Flutter Prediction Using Reduced-Order Modeling with Error Estimation

Brandon M. Lowe [*] and David W. Zingg [†]
Institute for Aerospace Studies, University of Toronto,
4925 Dufferin St, Toronto, Ontario, M3H 5T6, Canada

This paper presents a methodology for dynamic aeroelastic analysis of aircraft based on model order reduction with error estimation. A projection-based model order reduction approach is used to create an aerodynamic reduced-order model (ROM) which is coupled to a structural model to create an aeroelastic ROM. The governing aerodynamic equations are the linearized semi-discrete Euler equations. Flutter analysis is conducted by analyzing the eigenvalues of the aeroelastic ROM. A dual-weighted residual-based error estimator is presented which approximates the error in the eigenvalues obtained from the reduced eigenproblem relative to the eigenvalues from the high-dimensional aeroelastic model. The error estimator thus allows for the construction of aeroelastic ROMs with select eigenvalues that satisfy a user-prescribed accuracy. The aerodynamic ROM is constructed using approximate high-dimensional aeroelastic eigenvectors computed using the two-sided Jacobi-Davidson algorithm. Dynamic aeroelastic analyses are presented for the NACA 64A010 Isogai case, the AGARD 445.6 wing model, and a NACA 0012 benchmark case. The error estimator is shown to have good agreement with the exact error. For the test cases presented in this work, the cost of computing the flutter point at a given Mach number is equivalent to the cost of approximately 5 to 12 steady nonlinear flow evaluations of the high-dimensional Euler equations.

Nomenclature

$\mathbb{A}$	=	aeroelastic state matrix
A_{a}, $B_{\mathrm{a},1}$, $B_{\mathrm{a},2}$, E_{a}	=	linearized aerodynamic model matrices
A_{s}, C_{s}	=	linearized structural model matrices
a	=	nondimensional distance from elastic axis to midchord
b	=	semichord
$\boldsymbol{f}_{\mathrm{a}}$	=	aerodynamic forces

[*] PhD Candidate, University of Toronto, brandon.lowe@mail.utoronto.ca

[†] Distinguished Professor of Computational Aerodynamics and Sustainable Aviation, Director, Centre for Research in Sustainable Aviation, dwz@oddjob.utias.utoronto.ca

This work is associated to AIAA Paper 2021-2547, presented at AIAA Aviation 2021 Forum, August 4, Virtual Event.

$\boldsymbol{f}_\mathrm{s}$	=	generalized forces applied to structure
J	=	diagonal matrix of metric Jacobians
K_m	=	mesh movement matrix
$\mathbb{M}$	=	aeroelastic mass matrix
$\boldsymbol{R}_\mathrm{a}$	=	semi-discrete Euler equation residual vector
$\boldsymbol{r}_\mathrm{ROM}$	=	aerodynamic reduced-order model residual
$\boldsymbol{r}$	=	aeroelastic eigenproblem residual
s	=	thermodynamic entropy
T_d, T_f	=	displacement and force transfer matrices
t_a, t_s	=	time variable for aerodynamic and structural models
$\boldsymbol{u}$	=	aeroelastic state vector
$\boldsymbol{u}_\mathrm{a}$	=	aerodynamic state vector
$\boldsymbol{u}_\mathrm{s}$	=	generalized structural displacement vector
V, V_f	=	speed index $\left(V = \frac{V_\infty}{b\,\omega_\alpha\sqrt{\mu}}\right)$, and flutter speed index
V_∞	=	freestream speed
$\boldsymbol{v}$	=	right aeroelastic eigenvector
$\boldsymbol{v}^\mathrm{du}$	=	left aeroelastic eigenvector
W	=	weighted inner product matrix
$\boldsymbol{x}_\mathrm{a,surf}$, $\boldsymbol{x}_\mathrm{a,vol}$	=	aerodynamic surface and volume node coordinates
x_α	=	nondimensional distance from elastic axis to center of gravity
η, η_f	=	eigenvalue error estimator and flutter speed index error estimator
θ	=	Rayleigh quotient
λ	=	eigenvalue
μ	=	mass ratio
ρ	=	density
σ_f, σ_t	=	force and time scaling factors
σ^err, σ^diff, σ^corr	=	eigenvalue tracking metrics
$\boldsymbol{\Phi}$, $\boldsymbol{\Psi}$	=	primal reduced basis and test basis
$\boldsymbol{\Phi}^\mathrm{du}$, $\boldsymbol{\Psi}^\mathrm{du}$	=	dual reduced basis and test basis
$\boldsymbol{\phi}$	=	reduced basis vector
ω, ω_α	=	frequency, and uncoupled torsional frequency

Subscripts

a	=	aerodynamic quantity
s	=	structural quantity
tot	=	total quantity (steady-state + fluctuation)
∞	=	freestream quantity
0	=	steady-state quantity

Superscripts and Symbols

du	=	dual problem quantity
$\sim$	=	reduced-order model quantity
$\cdot$	=	time derivative

I. Introduction

In recent years, a strong emphasis has been placed on the need for increased aircraft fuel efficiency. Efficient designs can be conceived with the aid of aerodynamic shape optimization and multidisciplinary optimization tools based on high-fidelity computational fluid dynamics (CFD) [1]. However, the resulting unconventional aircraft are typically designed to fly at transonic Mach numbers, where a dip in the flutter boundary typically occurs due to the presence and motion of shock waves [2].

Linear aerodynamic methods, such as the doublet lattice method, have been widely used for flutter predictions in the past due to their low computational cost. However, these methods fail to model nonlinear flow features accurately in the transonic regime [3]. This lack of accuracy has led to overly conservative aircraft designs and hence reduced fuel efficiency [4]. Alternatively, time-dependent high-dimensional CFD simulations present a means to model aerodynamics accurately for dynamic aeroelastic analysis. However, due to the unsteady nature of the problem, the large number of degrees of freedom, and the number of flight conditions to consider, the use of unsteady CFD for time-dependent flutter predictions in a design context remains intractable. As a result, numerous CFD-based alternatives have been proposed for flutter predictions. Historically, CFD simulations have been used for the computation of generalized aerodynamic forces, which are incorporated into a traditional flutter analysis tool such as the *p*-*k* method [5–7]. Hopf point calculation methods have been used to obtain the flutter point of CFD-based aeroelastic systems [8–10]. Timme *et al*. [11] also presented a Schur complement approach to track aeroelastic eigenvalues, with the use of a Kriging approximation trained on CFD-based solutions to approximate aerodynamic terms. The harmonic balance method [12] has seen numerous applications for the computation of flutter and limit cycle oscillation (LCO) prediction [13–17]. Additionally, a number of researchers have applied CFD-based corrections or calibrations to low-fidelity models in order to obtain more accurate flutter predictions [18–21].

Model order reduction methods have also been widely applied to reduce the computational expense of CFD-based flutter analyses. The term encompasses a wide breadth of methods, all with the underlying goal of approximating the solution of a high-dimensional model (HDM) in a fraction of the computational time. In the case of flutter analysis, the CFD model generally constitutes the HDM, though occasionally the coupled CFD-structural model is used. Here we do not provide an exhaustive survey of the field, but we describe a few notable applications of reduced-order models (ROMs) for aeroelastic analyses. Model order reduction methodologies can be classified into two categories: data-driven ROMs and projection-based ROMs. We define data-driven model order reduction as any nonintrusive approach which solely relies on input and output data for the construction of the ROM, essentially treating the HDM as a "black-box". In this category, numerous researchers have applied Kriging approximations to model aerodynamic forces [11, 22, 23] and to directly model the flutter surface [24]. The Eigensystem Realization Algorithm [25] has been used with CFD-based aerodynamic force responses to construct low-order aerodynamic models for flutter prediction [26–28]. Autoregressive-based models have seen numerous applications in flutter analysis [29–32]. Additionally, artificial neural networks have also been applied to flutter analysis [33], though their use in nonlinear modeling has made them more popular for LCO prediction [34–36].

In contrast to data-driven approaches, projection-based model order reduction is intrusive because it requires access to the HDM's operators in order to project them onto a subspace spanned by a reduced basis. Though this approach requires more development time, it possesses significant advantages such as an explicit relation between the reduced states and the original system states, and the potential for deriving error estimation and error bounds [37]. In early work, Hall [38] suggested using truncated sets of aerodynamic eigenvectors as reduced bases, though the computational costs of obtaining the eigenvectors limited the application of this approach. Romanowski [39] presented the application of proper orthogonal decomposition (POD) [40, 41] for the construction of reduced bases for aerodynamic modeling. POD subsequently became a common method for projection-based aerodynamic ROMs, with researchers applying it towards flutter analysis [42–45], LCO predictions [46], and the dynamic aeroelastic analysis of a complete aircraft [47]. Previously, the authors of this paper have also presented a projection-based ROM approach for flutter predictions [48]. In this previous work, an aerodynamic ROM was constructed using a reduced basis constructed by applying POD to snapshots obtained by exciting the structural states during a single unsteady flow solve. The aerodynamic ROM was then coupled to the structural model, and the eigenvalues of the resulting aeroelastic ROM were analyzed to predict the onset of flutter.

All of the aforementioned projection-based ROM approaches are capable of predicting flutter; however none are capable of estimating the accuracy of their flutter prediction relative to the original aeroelastic HDM. The objective of this paper is to present a projection-based ROM approach for aeroelastic analysis with an error estimator capable of providing a user-prescribed level of accuracy for flutter predictions relative to the aeroelastic HDM. Dynamic aeroelastic analysis is performed by analyzing the eigenvalues of the aeroelastic ROM. The dual-weighted residual (DWR) method [49] is used to provide an error estimate for the eigenvalues. The error estimator provides the user with a measure of accuracy for the approximated aeroelastic eigenvalues and also serves as an indicator to determine at which flight conditions to update the reduced basis. This leads to an automated ROM training procedure for which the user needs only to supply some parameters for initial snapshots, the applicable flight conditions, and the desired level of accuracy in the eigenvalues. Furthermore, an extension of the eigenvalue error estimator is presented to obtain approximate errors for the predicted flutter point.

It has been shown that the DWR method provides effective error estimation for aerodynamic problems [50] as well as eigenvalue problems [51]. The DWR method requires the solution to a dual (or adjoint) problem in a subspace not spanned by the reduced basis used to create the original aerodynamic ROM. We approximate dual solutions with a second aerodynamic ROM created using a reduced basis trained on dual solutions. As will be discussed in Section IV, the dual solution to the aeroelastic eigenproblem is in fact the left eigenpair of the system. Thus, we denote the aeroelastic ROM used for the main aeroelastic analysis as the *primal ROM*, and the aeroelastic ROM used for the error analysis as the *dual ROM*.

Similar to the work by Lowe and Zingg [48], we rely on the linearized semi-discrete Euler equations as our high-dimensional aerodynamic system; however the reduced basis is now constructed using approximate eigenvectors of the aeroelastic HDM obtained using the two-sided refined Jacobi-Davidson algorithm [52–54]. Our use of eigenvectors and the two-sided Jacobi-Davidson algorithm for model order reduction is similar to the work presented by Benner *et al.* [55]. However, in our work, the left eigenvectors are not used for the test basis in the construction of the ROM, but are rather used to construct the dual ROM for error estimation. We also note that the methodology presented herein is not limited to the use of the linearized Euler equations, and can be extended to the linearized RANS equations.

The remainder of this paper is divided into the following sections. In Section II, we present the high-dimensional equations used to model aeroelastic behavior, including the structural and aerodynamic models. Section III describes the model order reduction approach used to create the primal aeroelastic ROM. In Section IV, we give an overview of the error estimator derivation, and describe the procedure to create the dual aeroelastic ROM. Section V presents our methodology for ROM-based flutter analysis with error estimation. Finally, Section VI contains the results obtained using the methodology derived in this paper.

II. High-Dimensional Semi-discrete Aeroelastic Model

This section presents the high-dimensional semi-discrete equations used to model dynamic aeroelastic behavior. An overview of the structural model is first presented. Subsequently, the linearized semi-discrete Euler equations are derived and used to form the aerodynamic model. Finally, the structural and aerodynamic models are coupled to form the high-dimensional monolithic aeroelastic model. For ease of presentation, certain quantities associated to the aerodynamic model are indicated with the subscript a, whereas those for the structural models are presented with the subscript s.

A. Structural Model

For dynamic aeroelastic analysis, structural states are commonly expanded into a modal series consisting of the summation of free vibration modes weighted by time-dependent generalized displacements. Neglecting contributions from damping, one may write the resulting equations of motion as follows:

$$\frac{d^2\boldsymbol{u}_{\mathrm{s,tot}}}{dt_{\mathrm{s}}^2} + \Omega_{\mathrm{s}}\boldsymbol{u}_{\mathrm{s,tot}} = \boldsymbol{f}_{\mathrm{s,tot}} \tag{1}$$

where t_{s} is the time variable for the structural model, $\boldsymbol{u}_{\mathrm{s,tot}}$ is the vector of generalized displacements, $\boldsymbol{f}_{\mathrm{s,tot}}$ is the vector of generalized forces applied to the structural grid, and Ω_{s} is a diagonal matrix of squared natural frequencies. We wish to linearize equation (1) about a steady-state solution. The relevant states are expanded as:

$$\boldsymbol{u}_{\mathrm{s,tot}} = \boldsymbol{u}_{\mathrm{s,0}} + \boldsymbol{u}_{\mathrm{s}}, \qquad \boldsymbol{u}_{\mathrm{a,tot}} = \boldsymbol{u}_{\mathrm{a,0}} + \boldsymbol{u}_{\mathrm{a}}, \tag{2}$$

where $\boldsymbol{u}_{\mathrm{a,tot}}$ is the vector of aerodynamic states, and steady-state quantities are denoted with the subscript 0. Interest lies in modeling the state fluctuations $\boldsymbol{u}_{\mathrm{s}}$ and $\boldsymbol{u}_{\mathrm{a}}$, and their impact on the fluctuations of the applied forces: $\boldsymbol{f}_{\mathrm{s}} = \boldsymbol{f}_{\mathrm{s,tot}} - \boldsymbol{f}_{\mathrm{s,0}}$.

Forces are computed on the aerodynamic grid, which does not correspond to the structural grid. Additionally, the aerodynamic equations are typically nondimensionalized, and thus applied forces require a dimensionalization factor. We introduce the force scaling factor σ_{f}, the force transfer matrix T_{f}, and the displacement transfer matrix T_{d}, such that

$$\boldsymbol{f}_{\mathrm{s}} = \sigma_{\mathrm{f}}T_{\mathrm{f}}\boldsymbol{f}_{\mathrm{a}}, \qquad \boldsymbol{x}_{\mathrm{a,surf}} = T_{\mathrm{d}}\boldsymbol{u}_{\mathrm{s}}, \tag{3}$$

where $\boldsymbol{f}_{\mathrm{a}}$ is the vector of aerodynamic force fluctuations computed on the aerodynamic surface grid, and $\boldsymbol{x}_{\mathrm{a,surf}}$ is the vector of aerodynamic surface grid node coordinate fluctuations. The transfer matrices T_{f} and T_{d} may be computed in a number of ways. In this work, we use uniform plunge and angular displacements (with a small angle approximation) for the two degree of freedom structural model, and Harder and Desmarais's thin plate spline interpolation [56] for the AGARD 445.6 wing model.

