



The balanced mode decomposition algorithm for data-driven LPV low-order models of aeroservoelastic systems



Andrea Iannelli^{a,*}, Urban Fasel^b, Roy S. Smith^a

^a Automatic Control Lab, Department of Electrical Engineering, ETH Zürich, Switzerland

^b Department of Mechanical Engineering, University of Washington, Seattle, USA

ARTICLE INFO

Article history:

Received 4 February 2021

Received in revised form 9 April 2021

Accepted 11 May 2021

Available online 20 May 2021

Communicated by Jérôme Morio

Keywords:

Reduced-order modeling

Aeroservoelasticity

Data-driven

Balanced reduction

Control systems

ABSTRACT

A novel approach to reduced-order modeling of high-dimensional systems with time-varying properties is proposed. It combines the problem formulation of the Dynamic Mode Decomposition method with the concept of balanced realization. It is assumed that the only information available on the system comes from input, state, and output trajectories, thus the approach is fully data-driven. The goal is to obtain an input-output low dimensional linear model which approximates the system across its operating range. Time-varying features of the system are retained by means of a Linear Parameter-Varying representation made of a collection of state-consistent linear time-invariant reduced-order models. The algorithm formulation hinges on the idea of replacing the orthogonal projection onto the Proper Orthogonal Decomposition modes, used in Dynamic Mode Decomposition-based approaches, with a balancing oblique projection constructed from data. As a consequence, the input-output information captured in the lower-dimensional representation is increased compared to other projections onto subspaces of same or lower size. Moreover, a parameter-varying projection is possible while also achieving state-consistency. The validity of the proposed approach is demonstrated on a morphing wing for airborne wind energy applications by comparing the performance against two recent algorithms. Analyses account for both prediction accuracy and closed-loop performance in model predictive control applications.

© 2021 The Author(s). Published by Elsevier Masson SAS. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Data-driven approaches to extract from trajectories of high-dimensional systems, parsimonious models capable of balancing accuracy of the prediction with complexity, are an increasingly popular research topic [1]. In fact, pioneering *ante litteram* contributions to the field, prompted by the goal of identifying low-order structures in complex physical problems such as turbulence, were made in the fluid mechanics and aerodynamics communities [2]. The fundamental idea common to many successful approaches, developed in the wake of these early contributions, is to project the high-dimensional data on a lower dimensional subspace (also constructed from data), such that the most important features of the dynamics are therein preserved. A celebrated example is the Dynamic Mode Decomposition (DMD) approach [3,4], whereby the spectrum of a low-order linear dynamical model approximating the training data is obtained by leveraging the Proper Orthogonal Decomposition (POD) [5] reduction technique. Specifically, the projecting subspace provided by POD is spanned by the left singular

vectors associated with the largest singular values of a data matrix gathered from observations of the dynamics. The exact interpretation of the largest singular values depend on the inner product used to define the data matrix. In standard applications, where the so-called *snapshot* matrix, corresponding to the correlation matrix between the dynamical states, is used, the largest singular values are associated with the modes capturing most of the energy in the system. Thus, the projection onto the lower dimensional subspace preserves the spatial structures with the highest energy content. This criterion for choosing the projection subspace might not always give the best results, as low-energy features can have a large effect on the dynamics, e.g. in the case of non-normal systems, which can be found in some fluid dynamics problems [6]. Moreover, as recently shown in [7], projections onto POD modes are not uniquely defined, due to the arbitrariness of the definition of the state. These findings reinforced the known fact that the quality of the approximation highly depends on the choice of inner product and thus care is required when the projection operator is computed.

Despite these potential shortcomings, POD- and DMD-based methods have been successfully applied in various aerospace and control flow problems [1,8–11]. However, a relatively unexplored

* Corresponding author.

