Simulation of Aeroservoelastic Dynamics for a Flight Vehicle from Subsonic to Hypersonic Regime," AIAA Atmospheric Flight Mechanics Conference and Exhibit, AIAA Paper 2008-6375, Honolulu, HI, Aug. 18–21 2008.

[10] Lieu, T., "Adaptation of Reduced Order Models for Applications in Aeroelasticity," Ph.D. Thesis, University of Colorado at Boulder, Boulder, CO, 2004.

[11] Lieu, T., Farhat, C., and Lesoinne, M., "Reduced-Order Fluid/Structure Modeling of a Complete Aircraft Configuration," *Computer Methods in Applied Mechanics and Engineering*, Vol. 195, Nos. 41–43, 2006, pp. 5730–5742. doi:10.1016/j.cma.2005.08.026

[12] Amsallem, D., Farhat, C., and Lieu, T., "Aeroelastic Analysis of F-16 and F-18/A Configurations Using Adapted CFD-Based Reduced-Order Models," 48th Structures, Structural Dynamics, and Materials Conference, AIAA Paper 2007-2364, Honolulu, HI, April 23–26 2007.

[13] Amsallem, D., Farhat, C., and Lieu, T., "High-Order Interpolation of Reduced-Order Models for Near Real-Time Aeroelastic Prediction," International Forum on Aeroelasticity and Structural Dynamics, Paper IF-081, Stockholm, June 18–20 2007.

[14] Lieu, T., Farhat, C., and Lesoinne, M., "POD-based Aeroelastic Analysis of a Complete F-16 Configuration: ROM Adaptation and Demonstration," AIAA Paper 2005-2295, 2005.

[15] Lesoinne, M., Sarkis, M., Hetmaniuk, U., and Farhat, C., "A Linearized Method for the Frequency Analysis of Three-Dimensional Fluid/Structure Interaction Problems in All Flow Regimes," *Computer Methods in Applied Mechanics and Engineering*, Vol. 190, Nos. 24–25, 2001, pp. 3121–3146. doi:10.1016/S0045-7825(00)00385-6

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

[17] Lieu, T., and Farhat, C., "Adaptation of POD-based Aeroelastic ROMs for Varying Mach Number and Angle of Attack: Application to a Complete F-16 Configuration," AIAA Paper 2005-7666, 2005.

[18] Amsallem, D., Cortial, J., Carlberg, K., and Farhat, C., "A Method for Interpolating on Manifolds Structural Dynamics Reduced-order Models," *International Journal for Numerical Methods in Engineering*, Vol. 80, No. 9, 2009, pp. 1241–1258.

doi:10.1002/nme.2681

[19] Lesoinne, M., and Farhat, C., "A Higher-Order Subiteration Free Staggered Algorithm for Nonlinear Transient Aeroelastic Problems," *AIAA Journal*, Vol. 36, No. 9, 1998, pp. 1754–1756. doi:10.2514/2.7555

[20] Farhat, C., van der Zee, G., and Geuzaine, P., "Provably Second-Order Time-Accurate Loosely-Coupled Solution Algorithms for Transient Nonlinear Computational Aeroelasticity," *Computer Methods in Applied Mechanics and Engineering*, Vol. 195, Nos. 17–18, 2006, pp. 1973–2001. doi:10.1016/j.cma.2004.11.031

[21] Geuzaine, P., Brown, G., Harris, C., and Farhat, C., "Aeroelastic Dynamic Analysis of a Full F-16 Configuration for Various Flight Conditions," *AIAA Journal*, Vol. 41, No. 3, 2003, pp. 363–371. doi:10.2514/2.1975

[22] Farhat, C., Geuzaine, P., and Brown, G., "Application of a Three-Field Nonlinear Fluid-Structure Formulation to the Prediction of the Aeroelastic Parameters of an F-16 Fighter," *Computers and Fluids*, Vol. 32, No. 1, 2003, pp. 3–29. doi:10.1016/S0045-7930(01)00104-9

[23] Hastie, T., Tibshirani, R., and Friedman, J., *The Elements of Statistical Learning*, Springer, New York, 2009.

[24] Amsallem, D., Cortial, J., and Farhat, C., "On-Demand CFD-Based Aeroelastic Predictions Using a Database of Reduced-Order Bases and Models," 47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition, AIAA Paper 2009-800, Orlando, FL, Jan. 5–8 2009.

E. Livne
Associate Editor