Expanding the fluctuations of the force into a Taylor series and truncating high-order terms gives,

$$\boldsymbol{f}_{\mathrm{s}} = \sigma_{\mathrm{f}}T_{\mathrm{f}}\boldsymbol{f}_{\mathrm{a}} \approx \sigma_{\mathrm{f}}T_{\mathrm{f}}\left(\frac{\partial \boldsymbol{f}_{\mathrm{a}}}{\partial \boldsymbol{x}_{\mathrm{a,surf}}}T_{\mathrm{d}}\boldsymbol{u}_{\mathrm{s}} + \frac{\partial \boldsymbol{f}_{\mathrm{a}}}{\partial \boldsymbol{u}_{\mathrm{a}}}\boldsymbol{u}_{\mathrm{a}}\right). \tag{4}$$

The linearized structural model thus becomes,

$$\frac{d^2\boldsymbol{u}_\mathrm{s}}{dt_\mathrm{s}^2} = A_\mathrm{s}\boldsymbol{u}_\mathrm{s} + C_s\boldsymbol{u}_\mathrm{a}, \tag{5}$$

where:

$$A_\mathrm{s} = \sigma_\mathrm{f}T_\mathrm{f}\frac{\partial \boldsymbol{f}_\mathrm{a}}{\partial \boldsymbol{x}_\mathrm{a,surf}}T_\mathrm{d} - \Omega_\mathrm{s}, \qquad C_s = \sigma_\mathrm{f}T_\mathrm{f}\frac{\partial \boldsymbol{f}_\mathrm{a}}{\partial \boldsymbol{u}_\mathrm{a}}.$$

B. Aerodynamic Model

The governing aerodynamic equations used for dynamic aeroelastic analysis are the linearized semi-discrete Euler equations. Before linearization, the Euler equations are discretized in space using a second-order summation-by-parts approach, and are then put into an arbitrary Lagrangian-Eulerian form. This leads to the following set of ordinary differential equations:

$$\frac{dJ^{-1}\boldsymbol{u}_\mathrm{a,tot}}{dt_\mathrm{a}} = \boldsymbol{R}_\mathrm{a}\left(\boldsymbol{u}_\mathrm{a,tot}(t_\mathrm{a}), \boldsymbol{x}_\mathrm{a,vol,tot}(t_\mathrm{a}), \dot{\boldsymbol{x}}_\mathrm{a,vol,tot}(t_\mathrm{a})\right). \tag{6}$$

Here $\boldsymbol{x}_\mathrm{a,vol,tot}$ and $\dot{\boldsymbol{x}}_\mathrm{a,vol,tot}$ are the vectors of volume grid node coordinates and velocities, respectively. Additionally, t_a is the time variable for the aerodynamic model, $\boldsymbol{R}_\mathrm{a}$ is the residual vector, and J^{-1} is a diagonal matrix of inverse metric Jacobians of the transformation to curvilinear coordinates. In order to preserve a uniform flow in the presence of a deforming grid, the evolution of the J^{-1} is governed by the Geometric Conservation Law [57].

The linearized semi-discrete Euler equations are obtained by assuming the flow states, grid node coordinates, and grid node velocities can be represented as linear fluctuations about a nonlinear steady state:

$$\boldsymbol{u}_\mathrm{a,tot} = \boldsymbol{u}_\mathrm{a,0} + \boldsymbol{u}_\mathrm{a}, \qquad \boldsymbol{x}_\mathrm{a,vol,tot} = \boldsymbol{x}_\mathrm{a,vol,0} + \boldsymbol{x}_\mathrm{a,vol}, \qquad \dot{\boldsymbol{x}}_\mathrm{a,vol,tot} = \dot{\boldsymbol{x}}_\mathrm{a,vol}. \tag{7}$$

Steady-state grid node velocities are assumed to be zero. The resulting linearized semi-discrete Euler equations are:

$$J_0^{-1}\frac{d\boldsymbol{u}_\mathrm{a}}{dt_\mathrm{a}} + \frac{\partial \boldsymbol{R}_\mathrm{a}}{\partial \boldsymbol{u}_\mathrm{a}}\boldsymbol{u}_\mathrm{a} + \frac{\partial \boldsymbol{R}_\mathrm{a}}{\partial \boldsymbol{x}_\mathrm{a,vol}}K_\mathrm{m}\boldsymbol{x}_\mathrm{a,surf} + \left(\frac{\partial}{\partial \dot{\boldsymbol{x}}_\mathrm{a,vol}}\left(\frac{dJ^{-1}}{dt_\mathrm{a}}\right)\boldsymbol{u}_\mathrm{a,0} + \frac{\partial \boldsymbol{R}_\mathrm{a}}{\partial \dot{\boldsymbol{x}}_\mathrm{a,vol}}\right)K_\mathrm{m}\dot{\boldsymbol{x}}_\mathrm{a,surf} = \boldsymbol{0}, \tag{8}$$

where K_m is a linear mesh movement matrix such that:

$$\boldsymbol{x}_\mathrm{a,vol} = K_\mathrm{m}\boldsymbol{x}_\mathrm{a,surf}, \qquad \dot{\boldsymbol{x}}_\mathrm{a,vol} = K_\mathrm{m}\dot{\boldsymbol{x}}_\mathrm{a,surf}. \tag{9}$$

Here $\boldsymbol{x}_\mathrm{a,surf}$ and $\dot{\boldsymbol{x}}_\mathrm{a,surf}$ are the vectors of aerodynamic surface grid node coordinate fluctuations and velocities. For further details on the derivation of the linearized Euler equations, see Lowe and Zingg [48].

The aerodynamic model is nondimensionalized in time differently than the structural model. To address this, the following time scaling factor is introduced:

$$t_{\mathrm{s}} = \sigma_{\mathrm{t}} t_{\mathrm{a}}. \tag{10}$$

Additionally, we must transfer displacements from the structural grid to the aerodynamic surface grid. Here, for the time scaling, we explicitly state the aerodynamic surface velocity vector as $\dot{\boldsymbol{x}}_{\mathrm{a,surf}} = \frac{d\boldsymbol{x}_{\mathrm{a,surf}}}{dt_{\mathrm{a}}}$. Applying the displacement transfer matrix T_{d}, we obtain:

$$\boldsymbol{x}_{\mathrm{a,surf}} = T_{\mathrm{d}} \boldsymbol{u}_{\mathrm{s}}, \qquad \frac{d\boldsymbol{x}_{\mathrm{a,surf}}}{dt_{\mathrm{a}}} = \sigma_{\mathrm{t}} T_{\mathrm{d}} \frac{d\boldsymbol{u}_{\mathrm{s}}}{dt_{\mathrm{s}}}, \tag{11}$$

Introducing equations (10) and (11) into (8), the aerodynamic model is given by,

$$E_{\mathrm{a}} \frac{d\boldsymbol{u}_{\mathrm{a}}}{dt_{\mathrm{s}}} = A_{\mathrm{a}} \boldsymbol{u}_{\mathrm{a}} + B_{\mathrm{a},1} \boldsymbol{u}_{\mathrm{s}} + B_{\mathrm{a},2} \frac{d\boldsymbol{u}_{\mathrm{s}}}{dt_{\mathrm{s}}}, \tag{12}$$

where the system matrices are:

$$E_{\mathrm{a}} = \sigma_{\mathrm{t}} J_0^{-1}, \qquad A_{\mathrm{a}} = \frac{\partial \boldsymbol{R}_{\mathrm{a}}}{\partial \boldsymbol{u}_{\mathrm{a}}}, \qquad B_{\mathrm{a},1} = \frac{\partial \boldsymbol{R}_{\mathrm{a}}}{\partial \boldsymbol{x}_{\mathrm{a,vol}}} K_{\mathrm{m}} T_{\mathrm{d}}, \qquad B_{\mathrm{a},2} = \sigma_{\mathrm{t}} \left(\frac{\partial}{\partial \dot{\boldsymbol{x}}_{\mathrm{a,vol}}} \left(\frac{dJ^{-1}}{dt_{\mathrm{a}}} \right) \boldsymbol{u}_{\mathrm{a},0} + \frac{\partial \boldsymbol{R}_{\mathrm{a}}}{\partial \dot{\boldsymbol{x}}_{\mathrm{a,vol}}} \right) K_{\mathrm{m}} T_{\mathrm{d}}.$$

C. Monolithic Aeroelastic Model

We wish to couple the linear structural and aerodynamic models into a single monolithic aeroelastic system. Introducing the notation,

$$\frac{d\boldsymbol{u}_{\mathrm{s}}}{dt_{\mathrm{s}}} = \dot{\boldsymbol{u}}_{\mathrm{s}}, \tag{13}$$

we write equations (5), (12), and (13) into a complete set of aeroelastic equations,

$$\begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & E_{\mathrm{a}} \end{bmatrix} \frac{d}{dt_{\mathrm{s}}} \begin{bmatrix} \dot{\boldsymbol{u}}_{\mathrm{s}} \\ \boldsymbol{u}_{\mathrm{s}} \\ \boldsymbol{u}_{\mathrm{a}} \end{bmatrix} = \begin{bmatrix} 0 & A_{\mathrm{s}} & C_{\mathrm{s}} \\ I & 0 & 0 \\ B_{\mathrm{a},2} & B_{\mathrm{a},1} & A_{\mathrm{a}} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{u}}_{\mathrm{s}} \\ \boldsymbol{u}_{\mathrm{s}} \\ \boldsymbol{u}_{\mathrm{a}} \end{bmatrix}, \tag{14}$$

where I is the identity matrix. In a more convenient form, we write this as

$$\mathbb{M} \frac{d\boldsymbol{u}}{dt_{\mathrm{s}}} = \mathbb{A} \boldsymbol{u}, \tag{15}$$

where $\boldsymbol{u}^T = \left[\dot{\boldsymbol{u}}_{\mathrm{s}}^T \; \boldsymbol{u}_{\mathrm{s}}^T \; \boldsymbol{u}_{\mathrm{a}}^T \right]$.

Analogous to the p method used in flutter analysis, we expand the states into simple exponential functions in time as follows,

$$\boldsymbol{u} = \boldsymbol{v} \exp\left(\lambda t_{\mathrm{s}}\right). \tag{16}$$

This leads to the following generalized eigenvalue problem to determine the stability of the system:

$$(\mathbb{A} - \lambda\mathbb{M})\,\boldsymbol{v} = \boldsymbol{0}. \tag{17}$$

We refer to equation (17) as the high-dimensional eigenproblem, and we denote any eigenvalue λ and (right) eigenvector $\boldsymbol{v}$ which satisfies (17) as a *truth* eigenvalue and *truth* eigenvector.

III. Model Order Reduction

Due to the use of the linearized Euler equations, the aeroelastic eigenproblem (17) is of very high dimension and thus remains impractical for fast dynamic aeroelastic analysis. If instead of using the high-dimensional aerodynamic model (12) we approximate the behavior of the aerodynamics with a ROM, the aeroelastic eigenproblem becomes tractable. This section presents an overview of the application of projection-based model order reduction to the aerodynamic model used in this work. The approach used to form the reduced basis is discussed in Section V below.

A. Projection-Based Model Order Reduction

To begin, assume we possess a set of n_a reduced basis vectors stored column-wise in the reduced basis $\Phi = \{\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, ..., \boldsymbol{\phi}_{n_\mathrm{a}}\}$. In projecting the linearized aerodynamic model (12) onto the reduced basis, one approximates the aerodynamic solution in the reduced space [37]:

$$\boldsymbol{u}_\mathrm{a} \approx \Phi\tilde{\boldsymbol{u}}_\mathrm{a}, \tag{18}$$

where $\tilde{\boldsymbol{u}}_\mathrm{a} \in \mathbb{R}^{n_\mathrm{a}}$ is a vector of modal coefficients. Inserting this approximation into the high-dimensional aerodynamic model (12), we obtain the residual,

$$\boldsymbol{r}_\mathrm{ROM} = E_\mathrm{a}\Phi\frac{d\tilde{\boldsymbol{u}}_\mathrm{a}}{dt_\mathrm{s}} - A_\mathrm{a}\Phi\tilde{\boldsymbol{u}}_\mathrm{a} - B_{\mathrm{a},1}\boldsymbol{u}_\mathrm{s} - B_{\mathrm{a},2}\dot{\boldsymbol{u}}_\mathrm{s}. \tag{19}$$

Enforcing the Petrov-Galerkin condition, which states that the residual is orthogonal to the test basis Ψ, leads to:

$$\Psi^T W\boldsymbol{r}_\mathrm{ROM} = \boldsymbol{0}, \tag{20}$$

where W is a matrix which approximates a continuous inner-product in Euclidean space. In this work, W is also used to stabilize the aerodynamic ROM; this is further discussed in Section III.B below. The resulting aerodynamic ROM is,

$$\tilde{E}_\mathrm{a}\frac{d\tilde{\boldsymbol{u}}_\mathrm{a}}{dt_\mathrm{s}} = \tilde{A}_\mathrm{a}\tilde{\boldsymbol{u}}_\mathrm{a} + \tilde{B}_{\mathrm{a},1}\boldsymbol{u}_\mathrm{s} + \tilde{B}_{\mathrm{a},2}\dot{\boldsymbol{u}}_\mathrm{s}, \tag{21}$$

where the reduced matrices are

$$\tilde{E}_{\mathrm{a}} = \Psi^T W E_{\mathrm{a}} \Phi, \qquad \tilde{A}_{\mathrm{a}} = \Psi^T W A_{\mathrm{a}} \Phi, \qquad \tilde{B}_{\mathrm{a},1} = \Psi^T W B_{\mathrm{a},1}, \qquad \tilde{B}_{\mathrm{a},2} = \Psi^T W B_{\mathrm{a},2}.$$

To differentiate this aerodynamic ROM from that used for the error estimator, we denote equation (21) as the *primal* aerodynamic ROM.

Coupling the primal aerodynamic ROM to equations (5) and (13) forms the (primal) aeroelastic ROM,

$$\begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & \tilde{E}_{\mathrm{a}} \end{bmatrix} \frac{d}{dt_{\mathrm{s}}} \begin{bmatrix} \dot{\boldsymbol{u}}_{\mathrm{s}} \\ \boldsymbol{u}_{\mathrm{s}} \\ \tilde{\boldsymbol{u}}_{\mathrm{a}} \end{bmatrix} = \begin{bmatrix} 0 & A_{\mathrm{s}} & \tilde{C}_{\mathrm{s}} \\ I & 0 & 0 \\ \tilde{B}_{\mathrm{a},2} & \tilde{B}_{\mathrm{a},1} & \tilde{A}_{\mathrm{a}} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{u}}_{\mathrm{s}} \\ \boldsymbol{u}_{\mathrm{s}} \\ \tilde{\boldsymbol{u}}_{\mathrm{a}} \end{bmatrix}, \tag{22}$$

where we denote $\tilde{C}_{\mathrm{s}} \equiv C_{\mathrm{s}}\Phi$. As before, we write this in a more convenient form,

$$\tilde{\mathbb{M}} \frac{d\tilde{\boldsymbol{u}}}{dt_{\mathrm{s}}} = \tilde{\mathbb{A}} \tilde{\boldsymbol{u}}, \tag{23}$$

where $\tilde{\boldsymbol{u}}^T = \left[\dot{\boldsymbol{u}}_{\mathrm{s}}^T \; \boldsymbol{u}_{\mathrm{s}}^T \; \tilde{\boldsymbol{u}}_{\mathrm{a}}^T\right]$. One can immediately see that the matrices $\tilde{A}_{\mathrm{a}}$ and $\tilde{E}_{\mathrm{a}}$ are now of order n_{a}, which is typically orders of magnitude smaller than the number of degrees of freedom in the high-dimensional aerodynamic model. Assuming $\tilde{\boldsymbol{u}} = \tilde{\boldsymbol{v}} \exp\left(\tilde{\lambda} t_{\mathrm{s}}\right)$, the stability of the aeroelastic system can now be determined by the tractable eigenproblem,

$$\left(\tilde{\mathbb{A}} - \tilde{\lambda}\tilde{\mathbb{M}}\right) \tilde{\boldsymbol{v}} = \boldsymbol{0}. \tag{24}$$

In general, $\tilde{\lambda} \neq \lambda$ due to the approximation of the right eigenvector in the reduced space,

$$\boldsymbol{v} \approx \check{\Phi} \tilde{\boldsymbol{v}}, \tag{25}$$

where we define the augmented reduced basis as:

$$\check{\Phi} = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & \Phi \end{bmatrix}.$$

The size of the identity matrices I here is equal to the number of structural degrees-of-freedom.