E-mail address: iannelli@control.ee.ethz.ch (A. Iannelli).

application domain of data-driven (or equation-free) reduced-order modeling (ROM) is aeroservoelasticity, where the coupling among multiple disciplines (e.g. aerodynamics, structural dynamics) and components of the system (e.g. wing, actuators) often results in high-order models. Except for a few notable recent exceptions, e.g. the works in [12,13] where nonlinear ROMs have been developed, the standard practice to reduce dimensionality is the use of well established model-based reduction technique [14]. See the work in [15] for an application and further references on this line of research. However, the increasing complexity of the high-fidelity solvers (often made up of distinct sub-solvers for the different disciplines) on one hand, and the potential advantage of recalibrating or directly substituting parts of the code with experimental or flight data on the other, favour the adoption of equation-free strategies. Among the possible reasons for the lack of their application in the field, two important issues are highlighted here.

First, a common feature of the majority of the available approaches is the focus on internal dynamics, meant here as partial or ordinary differential equations without external excitations and with fully observable states. The work in [16] recently extended the DMD framework to controlled systems (DMDc), but the key steps of the algorithm (specifically, the selection of the projecting subspace) do not substantially change. That is, emphasis is not put on preserving the input-output behaviour of the system, which is crucial for control systems.

Second, in aeroservoelastic applications, capturing the variation in the stability and response of the system as the operating conditions change is paramount. This can be done, for example, using the so-called Linear Parameter-Varying (LPV) representation [17], which are of acknowledged benefit for control related tasks [18–21]. Unfortunately, obtaining accurate models featuring low orders is notoriously a difficult task [22], even for the well explored class of model-based approaches [23–25]. One of the most common strategies is to seek low-order linear time-invariant (LTI) representations for frozen-parameter conditions (defining a grid) and then interpolate them for intermediate values of the parameters. Since states, and thus state-space models, are defined up to a nonsingular (similarity) transformation, a correct interpolation requires that the state of each frozen model is defined with respect to the same basis. The need to work with a consistent state-space basis for the local ROMs, required for a correct interpolation [26], poses a challenge for DMD-inspired data-driven approaches. State-consistency will depend indeed on the selection of the projecting subspace. If this changes across the parameter range, as it is the case when one computes the POD modes at each grid point, then state-consistency will not hold in general. Conversely, if the subspace is kept fixed for all the frozen-parameter LTI systems, then accuracy might deteriorate since projection will no longer take place onto the optimal (from an energy point of view) subspace for the considered parameter.

Motivated by the discussion above, the main contribution of this paper is the proposal of a novel equation-free approach to obtain LPV low-order models, namely the Balanced Mode Decomposition (BMD) algorithm. The key idea is to use, instead of an orthogonal projection associated with one subspace (as in standard DMD), an oblique projection, which is associated with two subspaces, namely a basis space and a test space, characterizing the range space and null space of the projection, respectively. Oblique projection, often encountered in model reduction [27] and system identification [28], was also used for model-based reduction of LPV models in [29]. As detailed in Section 3, the oblique projection proposed here can be interpreted, within the context of DMD-type approaches, as an alternative choice to the subspace spanned by the POD modes, and it is instrumental to achieve two favourable properties. The first is that emphasis can be put on the input-output behaviour of the ROM by defining the range and null

spaces of the projection as a function of the controllability and observability Gramians. These objects are well known in the context of model order reduction of linear systems, as they are the main ingredients to perform balanced truncation [14]. This technique consists of transforming the system in balanced coordinates and then removing the states associated with the lowest degree of controllability and observability. In the same spirit, range and null spaces of the proposed oblique projection are defined so that the identified model is (approximately) in balanced coordinates, and thus projection onto lower-dimensional subspaces will preserve the structures in the data matrices that are most observable and controllable. The second favourable property is that the LPV model has a consistent state-space basis [26] across the parameter space without having to sacrifice accuracy. This results from the fact that one subspace (the basis space) is common to all parameters and thus provides a common basis. At the same time, the other subspace (the test space) has no influence on the state's basis, and thus can be chosen different for each parameter in order to alleviate the limitations of a fixed subspace projection.