B. Stabilizing Inner Product and Test Basis Selection

It has previously been shown that the stability of projection-based ROMs constructed from linearized systems of equations can be ensured with the use of a symmetrizing weighted inner product [58, 59]. In this work, we use a similar approach as Lowe and Zingg [48] to stabilize the aerodynamic ROM. It has been shown that the Hessian of the mathematical entropy $-\rho s$ (where ρ is the density and s is the thermodynamic entropy) symmetrizes the flux Jacobian of the Euler equations, and thus can be used as an inner product to stabilize the aerodynamic ROM. Therefore, the matrix W used to approximate the continuous inner product for the Petrov-Galerkin projection in equation (20) is constructed as a block diagonal matrix containing the Hessian of the mathematical entropy at each node as well as trapezoidal quadrature weights to approximate the integration for the inner product.

Moreover, as will be discussed later, only eigenvalues originating from the wind-off structural modes are important for flutter analysis. In this paper we term these the “aeroelastic modes”. The remaining eigenvalues originate from the uncoupled aerodynamic model, and typically remain stable for the purposes of flutter analysis. Following Dowell *et al*. [60], we call these the “aerodynamic modes”; these are analogous to the aerodynamic lag roots obtained from classical flutter analysis methods [61]. When forming the aerodynamic ROM, the choice of test basis Ψ used for the Petrov-Galerkin projection (20) has a large impact on the distribution of the aerodynamic modes. To simplify the process of tracking and analyzing the aeroelastic modes, it is ideal to keep the aerodynamic modes separate from the aeroelastic modes. We have found that good separation between these two groups of modes is obtained by using the test basis

$$\Psi = J_0^{-1}\Phi,$$

where J_0^{-1} is the diagonal matrix of inverse metric Jacobians of the transformation to curvilinear coordinates for the steady-state grid.

IV. Error Estimation

The stability of the aeroelastic system is determined by the eigenvalues of the high-dimensional eigenproblem (17). In order to ensure that the aeroelastic ROM provides sufficiently accurate results relative to the high-dimensional model, we wish to estimate the error between the eigenvalues obtained from the reduced eigenproblem (24) and the corresponding truth eigenvalues. If the estimated error exceeds a user-prescribed tolerance, then the aerodynamic ROM can be trained to produce a more accurate eigenvalue. The following section presents the error estimator used in this work. The estimator is based on the work of Heuveline and Rannacher [51], who presented a DWR-based error estimator for eigenvalue problems. An overview of the approach is presented in this section; the reader is referred to Bangerth and Rannacher [62] for details of the derivation.

A. Dual-Weighted Residual-Based Error Estimation

To begin, we present the primal and dual problems of interest. The primal problem is the high-dimensional right eigenvalue problem from equation (17), rewritten here for convenience, with an extra normalization condition on the eigenvector,

$$\mathbb{A}\boldsymbol{v} = \lambda \mathbb{M}\boldsymbol{v}, \qquad \boldsymbol{v}^H \mathbb{M}\boldsymbol{v} = 1, \tag{26}$$

where a superscript ^{H} denotes the conjugate transpose. For flutter analysis, we are specifically interested in the eigenvalue; thus we define the output functional as:

$$J(\mu, \boldsymbol{w}) = \mu \boldsymbol{w}^H \mathbb{M}\boldsymbol{w}. \tag{27}$$

Due to the normalization condition, this output functional provides the eigenvalue: $J(\lambda, \boldsymbol{v}) = \lambda \boldsymbol{v}^H \mathbb{M}\boldsymbol{v} = \lambda$. For this output functional, the dual solution is obtained from the high-dimensional left eigenproblem with an added normalization condition:

$$(\boldsymbol{v}^{\mathrm{du}})^H \mathbb{A} = \lambda (\boldsymbol{v}^{\mathrm{du}})^H \mathbb{M}, \qquad (\boldsymbol{v}^{\mathrm{du}})^H \mathbb{M}\boldsymbol{v} = 1, \tag{28}$$

where $\boldsymbol{v}^{\mathrm{du}}$ is the truth left eigenvector. For further details see [62].

Now, assume we have solved the reduced (primal) eigenproblem (24) for an approximate right eigenpair $\left(\tilde{\lambda}, \tilde{\boldsymbol{v}}\right)$. Introducing this approximate eigenpair into the high-dimensional right eigenproblem (26), we obtain the following residual:

$$\boldsymbol{r} = \left(\mathbb{A} - \tilde{\lambda}\mathbb{M}\right) \check{\boldsymbol{\Phi}}\tilde{\boldsymbol{v}}. \tag{29}$$

Given the truth left eigenvector $\boldsymbol{v}^{\mathrm{du}}$, the error in the solution is given by the equation,

$$\lambda - \tilde{\lambda} = (\boldsymbol{v}^{\mathrm{du}})^H \boldsymbol{r} + \mathcal{R}^{(2)}, \tag{30}$$

where $\mathcal{R}^{(2)}$ is a second-order remainder term [62].

Obtaining the error through equation (30) (with the remainder term neglected) remains impractical because we require the truth left eigenvector $\boldsymbol{v}^{\mathrm{du}}$ from the high-dimensional left eigenproblem (28). To reduce computational cost, we obtain an approximate left eigenvector with a second ROM, denoted as the dual ROM.

B. Dual ROM and Error Estimator

To obtain the dual aeroelastic system, we transpose the state and mass matrices from the primal high-dimensional aeroelastic model (14), giving

$$\begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & E_\text{a} \end{bmatrix} \frac{d}{dt_\text{s}} \begin{bmatrix} \dot{\boldsymbol{u}}_\text{s}^\text{du} \\ \boldsymbol{u}_\text{s}^\text{du} \\ \boldsymbol{u}_\text{a}^\text{du} \end{bmatrix} = \begin{bmatrix} 0 & I & B_{\text{a},2}^T \\ A_\text{s}^T & 0 & B_{\text{a},1}^T \\ C_\text{s}^T & 0 & A_\text{a}^T \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{u}}_\text{s}^\text{du} \\ \boldsymbol{u}_\text{s}^\text{du} \\ \boldsymbol{u}_\text{a}^\text{du} \end{bmatrix}. \tag{31}$$

Note that all matrices are real and that the matrix E_a is symmetric and thus does not require a transpose. The dual aerodynamic model is then,

$$E_\text{a} \frac{d\boldsymbol{u}_\text{a}^\text{du}}{dt_\text{s}} = A_\text{a}^T \boldsymbol{u}_\text{a}^\text{du} + C_\text{s}^T \dot{\boldsymbol{u}}_\text{s}^\text{du}. \tag{32}$$

As for the primal aerodynamic ROM, we approximate the dual aerodynamic solution in a reduced space spanned by the reduced basis Φ^du:

$$\boldsymbol{u}_\text{a}^\text{du} \approx \Phi^\text{du} \tilde{\boldsymbol{u}}_\text{a}^\text{du}, \tag{33}$$

where $\tilde{\boldsymbol{u}}_\text{a}^\text{du}$ is a vector of modal coefficients. From this approximation, we obtain the residual,

$$\boldsymbol{r}_\text{ROM}^\text{du} = E_\text{a} \Phi^\text{du} \frac{d\tilde{\boldsymbol{u}}_\text{a}^\text{du}}{dt_\text{s}} - A_\text{a}^T \Phi^\text{du} \tilde{\boldsymbol{u}}_\text{a}^\text{du} - C_\text{s}^T \dot{\boldsymbol{u}}_\text{s}^\text{du}. \tag{34}$$

Enforcing the Petrov-Galerkin condition $\left(\Psi^\text{du}\right)^T W \boldsymbol{r}_\text{ROM}^\text{du} = \boldsymbol{0}$ leads to the dual aerodynamic ROM,

$$\tilde{E}_\text{a}^\text{du} \frac{d\tilde{\boldsymbol{u}}_\text{a}^\text{du}}{dt_\text{s}} = \tilde{A}_\text{a}^\text{du} \tilde{\boldsymbol{u}}_\text{a}^\text{du} + \tilde{C}_\text{s}^\text{du} \dot{\boldsymbol{u}}_\text{s}^\text{du}, \tag{35}$$

where the reduced matrices are

$$\tilde{E}_\text{a}^\text{du} = \left(\Psi^\text{du}\right)^T W E_\text{a} \Phi^\text{du}, \qquad \tilde{A}_\text{a}^\text{du} = \left(\Psi^\text{du}\right)^T W A_\text{a}^T \Phi^\text{du}, \qquad \tilde{C}_\text{s}^\text{du} = \left(\Psi^\text{du}\right)^T W C_\text{s}^T.$$

In this work, the test basis for the dual aerodynamic ROM is constructed in the same manner as the primal ROM, that is $\Psi^\text{du} = J_0^{-1} \Phi^\text{du}$. With this dual aerodynamic ROM, we can form the dual aeroelastic ROM,

$$\begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & \tilde{E}_\text{a}^\text{du} \end{bmatrix} \frac{d}{dt_\text{s}} \begin{bmatrix} \dot{\boldsymbol{u}}_\text{s}^\text{du} \\ \boldsymbol{u}_\text{s}^\text{du} \\ \tilde{\boldsymbol{u}}_\text{a}^\text{du} \end{bmatrix} = \begin{bmatrix} 0 & I & \tilde{B}_{\text{a},2}^\text{du} \\ A_\text{s}^T & 0 & \tilde{B}_{\text{a},1}^\text{du} \\ \tilde{C}_\text{s}^\text{du} & 0 & \tilde{A}_\text{a}^\text{du} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{u}}_\text{s}^\text{du} \\ \boldsymbol{u}_\text{s}^\text{du} \\ \tilde{\boldsymbol{u}}_\text{a}^\text{du} \end{bmatrix}, \tag{36}$$

where we introduce the matrices $\tilde{B}^{\mathrm{du}}_{\mathrm{a},1} \equiv B^{T}_{\mathrm{a},1}\Phi^{\mathrm{du}}$ and $\tilde{B}^{\mathrm{du}}_{\mathrm{a},2} \equiv B^{T}_{\mathrm{a},2}\Phi^{\mathrm{du}}$. In a more convenient form, we write this as

$$\tilde{\mathbb{M}}^{\mathrm{du}}\frac{d\tilde{\boldsymbol{u}}^{\mathrm{du}}}{dt_{\mathrm{s}}} = \tilde{\mathbb{A}}^{\mathrm{du}}\tilde{\boldsymbol{u}}^{\mathrm{du}}. \tag{37}$$

This directly leads to the reduced dual generalized eigenproblem,

$$\left(\tilde{\mathbb{A}}^{\mathrm{du}} - \tilde{\lambda}^{\mathrm{du}}\tilde{\mathbb{M}}^{\mathrm{du}}\right)\tilde{\boldsymbol{v}}^{\mathrm{du}} = \mathbf{0}, \tag{38}$$

where $\tilde{\boldsymbol{v}}^{\mathrm{du}}$ is a reduced approximation of the truth left eigenvector $\boldsymbol{v}^{\mathrm{du}}$ such that

$$\boldsymbol{v}^{\mathrm{du}} \approx \check{\boldsymbol{\Phi}}^{\mathrm{du}}\tilde{\boldsymbol{v}}^{\mathrm{du}}, \tag{39}$$

with the augmented dual reduced basis:

$$\check{\boldsymbol{\Phi}}^{\mathrm{du}} = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & \Phi^{\mathrm{du}} \end{bmatrix}. \tag{40}$$

Introducing the left eigenvector approximation (39) into equation (30), and neglecting the remainder term, we obtain the error estimator η used in this work:

$$\lambda - \tilde{\lambda} \approx \eta \equiv \left(\check{\boldsymbol{\Phi}}^{\mathrm{du}}\tilde{\boldsymbol{v}}^{\mathrm{du}}\right)^{H}\boldsymbol{r}. \tag{41}$$

V. ROM-Based Flutter Analysis with Error Estimation

In this section, we present our approach to constructing an aeroelastic ROM capable of approximating aeroelastic eigenvalues and the flutter point to a user-prescribed accuracy. We first define the parameter on which our aeroelastic ROMs depend. Aerodynamic parameters including the Mach number and angle of attack are held fixed during our analyses. Similarly, structural parameters such as the structural mass and Young's modulus are also held fixed. The parameter for the aeroelastic ROMs is the variable which affects the coupling between both the aerodynamic and structural models. For our purposes, the aeroelastic parameter is the speed index $V = \frac{V_\infty}{b\omega_\alpha\sqrt{\mu}}$, where V_∞ is the freestream speed, b is the root semichord, ω_α is the uncoupled first torsional frequency, and μ is the mass ratio. The speed index is a nondimensional analogue of the freestream dynamic pressure.

To introduce our method, we first show the flow chart of the algorithm in Figure 1. The novel method can be divided into three stages: ROM initialization, ROM training, and the flutter point calculation. In the ROM initialization stage, an initial ROM is constructed using snapshots from the high-dimensional aerodynamic system. In the ROM training stage, the algorithm tracks the aeroelastic eigenvalues relevant for flutter analysis for increasing speed index values.

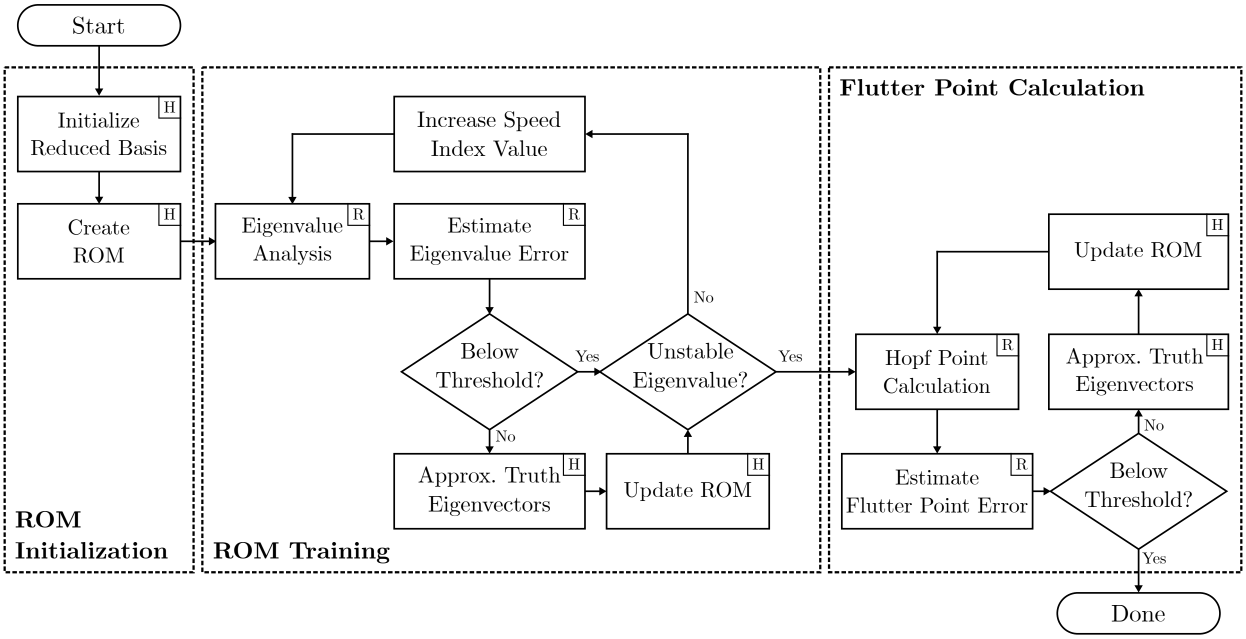


Fig. 1 ROM-based flutter prediction methodology with error estimation; reduced computations denoted with R, high-dimensional computations denoted with H.

This is accomplished by iteratively solving for the eigenvalues of the ROM at increasing values of speed index, and tracking the aeroelastic eigenvalues between increments. At each value of speed index, the reduced basis is updated if the error estimate of an eigenvalue is above the user-prescribed threshold. Once an unstable eigenvalue is found, the algorithm moves to the flutter point calculation stage. In this stage, a Hopf point calculation procedure is used to determine the flutter point of the aeroelastic ROM. Subsequently, the error in this approximate flutter point is estimated. The algorithm terminates when this error estimate satisfies the user-prescribed threshold; otherwise the ROM is updated and the Hopf point calculation is repeated. When the algorithm terminates, it provides an approximate flutter speed index and an estimate of the error in this quantity. Additional results include the flutter frequency and approximate left and right eigenvectors of the aeroelastic system at the flutter point. In this section, we discuss the details of each of these three stages.

A. ROM Initialization

Though the goal of constructing the aeroelastic ROM is to eventually obtain a desired level of accuracy for aeroelastic eigenvalue approximations, this accuracy will be achieved through updates to the reduced basis in the ROM training stage of the algorithm. Here we discuss the computation of the initial primal and dual reduced bases, Φ and Φ^{du}, as a first step towards constructing the primal and dual ROMs. In this work, the initial primal and dual reduced bases are constructed from solutions of the primal and dual high-dimensional aerodynamic models in the frequency domain. To obtain these equations, the aerodynamic and structural vectors are assumed to be harmonic,

$$\boldsymbol{u}_{\mathrm{a}}(t) = \bar{\boldsymbol{u}}_{\mathrm{a}} e^{i\omega t}, \qquad \boldsymbol{u}_{\mathrm{s}}(t) = \bar{\boldsymbol{u}}_{\mathrm{s}} e^{i\omega t}, \tag{42}$$

where $\bar{\boldsymbol{u}}_{\mathrm{a}}$ and $\bar{\boldsymbol{u}}_{\mathrm{s}}$ are the aerodynamic and structural Fourier coefficients, respectively. Inserting these expressions into the high-dimensional linearized aerodynamic model (12), we obtain the governing primal aerodynamic equations in the frequency domain,

$$\left(i\omega E_{\mathrm{a}} - A_{\mathrm{a}}\right) \bar{\boldsymbol{u}}_{\mathrm{a}} = \left(B_{\mathrm{a},1} + i\omega B_{\mathrm{a},2}\right) \bar{\boldsymbol{u}}_{\mathrm{s}}. \tag{43}$$

Using the same approach, we can convert the high-dimensional dual aerodynamic model (32) to the frequency domain,

$$\left(i\omega E_{\mathrm{a}} - A_{\mathrm{a}}^{T}\right) \bar{\boldsymbol{u}}_{\mathrm{a}}^{\mathrm{du}} = i\omega C_{\mathrm{s}}^{T} \bar{\boldsymbol{u}}_{\mathrm{s}}^{\mathrm{du}}. \tag{44}$$

To obtain initial snapshots for the reduced bases, equations (43) and (44) are solved for several speed index values. At each speed index value, these equations are solved with the frequency ω set to the natural frequencies of the structure. At each natural frequency, the corresponding generalized velocity and displacement are set to unity and the inverse frequency, respectively, in the input vectors $\bar{\boldsymbol{u}}_{\mathrm{s}}$ and $\bar{\boldsymbol{u}}_{\mathrm{s}}^{\mathrm{du}}$. Note that changing the speed index results in changes to the system matrices in equations (43) and (44) due to different time and force scaling factors σ_{t} and σ_{f}.

We have found that solving equations (43) and (44) for five speed index values produces initial ROMs with good error estimation capability. Moreover, the equations are solved only to a relative tolerance of 10^{-1} to obtain the initial snapshots. Once all the snapshots of $\bar{\boldsymbol{u}}_{\mathrm{a}}$ and $\bar{\boldsymbol{u}}_{\mathrm{a}}^{\mathrm{du}}$ have been obtained, they are orthogonalized using the modified Gram-Schmidt procedure and used to create the initial reduced bases Φ and Φ^{du}. Note that the real and imaginary parts of the snapshots are included as separate vectors in Φ and Φ^{du} to produce only real valued reduced bases.

B. ROM Training

The error estimator (41) allows us to approximate the error in any eigenvalue obtained from the reduced aeroelastic eigenproblem (24). However, not all eigenvalues obtained from the aeroelastic ROM are important for flutter analysis. Certain eigenmodes in the aeroelastic system are associated to the wind-off structural modes, and it is one of these eigenmodes that will become unstable. At low speed index values, these are easily determined because each eigenvalue has an imaginary part near the natural frequency of its associated structural mode. As the speed index is increased, the eigenvalues change in value. Figure 2 demonstrates an example root locus for increasing speed index for an aeroelastic system with two structural modes.

Based on this knowledge, we wish to create a reduced basis that spans the subspace of the eigenvectors of these important aeroelastic modes for a range of speed index values. In this pursuit, we require two algorithmic elements:

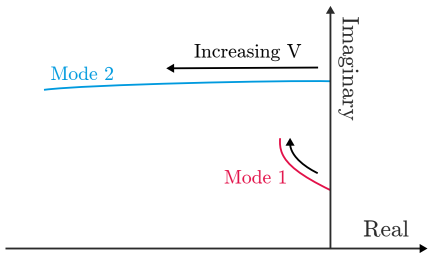


Fig. 2 Example of an aeroelastic root locus plot for increasing speed index, showing only important aeroelastic modes.

a robust eigenvalue tracking algorithm and a means of approximating truth eigenvectors. The eigenvalue tracking algorithm is required to know which eigenvalues are relevant for flutter analysis and necessitate error estimation. A means of approximating truth eigenvectors is required in order to train the reduced basis to obtain better approximations of aeroelastic eigenvalues. In this section, we present our approach to each of the aforementioned elements.

1. Eigenvalue Tracking

Once the aeroelastic ROMs have been initialized, the reduced eigenproblem (24) can be used to approximate aeroelastic eigenvalues. At speed index values near zero, the eigenvalues relevant for flutter analysis are those with an imaginary part near a natural frequency of the structure. As the speed index is increased, the values of these eigenvalues change, which leads to the need for tracking. Assume we have an approximate eigenvalue $\tilde{\lambda}^{(n-1)}$ known to be important for flutter analysis at the speed index increment $n-1$. At increment n, after solving the eigenproblem (24), we have a number of approximate eigenvalues $\tilde{\lambda}_i^{(n)}$ for $i = 1, 2, ...,$ and we wish to know which eigenvalue $\tilde{\lambda}_i^{(n)}$ is associated to the known eigenvalue $\tilde{\lambda}^{(n-1)}$. Several algorithms have been proposed for this purpose [63, 64]. In this work, we use a mixture of three metrics to track the eigenvalues.

The first tracking metric is based on the reciprocal of the error estimator (41). At increment n, we have the approximate eigenvalues $\tilde{\lambda}_i^{(n)}$, and approximate right and left eigenvectors, $\tilde{\boldsymbol{v}}_i^{(n)}$ and $\tilde{\boldsymbol{v}}_i^{\text{du}(n)}$, respectively. Assuming the eigenvectors vary smoothly with changing speed index value, we can use the left eigenvector at the previous increment associated to the known eigenvalue $\tilde{\lambda}^{(n-1)}$ to estimate the error in the new i^{th} approximate eigenvalue. This leads to the first metric:

$$\sigma_i^{\text{err}} \equiv \left| \left(\boldsymbol{w}_{\mathbb{A}}^H - \tilde{\lambda}_i^{(n)} \boldsymbol{w}_{\mathbb{M}}^H \right) \tilde{\boldsymbol{v}}_i^{(n)} \right|^{-1}, \tag{45}$$

where

$$\boldsymbol{w}_{\mathbb{A}}^H = \left(\check{\boldsymbol{\Phi}}^{\text{du}} \tilde{\boldsymbol{v}}^{\text{du}(n-1)} \right)^H \mathbb{A}^{(n)} \check{\boldsymbol{\Phi}}, \qquad \boldsymbol{w}_{\mathbb{M}}^H = \left(\check{\boldsymbol{\Phi}}^{\text{du}} \tilde{\boldsymbol{v}}^{\text{du}(n-1)} \right)^H \mathbb{M}^{(n)} \check{\boldsymbol{\Phi}},$$

and $\tilde{\boldsymbol{v}}^{\text{du}(n-1)}$ is the approximate left eigenvector for the known eigenvalue $\tilde{\lambda}^{(n-1)}$. We specifically use the reciprocal of the error estimator for tracking to ensure that a small error estimate leads to a large tracking merit. One can see that if we replace the vector $\tilde{\boldsymbol{v}}^{\text{du}(n-1)}$ with $\tilde{\boldsymbol{v}}^{\text{du}(n)}$ in equation (45) above, we regain the reciprocal of the standard error estimator from (41). Thus the only difference between error-based tracking metric (45) and the reciprocal of (41) is the use of a left eigenvector at a previous speed index value. The vectors $\boldsymbol{w}_{\mathbb{A}}^{H}$ and $\boldsymbol{w}_{\mathbb{M}}^{H}$ need only be computed once at each speed index increment, and therefore the term σ_j^{err} is efficient to compute for all approximate eigenvalues.

The second metric is the inverse of the magnitude of the difference between the eigenvalues,

$$\sigma_i^{\text{diff}} \equiv \left|\tilde{\lambda}^{(n-1)} - \tilde{\lambda}_i^{(n)}\right|^{-1}. \tag{46}$$

Once again, we use the reciprocal to ensure that small differences lead to large merits. The third metric is the eigenvector correlation-based method of van Zyl [63]. It can be compactly expressed as

$$\sigma_i^{\text{corr}} \equiv \frac{\left|\left(\tilde{\boldsymbol{v}}^{(n-1)}\right)^H \tilde{\boldsymbol{v}}_i^{(n)}\right|}{\left\|\tilde{\boldsymbol{v}}^{(n-1)}\right\|_2 \left\|\tilde{\boldsymbol{v}}_i^{(n)}\right\|_2}, \tag{47}$$

where $\tilde{\boldsymbol{v}}^{(n-1)}$ and $\tilde{\boldsymbol{v}}_i^{(n)}$ are the reduced right eigenvectors associated to $\tilde{\lambda}^{(n-1)}$ and $\tilde{\lambda}_i^{(n)}$, respectively, and $\|\cdot\|_2$ indicates the 2-norm of the vector.

Once computed, all the metrics σ_i^{err}, σ_i^{diff}, σ_i^{corr} are normalized so that their values over all indices i sum to one. Subsequently, the three metrics are aggregated into a logarithmic opinion pool,

$$p_i = \left(\sigma_i^{\text{err}}\right)^{w_1} \left(\sigma_i^{\text{diff}}\right)^{w_2} \left(\sigma_i^{\text{corr}}\right)^{w_3}, \tag{48}$$

where the weights w_1, w_2, and w_3 sum to one. The eigenvalue $\tilde{\lambda}_i^{(n)}$ with the largest value of p_i is assumed to be associated to $\tilde{\lambda}^{(n-1)}$. However, the largest value of p_i must be at least 50% higher than the second-largest value; otherwise the tracking is flagged as a failure and the speed index step is halved. For the results obtained in this paper, the following weight values were found to provide reliable eigenvalue tracking:

$$w_1 = 0.5, \qquad w_2 = 0.4, \qquad w_3 = 0.1.$$

2. Reduced Basis Updates using Approximate Eigenvectors

Using the eigenvalue tracking method outlined above, we can determine which approximate eigenvalues from the aeroelastic ROMs are relevant for flutter analysis. Furthermore, using the error estimator (41), we can estimate the error in these approximate eigenvalues. If the error of an eigenvalue is above a user-prescribed tolerance, then an approximate truth eigenvalue and approximate right and left truth eigenvectors are computed for the high-dimensional eigenvalue problem (17). The new truth eigenvectors are then added to the primal and dual reduced bases. By repeating this process for various speed index values, we can create aeroelastic ROMs capable of approximating the aeroelastic eigenvalues over a range of speed indices to a user-prescribed tolerance.

In this work, a two-sided refined Jacobi-Davidson method [52–54] is used to solve the high-dimensional eigenproblem (17). This algorithm allows for both the left and right eigenvectors to be obtained simultaneously for the eigenvalue nearest a user-specified target. We choose the approximate eigenvalue $\tilde{\lambda}^{(n)}$ obtained from the primal aeroelastic ROM as the target. To minimize computation costs, we have developed a new stopping criterion for the method. At each iteration k, the two-sided Jacobi-Davidson method provides a new approximate (truth) right eigenvector $\boldsymbol{v}^{(k)}$ and left eigenvector $\boldsymbol{v}^{\mathrm{du}(k)}$. The Rayleigh quotient of the right eigenvector approximates the truth eigenvalue, and is computed as

$$\theta^{(k)} = \frac{\left(\boldsymbol{v}^{(k)}\right)^H \mathbb{A}\boldsymbol{v}^{(k)}}{\left(\boldsymbol{v}^{(k)}\right)^H \mathbb{M}\boldsymbol{v}^{(k)}}, \tag{49}$$

Using the approximate truth left eigenvector $\boldsymbol{v}^{\mathrm{du}(k)}$, the error in this Rayleigh quotient can be estimated as,

$$\lambda - \theta^{(k)} \approx \left(\boldsymbol{v}^{\mathrm{du}(k)}\right)^H \boldsymbol{r}_\theta, \tag{50}$$

where $\boldsymbol{r}_\theta$ is the residual associated to the Rayleigh quotient,

$$\boldsymbol{r}_\theta = \mathbb{A}\boldsymbol{v}^{(k)} - \theta^{(k)}\mathbb{M}\boldsymbol{v}^{(k)}.$$

Thus, the two-sided Jacobi-Davidson method is run until the error estimate (50) is below the user-set tolerance for the Rayleigh quotient error.

Once the Jacobi-Davidson algorithm has sufficiently converged, it provides approximate truth right and left eigenvectors $\boldsymbol{v}$ and $\boldsymbol{v}^{\mathrm{du}}$, respectively. Note that the eigenvectors are composed of structural and aerodynamic parts: $\boldsymbol{v}^T = \left[\dot{\boldsymbol{v}}_\mathrm{s}^T,\ \boldsymbol{v}_\mathrm{s}^T,\ \boldsymbol{v}_\mathrm{a}^T\right]$. To update the primal and dual aerodynamic ROMs, the aerodynamic part of the eigenvalues are added to the existing reduced bases. The right eigenvectors are added to the primal reduced basis Φ, while the left eigenvectors are added to the dual reduced basis Φ^{du}. The real and imaginary parts are included separately in order to keep the reduced bases, and hence the reduced matrices, real valued,

$$\Phi = \{\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \ldots, \mathrm{Real}\{\boldsymbol{v}_\mathrm{a}\}, \mathrm{Imag}\{\boldsymbol{v}_\mathrm{a}\}\},$$
$$\Phi^{\mathrm{du}} = \left\{\boldsymbol{\phi}_1^{\mathrm{du}}, \boldsymbol{\phi}_2^{\mathrm{du}}, \ldots, \mathrm{Real}\{\boldsymbol{v}_\mathrm{a}^{\mathrm{du}}\}, \mathrm{Imag}\{\boldsymbol{v}_\mathrm{a}^{\mathrm{du}}\}\right\}.$$

To ensure proper conditioning of the reduced matrices, the new vectors are orthogonalized with respect to all other basis vectors using the modified Gram-Schmidt procedure.