The second contribution of the paper is to extensively compare the results of the BMD method with two recent extensions of DMD with control. The first algorithm is the algebraic DMDc (aDMDc) [30], which extended DMDc to parameter-varying systems described by algebraic, in addition to differential, equations. Including algebraic constraints is very important, for example, when considering state trajectories generated by aerodynamic solvers capturing unsteady effects, such as in panel methods or unsteady vortex lattice methods [31]. The second algorithm is the input-output reduced-order model (IOROM) approach, proposed in [32] to construct data-driven reduced-order LPV models. Improved ways of defining the low-dimensional subspace such that state-consistency is achieved while preserving accuracy in the (orthogonal) projection were proposed therein. However, the projection operator is the same for all parameters, and is obtained from the POD modes as in standard DMD. An extension of IOROM to handle algebraic constraints is also developed here in order to allow for a fair comparison. The algorithms are tested on a high-fidelity, fluid-structure interaction (FSI) numerical model of an airborne wind energy (AWE) morphing wing. The FSI simulator is described in [33] and the wing was analyzed in detail in [34]. Airborne wind energy and morphing wings are paradigmatic examples of application domains where the system's response originates from complex interactions across different domains, and thus could benefit from equation-free approaches. The first type of comparison investigates the accuracy of the reduced-order models to predict various outputs of the wing as the size of the model is decreased. In a second set of analyses, models featuring different orders are used by a model predictive control (MPC) algorithm to track predefined trajectories of the airborne wind energy system with the goal of gaining insight into the trade-off between size and performance, the latter evaluated by simulating the FSI solver in closed-loop with the MPC controller. Preliminary results of the work were presented in [35].

Fig. 1 shows a conceptual representation of the proposed data-driven ROM framework.

2. Data-driven reduced-order modeling

This section provides background material on the tools and concepts relevant to the reduced-order modeling algorithm proposed in this work. In Section 2.1 the general data-driven low-order modeling problem is presented. Section 2.2 reviews the algebraic DMD with Control (aDMDc) [30], and Section 2.3 reports on the input-output reduced-order model (IOROM) [32]. These are the two ROM algorithms from the literature used for comparison in this paper.

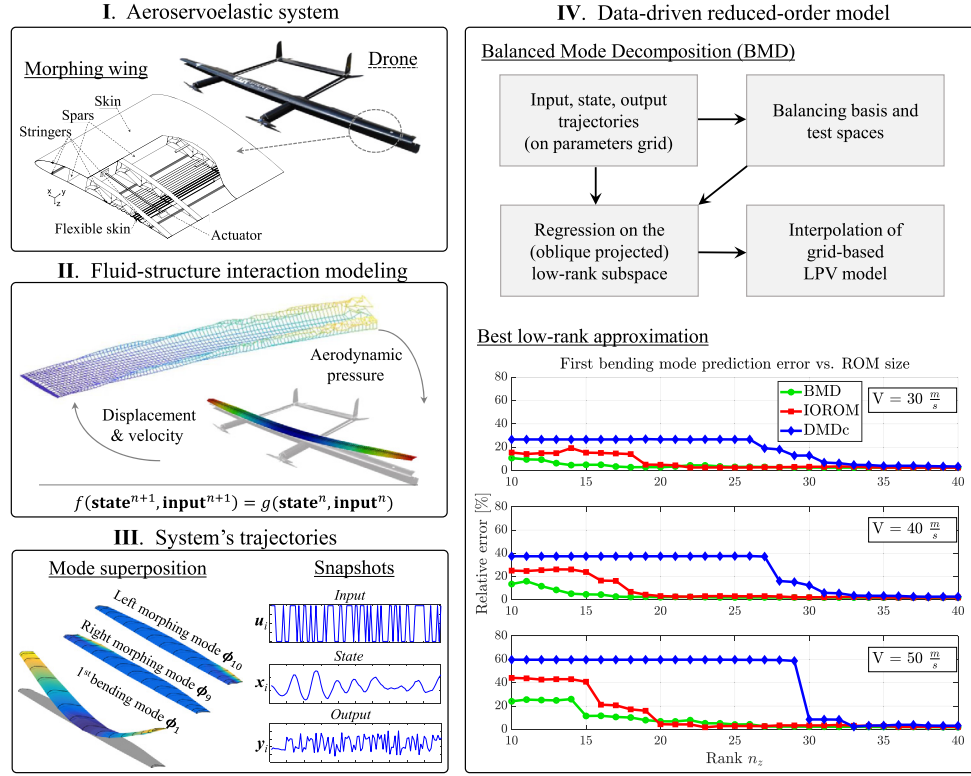


Fig. 1. Overview of the algorithm and its proposed application: **I.** illustrative aeroservoelastic test case [34]; **II.** typical fluid-structure interaction problem; **III.** system characterized uniquely by its states, inputs, and outputs trajectories; **IV.** sketch of the newly proposed BMD algorithm and comparison with two other algorithms.