C. Flutter Point Calculation

With the ROM training procedure presented Section V.B, we can create an aeroelastic ROM which can approximate important eigenvalues from the aeroelastic HDM to a user-prescribed accuracy. The training procedure is stopped once one of these aeroelastic eigenvalues crosses the imaginary axis, indicating the onset of flutter in the system. Recall that this process is shown in the flow chart in Figure 1 as the "ROM Training" stage. Once an unstable mode is found in the system, the calculation of the flutter point begins. The exact speed index at which flutter occurs (known as the flutter speed index) for the aeroelastic ROM is obtained using the Hopf point calculation method of Griewank and Reddien [65]. However, because the aeroelastic ROM is made to approximate the linearized aeroelastic equations, it cannot be used to approximate the solution of the nonlinear equations which govern the static aeroelastic deflection. For the test cases presented in this paper, the geometries are symmetric and the angles of attack are kept at zero degrees; therefore no static aeroelastic deflections are present in the system. This eliminates the need to solve the nonlinear static aeroelastic equations for changing aeroelastic parameter values. If a geometry is nonsymmetric or the angle of attack is non-zero, then an iterative approach can be used where the Hopf point is computed for a fixed static solution, followed by an update to the static solution to reflect the new Hopf point parameter value.

As the Hopf point is computed for the aeroelastic ROM, the computations are fast. However, the speed index at which the aeroelastic ROM flutters may be different from that of the high-dimensional aeroelastic system. Therefore, we wish to approximate the error in the flutter speed index of the ROM. We denote the flutter speed index of the aeroelastic HDM as V_f, and the flutter speed index of the aeroelastic ROM as $\tilde{V}_\mathrm{f}$. Assuming V_f can be expressed as a function of the real part of the aeroelastic eigenvalue, we expand it into a Taylor series about $\tilde{V}_\mathrm{f}$,

$$V_\mathrm{f} = \tilde{V}_\mathrm{f} + \frac{dV}{d\lambda_\mathrm{r}}\left(0 - \lambda_\mathrm{r}\right) + O\left(\lambda_\mathrm{r}^2\right), \tag{51}$$

where λ_r is the real part of the truth eigenvalue at the speed index value $\tilde{V}_\mathrm{f}$. The real part of the approximate eigenvalue from the aeroelastic ROM $\tilde{\lambda}_\mathrm{r}$ is zero at $\tilde{V}_\mathrm{f}$; we introduce it in equation (51) to eventually retrieve the eigenvalue error estimator. Hence, an error estimate for the flutter point can be obtained as,

$$\left|V_\mathrm{f} - \tilde{V}_\mathrm{f}\right| \approx \left|\frac{d\lambda_\mathrm{r}}{dV}\right|^{-1} \left|\lambda_\mathrm{r} - \tilde{\lambda}_\mathrm{r}\right|. \tag{52}$$

The difference in the real part of the truth and approximate eigenvalues is estimated using,

$$\left|\lambda_\mathrm{r} - \tilde{\lambda}_\mathrm{r}\right| \approx \left|\mathrm{Real}\{\eta\}\right|, \tag{53}$$

where η is the error estimator from (41). The derivative in equation (52) is inverted to simplify its computation. We approximate this derivative using the approximate right and left eigenvectors from the primal and dual ROMs such that

$$\frac{d\lambda_r}{dV} \approx \text{Real}\left\{\left(\check{\Phi}^{\text{du}}\tilde{\boldsymbol{v}}^{\text{du}}\right)^H \left(\frac{d\mathbb{A}}{dV} - \tilde{\lambda}\frac{d\mathbb{M}}{dV}\right)\check{\Phi}\tilde{\boldsymbol{v}}\right\}. \tag{54}$$

Ultimately, we can combine equations (52), (53), and (54) to obtain an error estimator for the flutter point,

$$\left|V_{\text{f}} - \tilde{V}_{\text{f}}\right| \approx \eta_{\text{f}} \equiv \left|\text{Real}\left\{\left(\check{\Phi}^{\text{du}}\tilde{\boldsymbol{v}}^{\text{du}}\right)^H \left(\frac{d\mathbb{A}}{dV} - \tilde{\lambda}\frac{d\mathbb{M}}{dV}\right)\check{\Phi}\tilde{\boldsymbol{v}}\right\}\right|^{-1} \left|\text{Real}\{\eta\}\right|. \tag{55}$$

Using this error estimator, we can determine if the approximated flutter speed index $\tilde{V}_{\text{f}}$ is sufficiently accurate. If it is not, then the aeroelastic primal and dual ROMs are updated using eigenvectors obtained at $\tilde{V}_{\text{f}}$ using the Jacobi-Davidson approach outlined in Section V.B.2. The Hopf point is then recalculated for the updated aeroelastic ROM, and the flutter speed index error is once again computed. This process is repeated until the error estimator η_{f} is below the user-specified tolerance. This process is shown in the flow chart in Figure 1 as the "Flutter Point Calculation" stage.

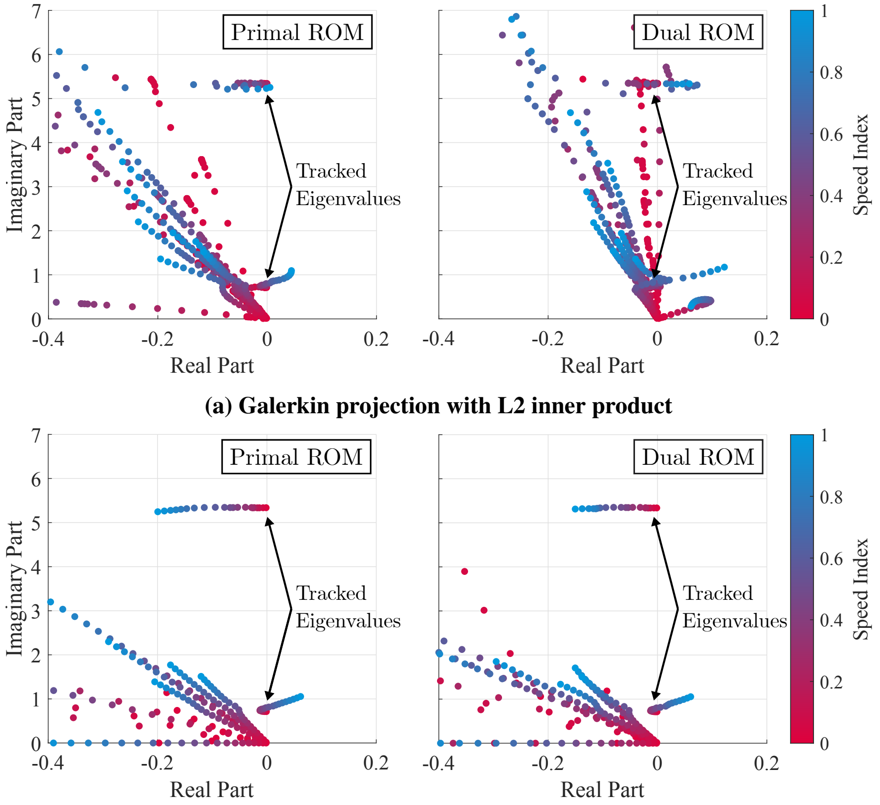
VI. Results

This section presents several results obtained using the approach in this paper. We first present a comparison of the eigenvalue spectrum for aeroelastic ROMs constructed using a standard Galerkin approach and the proposed Petrov-Galerkin approach. Subsequently, flutter analyses are presented for the two degree of freedom NACA 64A010 test case from Isogai [66], the AGARD 445.6 weakened wing model from Yates [67], and the NACA 0012 wing benchmark case from Rivera *et al.* [68].

A. Petrov-Galerkin Projection with Stabilizing Inner Product

In Section III.B we discussed the use of the test basis Ψ and weighted inner product for the construction of the projection-based aerodynamic ROM. Here, we present a comparison of the spectrum of the matrix pencil $(\tilde{\mathbb{A}}, \tilde{\mathbb{M}})$ obtained for the primal and dual aeroelastic ROMs constructed for a two degree of freedom structural model with the use of two different projection methods. The first method is a Galerkin projection ($\Psi = \Phi$) with the matrix W (from equation (20)) approximating a standard L^2 inner product. The second method is a Petrov-Galerkin projection with the test basis $\Psi = J_0^{-1}\Phi$ with the matrix W approximating the weighted L^2 inner product described in Section III.B. The system contains two important aeroelastic modes that require tracking for flutter analysis.

Figure 3 shows the results of the complete spectra $\sigma(\tilde{\mathbb{A}}, \tilde{\mathbb{M}})$ for the resulting primal and dual aeroelastic ROMs. In this figure, the aeroelastic eigenvalues we wish to accurately compute for flutter analysis are labeled "tracked eigenvalues". All other eigenvalues are the aerodynamic modes in the system originating from the uncoupled aerodynamic ROM. From this figure, we see that the Petrov-Galerkin projection provides much better separation between the important aeroelastic modes and the aerodynamic modes. This increased separation leads to less potential for incorrect mode



(a) Galerkin projection with L2 inner product

(b) Petrov-Galerkin projection with weighted inner product

Fig. 3 Eigenvalue spectra of the primal and dual aeroelastic ROMs using different types of projections and inner products.

tracking or failure of the mode tracking algorithm altogether. Moreover, we can see that the Galerkin projection-based ROMs incorrectly predict multiple unstable modes in the system, most notably in the spectrum of the dual ROM. In contrast, the Petrov-Galerkin projection-based ROMs correctly predict an instability in the first aeroelastic mode, seen as the only eigenvalue crossing the imaginary axis in the spectrum.

B. NACA 64A010 Isogai Test Case

To test the accuracy and efficiency of the proposed approach, the two degree of freedom airfoil structure shown in Figure 4 is used. The structure is supported by a torsional and a linear spring and is capable of both pitching and plunging. Aeroelastic results are presented for Case A of Isogai [66], who placed the elastic axis in front of the airfoil leading edge in order to simulate the behavior of a sweptback wing. Dynamic aeroelastic analyses are performed about a steady-state angle of attack of 0°. All simulations pertaining to this model are performed on the NACA 64A010 airfoil with a structured O-mesh. The mesh consists of 38,000 nodes around the airfoil subdivided into 32 blocks. All results are obtained using 32 Intel "Skylake" cores at 2.4 GHz.

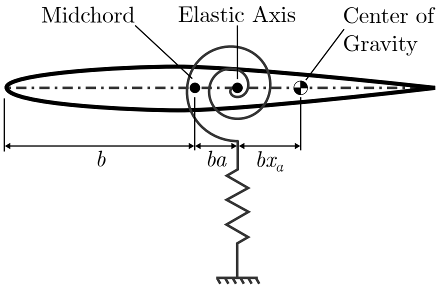


Fig. 4 Two degree of freedom airfoil structure capable of pitching and plunging.

1. Aeroelastic Root Locus

The aeroelastic root locus for this test case was obtained for a Mach number of 0.85 and a series of speed indices between 0.01 and 1.0. A user-prescribed tolerance of

$$|\eta| \leq 10^{-3}$$

was used for all aeroelastic eigenvalues. For analysis purposes, the truth eigenvalues are obtained by reducing the norm of the residual of the high-dimensional right eigenproblem (17) below a tolerance of 10^{-14}. The truth eigenvalues are used to compute the exact errors, which are used to determine the effectiveness of the error estimator.

Figure 5 displays the root locus obtained from the aeroelastic ROM after initializing the reduced bases, but prior to training the ROM using eigenvectors from the Jacobi-Davidson algorithm. Figure 6 shows the converged root locus once the training procedure is complete and all eigenvalue error estimates are below the user-prescribed tolerance. On the right of these figures, we present the root locus of the aeroelastic eigenvalues. On the left, we show the nondimensional damping and frequencies for each mode in the system, these are the real and imaginary parts of the associated eigenvalues, respectively. Moreover, both the error estimate and exact error are plotted for all speed indices. Exact errors for mode 2 at speed indices 0.65 and 0.7 are omitted from the results because the norm of the high-dimensional eigenproblem residual could not be brought to a tolerance of 10^{-14}.

The results in Figure 5 are presented to demonstrate the behavior of the error estimator at the beginning of the ROM training process. From the figure, we can see that the error estimator slightly under predicts the exact error, but generally captures the order of magnitude of the exact error quite well. Moreover, we see that the initial aeroelastic ROM does not predict any instability in the system as no eigenvalues are seen crossing the imaginary axis in Figure 5. One can expect that this may change as the ROM is trained to reduce the error in eigenvalues.

Results in Figure 6 show the behavior of the aeroelastic system once the training is completed and all estimated eigenvalue errors are below the user-specified tolerance of 10^{-3}. We can see that the behavior of the aeroelastic eigenvalues has changed significantly from that presented in Figure 5. We now observe an instability in Mode 1 above

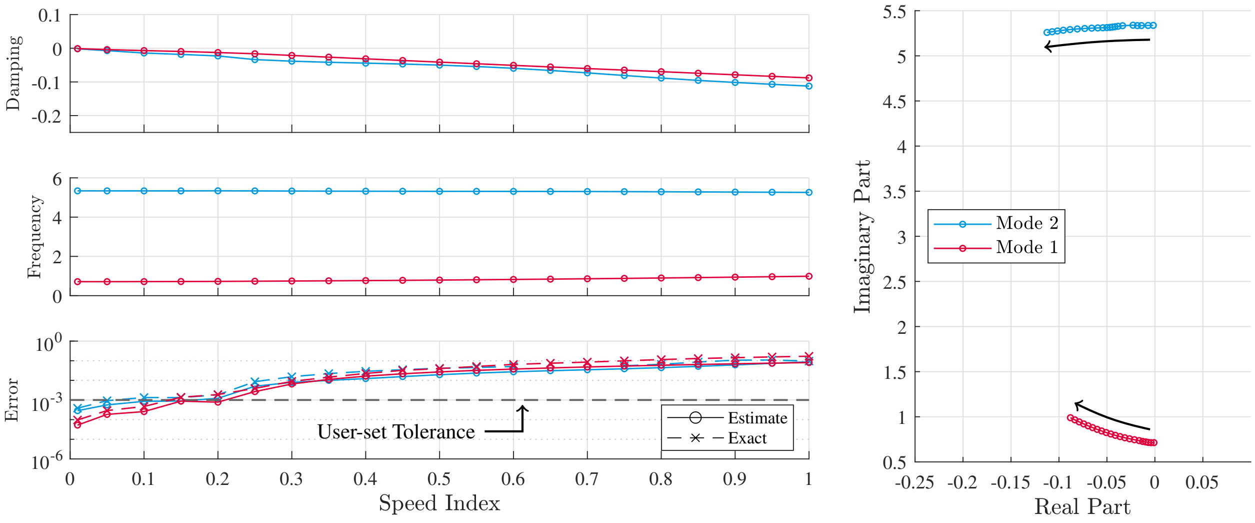


Fig. 5 Aeroelastic mode damping, frequency, and error (left), and root locus plot (right) for the NACA 64A010 Isogai test case at Mach 0.85 obtained immediately after the reduced basis initialization.

a speed index of approximately 0.5. This indicates the onset of flutter in the system, which is analyzed further in subsequent results. Moreover, Mode 2 now demonstrates higher damping, but also shows early signs of reversal at speed index 1.0. For these final results, we see that the error estimator agrees very well with the exact error at speed indices above 0.2. Below this, the error estimator tends to slightly underpredict the error, just as it did for the initial results from Figure 5. This is because the error estimates at these speed indices are below the user-prescribed tolerance, and therefore the ROM is not updated at these low speed indices. In general, when comparing the error estimate to the exact error, we observe very good agreement for the converged results in Figure 6.

2. Flutter Boundary

The inviscid flutter predictions for this test case are well known to have a characteristic S shape in the transonic regime. In this subsection, we show the results obtained for this well-studied flutter boundary, and compare to those found in the literature. To obtain the flutter boundary for the two degree of freedom system, aeroelastic ROMs were constructed for a series of Mach numbers between 0.70 and 0.95. For the ROM training procedure, a tolerance of

$$\left|\frac{\text{Real}\{\eta\}}{\text{Real}\{\tilde{\lambda}\}}\right| \leq 1$$

was used. In other words, a relative error tolerance was placed on the real part of the eigenvalues. A step size of 0.05 in speed index was used for the computation of the root locus. The ROM training persisted until one of the eigenvalues in the system crossed the imaginary axis, at which point the Hopf point calculation was used to obtain the exact flutter

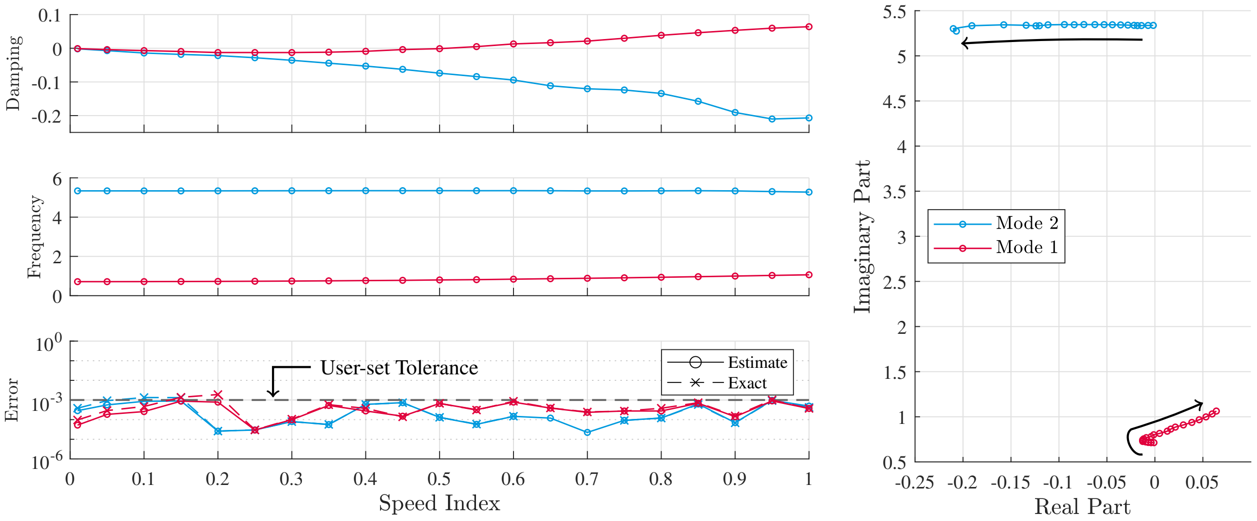


Fig. 6 Aeroelastic mode damping, frequency, and error (left), and root locus plot (right) for the NACA 64A010 Isogai test case at Mach 0.85 obtained after the complete ROM training procedure.

point. A relative flutter speed index error tolerance of

$$\frac{\eta_{\mathrm{f}}}{\tilde{V}_{\mathrm{f}}} \leq 0.005$$

was used as a stopping criterion, where η_{f} is the error estimator for the flutter point given in equation (55), and $\tilde{V}_{\mathrm{f}}$ is the flutter speed index of the aeroelastic ROM. For transonic Mach numbers at which a resurgence and subsequent loss of stability were expected (the characteristic “S” shape), the ROM training procedure was continued after the first Hopf point calculation, allowing for subsequent Hopf points to be determined.

Figure 7 shows the flutter boundary obtained using our approach, along with a pie chart showing the time breakdown for each part of the analysis. One can see that the results are in good agreement with those from the literature. Our approach is capable of determining the three Hopf points at Mach number 0.885 and 0.895. However, for the Mach numbers 0.875 and 0.9, our approach was not able to obtain the third Hopf point (i.e. the second point of instability). At these points, the algorithm had difficulty approximating the eigenvalues of the second aeroelastic mode at large values of speed indices, which lead to breakdowns of the eigenvalue tracking algorithm. However, the algorithm had no difficulty determining the first points of instability in the system for any Mach number.

The time analysis on the right of Figure 7 represents the average time breakdown to obtain the first point of instability at a given Mach number. From this analysis, we can see that all stages in the flutter analysis procedure require similar amounts of time for this test case. Moreover, it can be concluded that obtaining the flutter point at a given Mach number requires on average an amount of time equivalent to 4.5 steady nonlinear flow evaluations.

In Figure 7, one can see some discrepancies between the obtained results and those from the literature, notably at and above Mach 0.9. To reconcile these discrepancies, a grid convergence study was performed. The three grids used were:

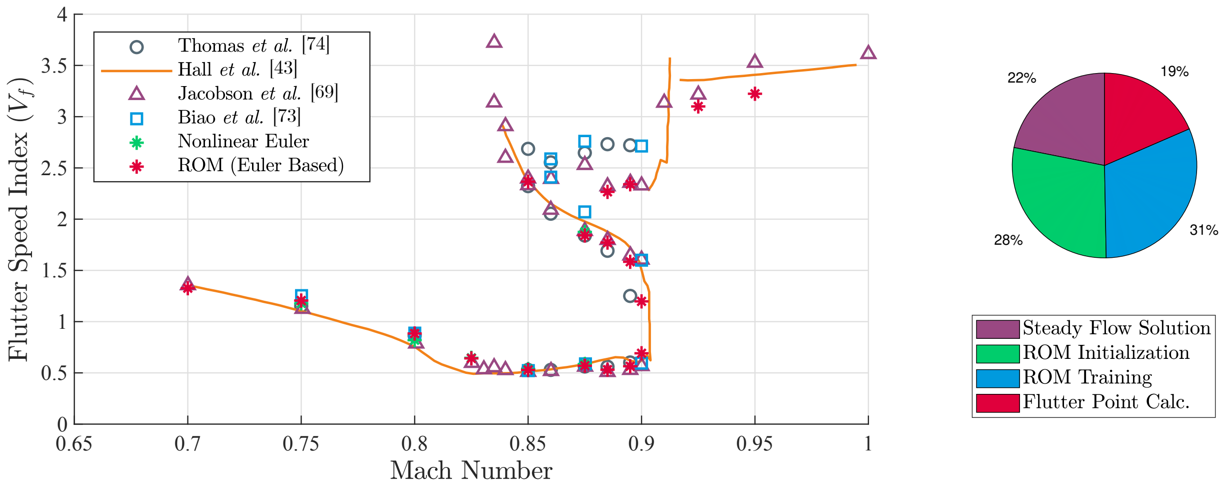


Fig. 7 Flutter boundary (left) and time breakdown (right) for the NACA 64A010 Isogai test case.

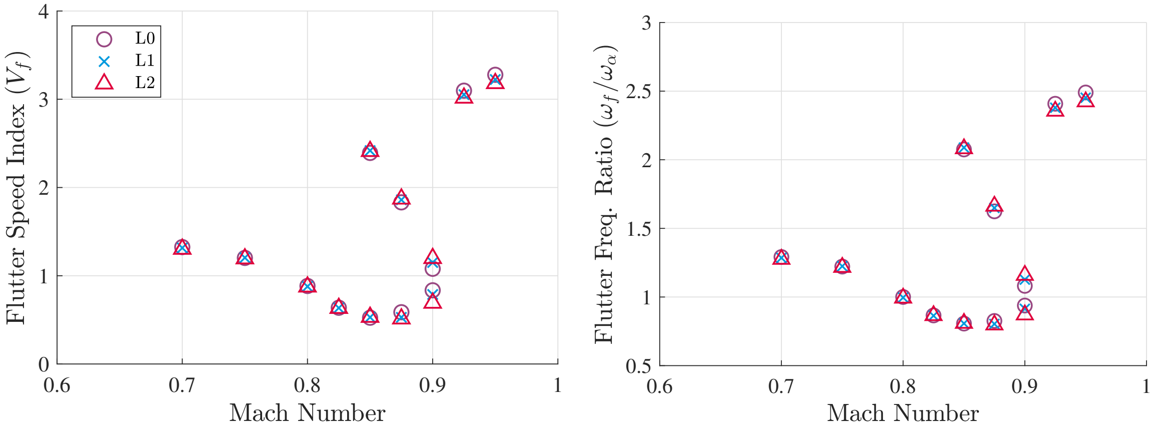
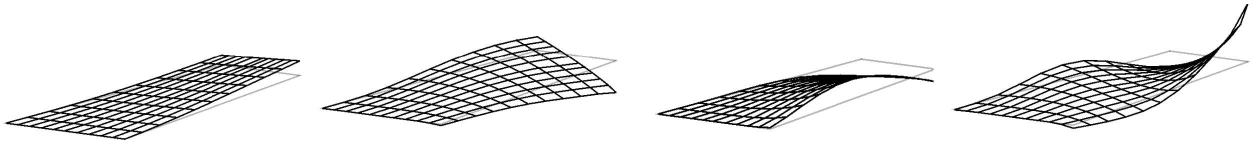


Fig. 8 Grid convergence study for the flutter boundary (left) and flutter frequency (right) for the NACA 64A010 Isogai test case.

L0 with 3.80×10^4 nodes, L1 with 7.21×10^4 nodes, and L2 with 1.47×10^5 nodes. The flutter speed indices and flutter frequency ratios of the grid convergence study are shown in Figure 8. Notably, we see at Mach 0.9 that the flutter speed index changes considerably with the mesh resolution. This is likely due to the effects of the shock wave on the flutter behavior, where the finer mesh better resolves the shock in the flow. The dependence of the predicted flutter speed on the mesh resolution has led other researchers to study the use of mesh adaptation for transonic flutter predictions [69].

C. AGARD 445.6 Test Case

We present results obtained for the dynamic aeroelastic analysis of the AGARD 445.6 test case. Flutter predictions for the wing were originally obtained with wind tunnel testing performed by Yates [67]. The structural model provided by Yates [67] is composed of four natural frequencies and mode shapes, the latter shown in Figure 9. Simulations are performed on the AGARD 445.6 wing geometry using a structured mesh with 1,987,392 nodes subdivided into 160 blocks. All results are obtained using 160 Intel "Skylake" cores at 2.4 GHz.



(a) Mode 1 **(b) Mode 2** **(c) Mode 3** **(d) Mode 4**

Fig. 9 First four mode shapes of the AGARD 445.6 wing structure.

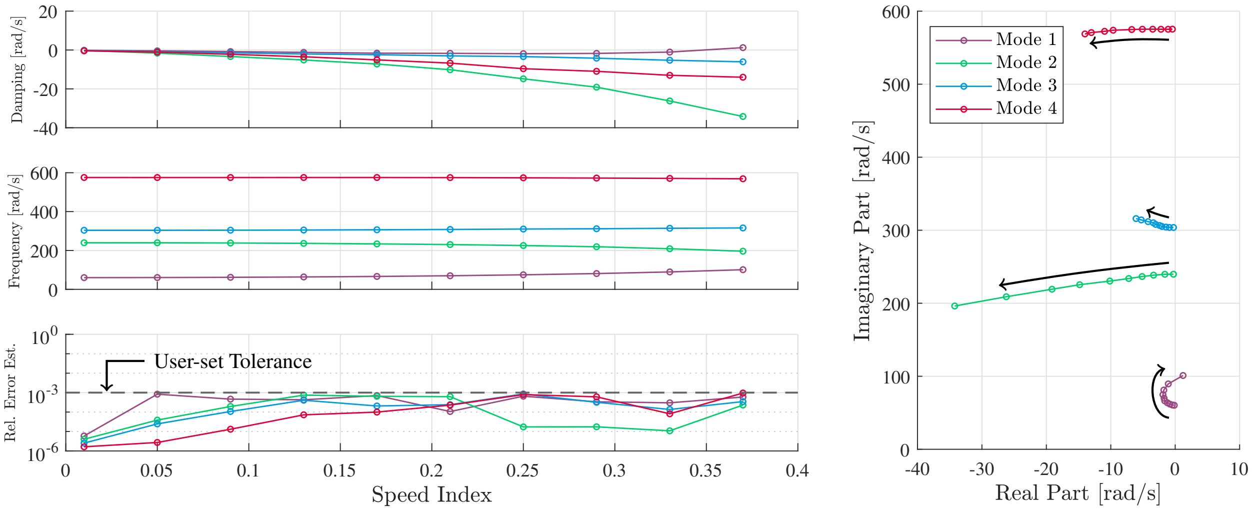


Fig. 10 Aeroelastic mode damping, frequency, and relative estimated error (left), and root locus plot (right) for the AGARD 445.6 wing at Mach 0.901.

1. Aeroelastic Root Locus

Here we present the aeroelastic root locus of the AGARD 445.6 wing model obtained for a Mach number of 0.901. To obtain this root locus, a relative eigenvalue tolerance of

$$\frac{|\eta|}{|\tilde{\lambda}|} \leq 10^{-3}$$

was used. Figure 10 shows the damping, frequency, and estimated errors for the four modes of the AGARD wing at various values of speed indices, along with the root locus of the eigenvalues. From this figure, we observe that all eigenvalue error estimates are below the user-specified relative tolerance. The instability in the system for this case is due to the first mode, which crosses the imaginary axis at a speed index of approximately 0.35.

2. Flutter Boundary

To obtain the flutter boundary for the AGARD 445.6 wing model, aeroelastic ROMs were constructed for each Mach number of interest. For this analysis, a tolerance of

$$\left|\frac{\text{Real}\{\eta\}}{\text{Real}\{\tilde{\lambda}\}}\right| \leq 0.5$$

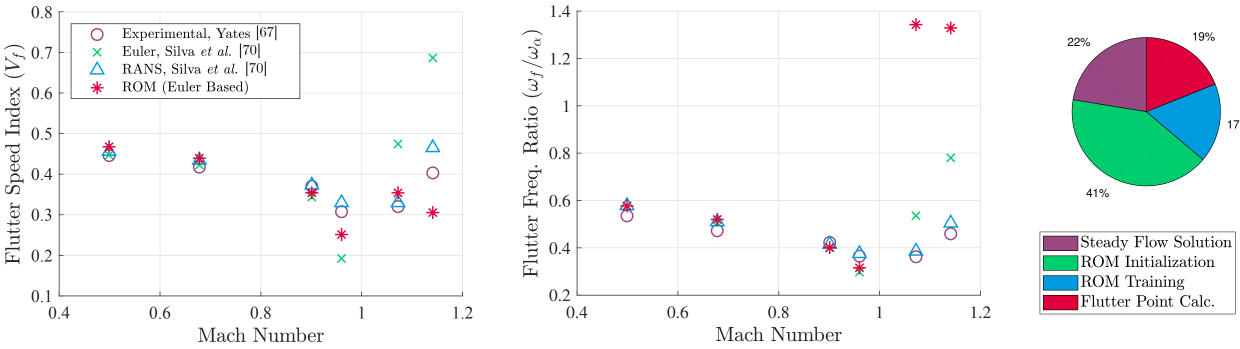


Fig. 11 Flutter boundary (left), flutter frequency ratio (middle), and time breakdown (right) for the AGARD 445.6 wing model.

was used for the eigenvalues. Moreover, a step size of 0.05 in speed index was used for the computation of the root locus, and a relative flutter speed index error tolerance of

$$\frac{\eta_\mathrm{f}}{\tilde{V}_\mathrm{f}} \leq 0.005$$

was used as a stopping criterion.