2.1. Problem statement and preliminaries

The starting point is a generic discrete-time nonlinear parameter-varying model which can be used to describe typical control systems, such as aeroservoelastic systems modelled by FSI solvers

$$\begin{aligned} x_{k+1} &= f(x_k, u_k, \rho_k), \\ y_k &= h(x_k, u_k, \rho_k), \end{aligned} \quad (1)$$

where $x \in \mathbb{R}^{n_x}$, $u \in \mathbb{R}^{n_u}$, $y \in \mathbb{R}^{n_y}$ are the state, input and output, and $\rho: \mathbb{Z} \rightarrow \mathbb{R}^{n_\rho}$ is a vector of time-varying parameters defining the operating conditions of the system. The problem of finding an LPV low-order approximation of (1) can be divided into two phases: first, *local* LTI approximations for frozen values of ρ in a pre-defined grid $\{\rho^j\}_{j=1}^{n_g}$ are computed; then, an LPV model is obtained through interpolation. The following discussion is concerned with the former phase.

It is assumed that for each frozen value $\bar{\rho}$ there exists an equilibrium (or trim) point characterized by the tuple $(\bar{x}(\bar{\rho}), \bar{u}(\bar{\rho}), \bar{y}(\bar{\rho}))$ such that

$$\begin{aligned} \bar{x}(\bar{\rho}) &= f(\bar{x}(\bar{\rho}), \bar{u}(\bar{\rho}), \bar{\rho}), \\ \bar{y}(\bar{\rho}) &= h(\bar{x}(\bar{\rho}), \bar{u}(\bar{\rho}), \bar{\rho}). \end{aligned}$$

The deviation vectors $\tilde{x}_k := x_k - \bar{x}(\bar{\rho})$, $\tilde{u}_k := u_k - \bar{u}(\bar{\rho})$, and $\tilde{y}_k := y_k - \bar{y}(\bar{\rho})$ can then be used as states of an LTI approximation of the system around the equilibrium:

$$\tilde{x}_{k+1} = A(\bar{\rho})\tilde{x}_k + B(\bar{\rho})\tilde{u}_k, \quad (2a)$$

$$\tilde{y}_k = C(\bar{\rho})\tilde{x}_k + D(\bar{\rho})\tilde{u}_k, \quad (2b)$$

where $(A(\bar{\rho}), B(\bar{\rho}), C(\bar{\rho}), D(\bar{\rho}))$ is a state-space representation completely describing the linearization about the trim point associated with $\bar{\rho}$. The dependence of local (i.e. related to LTI approximations for frozen values of ρ) quantities on the parameter

ρ will be dropped in the remainder. It is understood that they depend on the particular value of the parameter considered. Instead, the subscript ρ will be used when discussing LPV models.

In the data-driven setting, the only information on the system comes from input, state, and output trajectories $\{x_k, u_{k-1}, y_{k-1}\}_{k=1}^{n_s}$ of length n_s . After having subtracted from them the corresponding trim values, these trajectories can be used to form the following snapshot matrices

$$\begin{aligned} X_0 &= [x_0 - \bar{x} \quad x_1 - \bar{x} \quad \dots \quad x_{n_s-1} - \bar{x}] \in \mathbb{R}^{n_x \times n_s}, \\ X_1 &= [x_1 - \bar{x} \quad x_2 - \bar{x} \quad \dots \quad x_{n_s} - \bar{x}] \in \mathbb{R}^{n_x \times n_s}, \\ U_0 &= [u_0 - \bar{u} \quad u_1 - \bar{u} \quad \dots \quad u_{n_s-1} - \bar{u}] \in \mathbb{R}^{n_u \times n_s}, \\ U_1 &= [u_1 - \bar{u} \quad u_2 - \bar{u} \quad \dots \quad u_{n_s} - \bar{u}] \in \mathbb{R}^{n_u \times n_s}, \\ Y_0 &= [y_0 - \bar{y} \quad y_1 - \bar{y} \quad \dots \quad y_{n_s-1} - \bar{y}] \in \mathbb{R}^{n_y \times n_s}. \end{aligned} \quad (3)$$

The notation $[X_0; U_0]$ will denote the operation of stacking row-wise two matrices X_0 and U_0 .

The first goal is to obtain a linear time-invariant low-order approximation of (2), that is

$$\begin{aligned} \tilde{z}_{k+1} &= F\tilde{z}_k + G\tilde{u}_k, \\ \tilde{y}_k &= H\tilde{z}_k + D\tilde{u}_k, \end{aligned}$$

where $\tilde{z} \in \mathbb{R}^{n_z}$ and $n_z \ll n_x$. Once this is available, the family of frozen LTI systems, or directly the signals of interest, are interpolated so that the response of the system is available at each value ρ_k for a generic time-varying trajectory of the parameter.

2.2. Algebraic dynamic mode decomposition with control algorithm

The algebraic Dynamic Mode Decomposition with Control (aD-MDC) algorithm was recently proposed in [30] to extend the DMDc

algorithm to systems described by algebraic-differential equations. The DMDc algorithm from [16] is first briefly reviewed. This algorithm seeks a data-driven approximation of the matrices involved in the state equation (2) by means of two truncated singular value decompositions (SVD) of the snapshot matrices. The first one is

$$[X_0; U_0] = U \Sigma V^\top \cong U_r \Sigma_r V_r^\top, \quad (4)$$

where the subscript r denotes a truncation of the SVD decomposition of order r (obtained by keeping only the r largest singular values in the decomposition). Note that the value of r does not define the size of the final reduced-order model, and it could be set for example by using the hard threshold criterion suggested in [36]. The effect of choosing r on the accuracy of the model will be discussed in the result section. The second truncated SVD is computed from the snapshot matrix X_1

$$X_1 = \hat{U} \hat{\Sigma} \hat{V}^\top \cong \hat{U}_{n_z} \hat{\Sigma}_{n_z} \hat{V}_{n_z}^\top, \quad (5)$$

where the columns of \hat{U}_{n_z} are also called POD modes of X_1 and are used for the projection onto a lower dimensional space. The selection of n_z defines the size of the reduced-order model. The thresholds used in (4) and (5) should be chosen such that $r > n_z$ [16].

An approximation of the high-order matrices appearing in (2) can be formulated in terms of the truncated SVD (4) and the snapshot matrix X_1 as

$$[A \ B] = X_1 V_r^\top \Sigma_r^{-1} U_r^\top. \quad (6)$$

Then, a low-order approximation is obtained by projecting (6) onto the set of POD modes by making use of (5)

$$[F \ G] = \left[\hat{U}_{n_z}^\top A \hat{U}_{n_z} \quad \hat{U}_{n_z}^\top B \right].$$

Therefore, the low-order model obtained by DMDc (which only includes the state equation) is

$$\tilde{z}_{k+1} = F \tilde{z}_k + G \tilde{u}_k,$$

where $\tilde{z} \in \mathbb{R}^{n_z}$ is the state of the low-order model and the high-order state can be recovered by $\tilde{x} = \hat{U}_{n_z} \tilde{z}$.

The aDMDc algorithm [30] builds on the DMDc approach and addresses the presence of algebraic constraints in the dynamic equations which might arise when considering unsteady aerodynamics features. Specifically, the morphing wing analyzed in [30] is described by an FSI solver that implements a 3D panel method with a free evolving wake inspired by the method in [31]. This leads to a dependence of the states' evolution on the inputs at the next time step. Therefore, a slightly different starting point from the general one presented in (1) has to be considered, namely

$$\begin{aligned} g(x_{k+1}, u_{k+1}) &= f(x_k, u_k, \rho_k), \\ y_k &= h(x_k, u_k, \rho_k), \end{aligned} \quad (7)$$

where g is in general a nonlinear function taking into account the dependence of the states on the control inputs at the next time step. This dependence results from algebraic equations relating the doublet strengths (aerodynamics states) and downwash (function of the other states and the control inputs). This effect is sometimes accounted for with artificial aerodynamic states by simply changing the feedthrough matrix to the outputs. However, to correctly capture the evolution of the states it is important to formulate the problem as stated in (7). The reader is referred to [30] for further discussion on this aspect.