Figure 11 shows the flutter boundary and flutter frequency ratios obtained using our approach, along with a pie chart showing the time breakdown for each part of the analysis. The flutter boundaries obtained experimentally by Yates [67] and computationally by Silva *et al.* [70] are also included in this figure. As expected, flutter predictions at the lower Mach numbers match well with those obtained experimentally by Yates [67]. At high Mach numbers, namely 0.960, 1.072, and 1.141, it has been shown that the Euler equations do not accurately predict the flutter boundary due to the absence of viscosity [71]. From the flutter frequency ratio plot in Figure 11, we observe that the predicted frequencies at Mach numbers 1.072 and 1.141 are much higher than observed by other approaches. To further analyze this behavior, Figure 12 shows the root locus plots obtained for each Mach number for the AGARD 445.6 wing. At the Mach numbers 1.072 and 1.141, the onset of flutter is due to an instability in the third aeroelastic mode. It has been shown by Silva *et al.* [70] that this instability is an artifact resulting from the use of the Euler equations.

From the time analysis in Figure 11, we can see that computing each point on the flutter boundary requires an average equivalent time of approximately 4.5 steady nonlinear Euler flow evaluations. In this case, the largest portion of time is required for the initialization of the reduced bases.

D. NACA 0012 Benchmark Case

As part of the Benchmark Models Program conducted at NASA Langley, an experimental flutter boundary was obtained for an unswept rectangular wing with a NACA 0012 airfoil [68]. The wing itself is effectively rigid, but is mounted to a pitch and plunge apparatus providing it with two degrees-of-freedoms. The structural model is represented

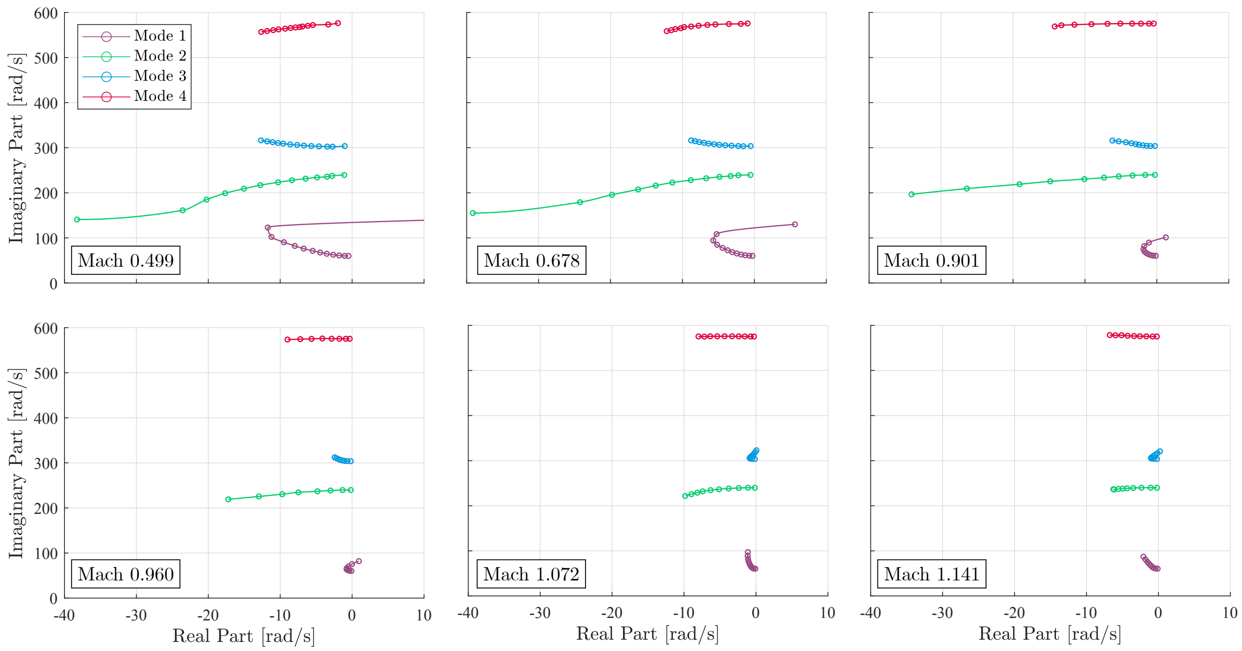


Fig. 12 Aeroelastic root locus plots for the AGARD 445.6 wing test case at various Mach numbers.

in Figure 4, with the center of gravity, elastic axis, and midchord positions all coinciding for this case. The two natural frequencies of the structure are 3.33 Hz for the plunge mode and 5.20 Hz for the pitch mode. Details of the structural properties, test setup, and experimental results are provided by Rivera *et al*. [68]. Here we present a flutter boundary for a three-dimensional wing geometry obtained using a structured mesh consisting of 1,906,624 nodes around the rectangular wing, subdivided into 64 blocks. To obtain the flutter boundary, an eigenvalue tolerance of

$$\left|\frac{\text{Real}\{\eta\}}{\text{Real}\{\tilde{\lambda}\}}\right| \leq 1$$

and a flutter speed index error tolerance of

$$\frac{\eta_\text{f}}{\tilde{V}_\text{f}} \leq 0.01$$

were used.

Rivera *et al*. [68] reported two regions of instabilities. At Mach numbers between 0.3 and 0.82, a conventional flutter boundary with a transonic dip was obtained. At Mach numbers between 0.88 and 0.95, a plunge instability region was observed, where much lower flutter speeds were obtained. In this study, we do not show results for the plunge instability region. The structural model in our work does not include structural damping, which leads to flutter speeds of effectively zero between Mach number 0.88 and 0.95. This is similar to the results obtained by Stanford and Jacobson [72], who obtained a near-zero flutter boundary in the plunge instability region without the inclusion of structural damping.

Figure 13 shows the flutter boundary obtained using the methodology outlined in this paper along with the time

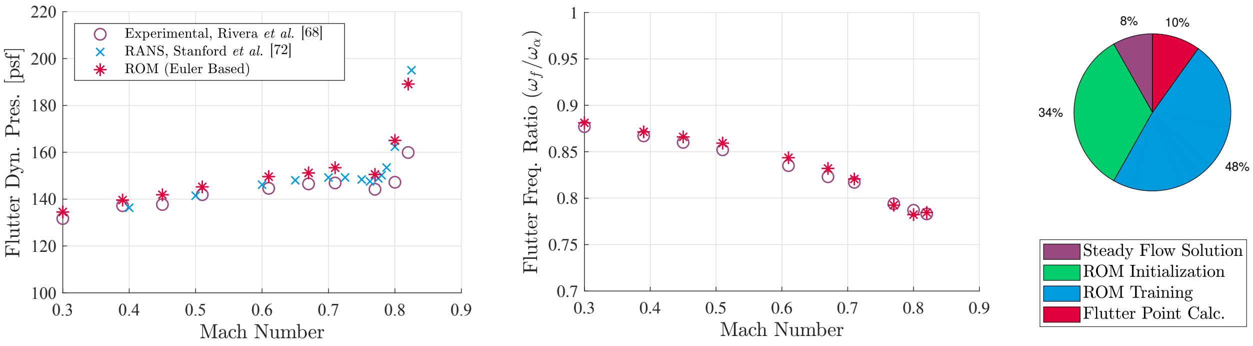


Fig. 13 Flutter boundary (left), flutter frequency ratio (middle), and time breakdown (right) for the NACA 0012 benchmark case.

analysis. The flutter boundary is plotted for dimensional values of dynamic pressure in order to compare with results from the literature. In this figure, experimental results obtained by Rivera *et al*. [68] and RANS-based computational results from Stanford and Jacobson [72] are also plotted. The flutter boundary predicted by our method is slightly above that found experimentally, which is most apparent at the higher Mach numbers. The slight overprediction of the flutter boundary at the higher Mach numbers was also seen by Stanford and Jacobson [72], who speculate one potential cause as the omission of the splitter plate from the computational domain, on which the wing was mounted in the wind tunnel.

From the time analysis shown in Figure 13, one observes that this test case requires more computational time for the flutter analysis when compared to the Isogai and AGARD wing test cases. This is partly due to the low values of aerodynamic damping in the system prior to the flutter point, which leads to increased computational effort by the two-sided Jacobi-Davidson algorithm in order to obtain accurate approximations of the real part of the eigenvalues. In all, each point on the flutter boundary requires an average equivalent time of approximately 12 steady nonlinear Euler flow evaluations.

VII. Conclusions

This paper presents a methodology for dynamic aeroelastic analysis using a projection-based model order reduction approach with error estimation capable of ensuring that the error relative to the high-dimensional model is within a user-defined tolerance. The DWR method is used to provide an estimate of the error of the eigenvalues obtained from the reduced aeroelastic eigenproblem. Moreover, this paper presents a procedure for training the ROM which incorporates the use of the error estimator. Results show that this methodology is fast and effective for obtaining the root locus and the flutter boundary of several test cases. The error estimator is shown to compare well with the exact error for the NACA 64A010 test case by Isogai [66]. This work presents a means by which dynamic aeroelastic analysis can be performed such that a prescribed level of accuracy is obtained relative to the high-dimensional aeroelastic model. For the test cases presented in this work, the cost of computing the flutter point at a given Mach number is equivalent to the cost of approximately 5 to 12 steady nonlinear flow evaluations of the high-dimensional Euler equations.

This work has demonstrated the algorithm's performance for structural models with relatively few degrees of freedom. Future work includes applying the algorithm to more complex models with greater numbers of structural modes. Though a greater number of structural modes will typically increase the computational cost of the algorithm, note that the method can be adjusted to track a subset of the aeroelastic modes in the system, which will help alleviate this additional expense. Additionally, the current method does not account for changing static aeroelastic deflections for nonsymmetric wing geometries. However, introducing varying static aeroelastic deflection capability into the aeroelastic ROM may not be entirely necessary. Studies suggest that ROM-based flutter predictions for nonsymmetric wing geometries are somewhat insensitive to the steady-state static aeroelastic solution about which the ROM is constructed [4]. This insensitivity is likely to depend on the flexibility of the wing; further research on this topic is warranted. Future work also includes using the linearized RANS equations as the aerodynamic model to provide better flutter predictions at high Mach numbers. Additionally, the aeroelastic eigenvalues obtained from this algorithm can be used as a flutter constraint for aerodynamic shape optimization. This can be accomplished by placing a constraint on the real part of the eigenvalues. Significant computational savings may be obtained by using the reduced basis constructed at a previous optimization iteration to form the initial aeroelastic ROM at the current iteration.

Acknowledgements

This research was enabled in part by the financial support provided by Bombardier, the Government of Ontario, the Natural Sciences and Research Council of Canada, and the University of Toronto. Computational resources were provided by SciNet High Performance Computing Consortium, a part of Compute Canada. The authors are extremely grateful for the input and support provided by Masayuki Yano.

References

[1] Reist, T. A., Zingg, D. W., Rakowitz, M., Potter, G., and Banerjee, S., "Multifidelity Optimization of Hybrid Wing–Body Aircraft with Stability and Control Requirements," *Journal of Aircraft*, Vol. 56, No. 2, 2019, pp. 442–456. doi:10.2514/1.C034703.

[2] Ashley, H., "Role of Shocks in the "Sub-Transonic" Flutter Phenomenon," *Journal of Aircraft*, Vol. 17, No. 3, 1980, pp. 187–197. doi:10.2514/3.57891.

[3] Bendiksen, O. O., "Review of Unsteady Transonic Aerodynamics: Theory and Applications," *Progress in Aerospace Sciences*, Vol. 47, No. 2, 2011, pp. 135–167. doi:10.1016/j.paerosci.2010.07.001.

[4] Henshaw, M., Badcock, K., Vio, G., Allen, C., Chamberlain, J., Kaynes, I., Dimitriadis, G., Cooper, J., Woodgate, M., Rampurawala, A., Jones, D., Fenwick, C., Gaitonde, A., Taylor, N., Amor, D., Eccles, T., and Denley, C., "Non-Linear Aeroelastic Prediction for Aircraft Applications," *Progress in Aerospace Sciences*, Vol. 43, No. 4, 2007, pp. 65–137. doi:10.1016/j.paerosci.2007.05.002.

[5] Ballhaus, W. F., and Goorjian, P. M., "Computation of Unsteady Transonic Flows by the Indicial Method," *AIAA Journal*, Vol. 16, No. 2, 1978, pp. 117–124. doi:10.2514/3.60868.

[6] Mohr, R. W., Batina, J. T., and Yang, H. T. Y., "Mach Number Effects on Transonic Aeroelastic Forces and Flutter Characteristics," *Journal of Aircraft*, Vol. 26, No. 11, 1989, pp. 1038–1046. doi:10.2514/3.45877.

[7] Jacobson, K., Stanford, B., Wood, S., and Anderson, W. K., "Flutter Analysis with Stabilized Finite Elements based on the Linearized Frequency-domain Approach," *AIAA SciTech 2020 Forum*, 2020. doi:10.2514/6.2020-0403.

[8] Beran, P., and Lutton, M., "Hopf Bifurcation in Viscous Flows about Airfoils at Low Speeds," *22nd Fluid Dynamics, Plasma Dynamics and Lasers Conference*, Honolulu, Hawaii, 1991. doi:10.2514/6.1991-1807.

[9] Morton, S. A., and Beran, P. S., "Hopf-Bifurcation Analysis of Airfoil Flutter at Transonic Speeds," *Journal of Aircraft*, Vol. 36, No. 2, 1999, pp. 421–429. doi:10.2514/2.2447.

[10] Woodgate, M., and Badcock, K., "Fast Prediction of Transonic Aeroelastic Stability and Limit Cycles," *AIAA Journal*, Vol. 45, No. 6, 2007, pp. 1370–1381. doi:10.2514/1.25604.

[11] Timme, S., Marques, S., and Badcock, K. J., "Transonic Aeroelastic Stability Analysis Using a Kriging-Based Schur Complement Formulation," *AIAA Journal*, Vol. 49, No. 6, 2011, pp. 1202–1213. doi:10.2514/1.J050975.

[12] Hall, K. C., Thomas, J. P., and Clark, W. S., "Computation of Unsteady Nonlinear Flows in Cascades Using a Harmonic Balance Technique," *AIAA Journal*, Vol. 40, No. 5, 2002, pp. 879–886. doi:10.2514/2.1754.

[13] Thomas, J. P., Dowell, E. H., and Hall, K. C., "Modeling Viscous Transonic Limit Cycle Oscillation Behavior Using a Harmonic Balance Approach," *Journal of Aircraft*, Vol. 41, No. 6, 2004, pp. 1266–1274. doi:10.2514/1.9839.

[14] Ekici, K., and Hall, K. C., "Harmonic Balance Analysis of Limit Cycle Oscillations in Turbomachinery," *AIAA Journal*, Vol. 49, No. 7, 2011, pp. 1478–1487. doi:10.2514/1.J050858.

[15] Yao, W., and Marques, S., "Application of a High-order CFD Harmonic Balance Method to Nonlinear Aeroelasticity," *Journal of Fluids and Structures*, Vol. 74, 2017, pp. 427–444. doi:10.1016/j.jfluidstructs.2017.06.014.

[16] Thomas, J., and Dowell, E., "A Fixed Point Iteration Approach for Harmonic Balance Based Aeroelastic Computations," *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Kissimmee, Florida, 2018. doi: 10.2514/6.2018-1446.

[17] He, S., Jonsson, E., Mader, C. A., and Martins, J. R. R. A., "Coupled Newton–Krylov Time-Spectral Solver for Flutter and Limit Cycle Oscillation Prediction," *AIAA Journal*, Vol. 59, No. 6, 2021, pp. 2214–2232. doi:10.2514/1.J059224.

[18] Jadic, I., Hermann, T., Hartley, D., and Giri, J., "Generalized Aerodynamic Forces Based on CFD and Correction Factor Techniques for AGARD Wing 445.6," *19th AIAA Applied Aerodynamics Conference*, Anaheim,California, 2001. doi: 10.2514/6.2001-1208.

[19] Opgenoord, M. M. J., Drela, M., and Willcox, K. E., “Influence of Transonic Flutter on the Conceptual Design of Next-Generation Transport Aircraft,” *AIAA Journal*, Vol. 57, No. 5, 2019, pp. 1973–1987. doi:10.2514/1.J057302.

[20] Thelen, A. S., Leifsson, L. T., and Beran, P. S., “Multifidelity Flutter Prediction Using Local Corrections to the Generalized AIC,” *Aerospace Science and Technology*, Vol. 106, No. 106032, 2020. doi:10.1016/j.ast.2020.106032.

[21] Nguyen, N. T., and Xiong, J., “Transonic Correction to Theodorsen’s Theory for Oscillating Airfoil in Pitch and Plunge Toward Flutter,” *AIAA Scitech 2021 Forum*, Virtual Event, 2021. doi:10.2514/6.2021-1913.

[22] Liu, H., Hu, H., Zhao, Y., and Huang, R., “Efficient Reduced-order Modeling of Unsteady Aerodynamics Robust to Flight Parameter Variations,” *Journal of Fluids and Structures*, Vol. 49, 2014, pp. 728–741. doi:10.1016/j.jfluidstructs.2014.06.015.

[23] Thelen, A., Leifsson, L., and Beran, P., “Aeroelastic Flutter Prediction Using Multifidelity Modeling of the Generalized Aerodynamic Influence Coefficients,” *AIAA Journal*, Vol. 58, No. 11, 2020, pp. 4764–4780. doi:10.2514/1.J059208.

[24] Rumpfkeil, M. P., and Beran, P., “Multi-Fidelity Surrogate Models for Flutter Database Generation,” *Computers & Fluids*, Vol. 197, No. 104372, 2020. doi:10.1016/j.compfluid.2019.104372.

[25] Juang, J.-N., and Pappa, R. S., “An Eigensystem Realization Algorithm for Modal Parameter Identification and Model Reduction,” *Journal of Guidance, Control, and Dynamics*, Vol. 8, No. 5, 1985, pp. 620–627. doi:10.2514/3.20031.

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

[27] Silva, W., “AEROM: NASA’s Unsteady Aerodynamic and Aeroelastic Reduced-Order Modeling Software,” *Aerospace*, Vol. 5, No. 2, 2018, pp. 41–58. doi:10.3390/aerospace5020041.

[28] Liu, H., Gao, X., and Huang, R., “Efficient Reduced-order Aerodynamic Modeling for Fast Prediction of Transonic Flutter Boundary,” *International Journal of Dynamics and Control*, Vol. 8, 2020, pp. 1080–1099. doi:10.1007/s40435-020-00694-z.

[29] Cowan, T. J., Arena, A. S., and Gupta, K. K., “Accelerating Computational Fluid Dynamics Based Aeroelastic Predictions Using System Identification,” *Journal of Aircraft*, Vol. 38, No. 1, 2001, pp. 81–87. doi:10.2514/2.2737.

[30] McNamara, J. J., and Friedmann, P. P., “Flutter Boundary Identification for Time-Domain Computational Aeroelasticity,” *AIAA Journal*, Vol. 45, No. 7, 2007, pp. 1546–1555. doi:10.2514/1.26706.

[31] Zhang, W., Chen, K., and Ye, Z., “Unsteady Aerodynamic Reduced-order Modeling of an Aeroelastic Wing Using Arbitrary Mode Shapes,” *Journal of Fluids and Structures*, Vol. 58, 2015, pp. 254–270. doi:10.1016/j.jfluidstructs.2015.07.007.

[32] Argaman, M., and Raveh, D. E., “Multioutput Autoregressive Aeroelastic System Identification and Flutter Prediction,” *Journal of Aircraft*, Vol. 56, No. 1, 2019, pp. 30–42. doi:10.2514/1.C034789.

[33] Winter, M., and Breitsamter, C., “Neurofuzzy-Model-Based Unsteady Aerodynamic Computations Across Varying Freestream Conditions,” *AIAA Journal*, Vol. 54, No. 9, 2016, pp. 2705–2720. doi:10.2514/1.J054892.

[34] Voitcu, O., and Wong, Y. S., "Neural Network Approach for Nonlinear Aeroelastic Analysis," *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 1, 2003, pp. 99–105. doi:10.2514/2.5019.

[35] Zhang, W., Wang, B., Ye, Z., and Quan, J., "Efficient Method for Limit Cycle Flutter Analysis Based on Nonlinear Aerodynamic Reduced-Order Models," *AIAA Journal*, Vol. 50, No. 5, 2012, pp. 1019–1028. doi:10.2514/1.J050581.

[36] Mannarino, A., and Mantegazza, P., "Nonlinear Aeroelastic Reduced Order Modeling by Recurrent Neural Networks," *Journal of Fluids and Structures*, Vol. 48, 2014, pp. 103–121. doi:10.1016/j.jfluidstructs.2014.02.016.

[37] Benner, P., Gugercin, S., and Willcox, K., "A Survey of Projection-Based Model Reduction Methods for Parametric Dynamical Systems," *SIAM Review*, Vol. 57, No. 4, 2015, pp. 483–531. doi:10.1137/130932715.

[38] Hall, K. C., "Eigenanalysis of Unsteady Flows About Airfoils, Cascades, and Wings," *AIAA Journal*, Vol. 32, No. 12, 1994, pp. 2426–2432. doi:10.2514/3.12309.

[39] Romanowski, M., "Reduced Order Unsteady Aerodynamic and Aeroelastic Models Using Karhunen-Loeve Eigenmodes," *6th Symposium on Multidisciplinary Analysis and Optimization*, Bellevue,Washington, 1996. doi:10.2514/6.1996-3981.

[40] Lumley, J. L., "The Structure of Inhomogeneous Turbulent Flows," *Atmospheric Turbulence and Radio Wave Propagation*, edited by A. M. Yaglom and V. I. Tatarski, Nauka, Moscow, 1967, pp. 166–177.

[41] Holmes, P., Lumley, J. L., Berkooz, G., and Rowley, C. W., *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*, Cambridge University Press, Cambridge,UK, 1996. doi:10.1017/CBO9780511622700.

[42] 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, 2000, pp. 1853–1862. doi:10.2514/2.867.

[43] Thomas, J. P., Dowell, E. H., and Hall, K. C., "Three-Dimensional Transonic Aeroelasticity Using Proper Orthogonal Decomposition-Based Reduced-Order Models," *Journal of Aircraft*, Vol. 40, No. 3, 2003, pp. 544–551. doi:10.2514/2.3128.

[44] Amsallem, D., Tezaur, R., and Farhat, C., "Real-Time Solution of Linear Computational Problems Using Databases of Parametric Reduced-Order Models with Arbitrary Underlying Meshes," *Journal of Computational Physics*, Vol. 326, 2016, pp. 373–397. doi:10.1016/j.jcp.2016.08.025.

[45] Kalashnikova, I., Barone, M., and Brake, M., "A Stable Galerkin Reduced Order Model for Coupled Fluid–Structure Interaction Problems," *International Journal for Numerical Methods in Engineering*, Vol. 95, No. 2, 2013, pp. 121–144. doi:10.1002/nme.4499.

[46] Beran, P. S., Lucia, D. J., and Pettit, C. L., "Reduced-Order Modelling of Limit-Cycle Oscillation for Aeroelastic Systems," *Journal of Fluids and Structures*, Vol. 19, No. 5, 2004, pp. 575–590. doi:10.1016/j.jfluidstructs.2004.04.002.

[47] 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.

[48] Lowe, B. M., and Zingg, D. W., "Efficient Flutter Prediction Using Reduced-Order Modeling," *AIAA Journal*, Vol. 59, No. 7, 2021, pp. 2670–2683. doi:10.2514/1.J060006.

[49] Becker, R., and Rannacher, R., "An Optimal Control Approach to A Posteriori Error Estimation in Finite Element Methods," *Acta Numerica*, Vol. 10, No. 1, 2001, pp. 1–102. doi:10.1017/S0962492901000010.

[50] Fidkowski, K. J., and Darmofal, D. L., "Review of Output-based Error Estimation and Mesh Adaptation in Computational Fluid Dynamics," *AIAA Journal*, Vol. 49, No. 4, 2011, pp. 673–694. doi:10.2514/1.J050073.

[51] Heuveline, V., and Rannacher, R., "A Posteriori Error Control for Finite Element Approximations of Elliptic Eigenvalue Problems," *Advances in Computational Mathematics*, Vol. 15, No. 1-4, 2001, pp. 107–138. doi:10.1023/A:1014291224961.

[52] Sleijpen, G. L., and Van der Vorst, H. A., "A Jacobi–Davidson Iteration Method for Linear Eigenvalue Problems," *SIAM Review*, Vol. 42, No. 2, 2000, pp. 267–293. doi:10.1137/S0036144599363084.

[53] Hochstenbach, M. E., and Sleijpen, G. L., "Two-sided and Alternating Jacobi–Davidson," *Linear Algebra and its Applications*, Vol. 358, No. 1-3, 2003, pp. 145–172. doi:10.1016/S0024-3795(01)00494-3.

[54] Feng, S., and Jia, Z., "A Refined Jacobi-Davidson Method and its Correction Equation," *Computers & Mathematics with Applications*, Vol. 49, No. 2-3, 2005, pp. 417–427. doi:10.1016/j.camwa.2003.01.018.

[55] Benner, P., Hochstenbach, M. E., and Kürschner, P., "Model Order Reduction of Large-scale Dynamical Systems with Jacobi-Davidson Style Eigensolvers," *2011 International Conference on Communications, Computing and Control Applications (CCCA)*, 2011, pp. 1–6. doi:10.1109/CCCA.2011.6031208.

[56] Harder, R. L., and Desmarais, R. N., "Interpolation Using Surface Splines," *Journal of Aircraft*, Vol. 9, No. 2, 1972, pp. 189–191. doi:10.2514/3.44330.

[57] Thomas, P., and Lombard, C., "Geometric Conservation Law and its Application to Flow Computations on Moving Grids," *AIAA Journal*, Vol. 17, No. 10, 1979, pp. 1030–1037. doi:10.2514/3.61273.

[58] Barone, M. F., Kalashnikova, I., Segalman, D. J., and Thornquist, H. K., "Stable Galerkin Reduced Order Models for Linearized Compressible Flow," *Journal of Computational Physics*, Vol. 228, No. 6, 2009, pp. 1932–1946. doi:10.1016/j.jcp.2008.11.015.

[59] Kalashnikova, I., Barone, M. F., Arunajatesan, S., and van Bloemen Waanders, B. G., "Construction of Energy-stable Projection-based Reduced Order Models," *Applied Mathematics and Computation*, Vol. 249, 2014, pp. 569–596. doi:10.1016/j.amc.2014.10.073.

[60] Dowell, E. H., Thomas, J. P., and Hall, K. C., "Transonic Limit Cycle Oscillation Analysis Using Reduced Order Aerodynamic Models," *Journal of Fluids and Structures*, Vol. 19, No. 1, 2004, pp. 17–27. doi:10.1016/j.jfluidstructs.2003.07.018.

[61] Rodden, W. P., and Stahl, B., "A Strip Method for Prediction of Damping in Subsonic Wind Tunnel and Flight Flutter Tests," *Journal of Aircraft*, Vol. 6, No. 1, 1969, pp. 9–17. doi:10.2514/3.43994.

[62] Bangerth, W., and Rannacher, R., *Adaptive Finite Element Methods for Differential Equations*, Birkhäuser, Basel, Switzerland, 2003. doi:10.1007/978-3-0348-7605-6.

[63] van Zyl, L. H., "Use of Eigenvectors in the Solution of the Flutter Equation," *Journal of Aircraft*, Vol. 30, No. 4, 1993, pp. 553–554. doi:10.2514/3.46380.

[64] Eldred, M., Venkayya, V., and Anderson, W., "New Mode Tracking Methods in Aeroelastic Analysis," *AIAA Journal*, Vol. 33, No. 7, 1995, pp. 1292–1299. doi:10.2514/3.12552.

[65] Griewank, A., and Reddien, G., "The Calculation of Hopf Points by a Direct Method," *IMA Journal of Numerical Analysis*, Vol. 3, No. 3, 1983, pp. 295–303. doi:10.1093/imanum/3.3.295.

[66] Isogai, K., "Numerical Study of Transonic Flutter of a Two-Dimensional Airfoil," Tech. Rep. TR-617T, National Aerospace Laboratory, July 1980.

[67] Yates Jr, E. C., "AGARD Standard Aeroelastic Configurations for Dynamic Response. Candidate Configuration I.-Wing 445.6," Tech. Rep. N88-11202, National Aeronautics and Space Administration, 1987.

[68] Rivera, J., JR, Dansberry, B., Bennett, R., Durham, M., and Silva, W., "NACA 0012 Benchmark Model Eexperimental Flutter Results with Unsteady Pressure Distributions," *33rd Structures, Structural Dynamics and Materials Conference*, Dallas, Texas, 1992. doi:10.2514/6.1992-2396.

[69] Jacobson, K., Stanford, B., Kiviaho, J. F., Ozoroski, T. A., Park, M. A., and Chwalowski, P., "Multiscale Mesh Adaptation for Transonic Aeroelastic Flutter Problems," *AIAA Aviation 2021 Forum*, Virtual Event, 2021. doi:10.2514/6.2021-2700.

[70] Silva, W. A., Chwalowski, P., and Perry III, B., "Evaluation of Linear, Inviscid, Viscous, and Reduced-order Modelling Aeroelastic Solutions of the AGARD 445.6 Wing using Root Locus Analysis," *International Journal of Computational Fluid Dynamics*, Vol. 28, No. 3-4, 2014, pp. 122–139. doi:10.1080/10618562.2014.922179.

[71] Lee-Rausch, E., and Batina, J., "Calculation of AGARD Wing 445.6 Flutter Using Navier-Stokes Aerodynamics," *11th Applied Aerodynamics Conference*, Monterey, California, 1993. doi:10.2514/6.1993-3476.

[72] Stanford, B. K., and Jacobson, K. E., "Transonic Aeroelastic Modeling of the NACA 0012 Benchmark Wing," *AIAA Journal*, Vol. 59, No. 10, 2021, pp. 4134–4143. doi:10.2514/1.J060328.

[73] Biao, Z., Zhide, Q., and Chao, G., "Transonic Flutter Analysis of an Airfoil with Approximate Boundary Method," *26th International Congress of the Aeronautical Sciences*, ICAS, Bonn, Germany, 2008.

[74] Thomas, D., Cerquaglia, M., Boman, R., Economon, T., Alonso, J., Dimitriadis, G., and Terrapon, V., "CUPyDO - An Integrated Python Environment for Coupled Fluid-structure Simulations," *Advances in Engineering Software*, Vol. 128, 2019, pp. 69–85. doi:10.1016/j.advengsoft.2018.05.007.