The proposed LTI representation of the system accounting for the algebraic constraints due to the unsteady aerodynamics is

$$\tilde{x}_{k+1} = A \tilde{x}_k + B \tilde{u}_k + R \tilde{u}_{k+1},$$

where, as in DMDc, the objective is to find a low-order approximation for the state equation only.

The only difference with respect to DMDc is that now the first SVD decomposition is computed with respect to the snapshot matrices X_0 , U_0 , and U_1 , that is

$$[X_0; U_0; U_1] = U \Sigma V^\top \cong U_r \Sigma_r V_r^\top.$$

And the high-order matrices are thus approximated by

$$[A \ B \ R] = X_1 V_r^\top \Sigma_r^{-1} U_r^\top.$$

A low-order approximation is then obtained by projecting (6) onto the same set of POD modes used in DMDc (5)

$$[F \ G \ L] = \left[\hat{U}_{n_z}^\top A \hat{U}_{n_z} \quad \hat{U}_{n_z}^\top B \quad \hat{U}_{n_z}^\top R \right].$$

This procedure results in the aDMDc low-order model

$$\tilde{z}_{k+1} = F \tilde{z}_k + G \tilde{u}_k + L \tilde{u}_{k+1}, \quad (8)$$

where the high-order state can again be obtained from $\tilde{x} = \hat{U}_{n_z} \tilde{z}$.

The approach proposed in the parametrically varying version of the aDMDc algorithm is to use a different set of POD modes for each value of ρ in the grid $\{\rho^j\}_{j=1}^{n_g}$. The frozen LTI models (8) are then simulated simultaneously, the relative states are lifted to the high-order ones using the corresponding projection matrices (e.g. $\hat{U}_{n_z}(\rho^j)$ for the model corresponding to the j th element in the parameter space), and the state corresponding to the desired value of ρ is obtained by interpolating the high-dimensional states. A first consequence of this approach is that the frozen LTI models (8) do not have a consistent basis for the state, because the basis of the state is determined by the POD modes, which change across the parameter grid. While this has the advantage of projecting over POD modes specifically computed for a particular value of ρ , it also requires running in parallel all of the low-order models. Moreover, this algorithm does not provide an LPV model and thus the use of LPV robust control design strategies is precluded [21]. While other control techniques, such as model predictive control, can still be successfully used (see Section 4.4), the necessity to run in parallel, multiple low-order models, is a drawback of the method when targeting real-time applications.

2.3. Input-output reduced-order model algorithm

The input-output reduced-order model (IOROM) algorithm was proposed in [32] to compute a family of state-consistent data-driven low order LTI state-space models (including the output equation) which can be directly parameterized by the vector ρ .

Consider first the case when there is no parameter dependence (or equivalently, ρ is fixed). Drawing inspiration from the interpretation of DMD as linear dynamics fitting [4], the main idea is that, given the snapshot matrices (3), the matrices (A, B, C, D) defining (2) can be obtained by solving the following least-squares problem

$$\min_{A, B, C, D} \left\| \begin{bmatrix} X_1 \\ Y_0 \end{bmatrix} - \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} X_0 \\ U_0 \end{bmatrix} \right\|_F^2, \quad (9)$$

where the subscript F denotes the Frobenius norm of a matrix. Without appropriate regularization, this problem would be ill-posed for high-dimensional systems ($n_x \gg 1$). Most importantly, even if (9) was solved accurately, it would not provide a low dimensional representation of the system. For these reasons, an orthogonal projection of the state onto a low dimensional subspace of dimension n_z is performed by introducing the projection

matrix $Q \in \mathbb{R}^{n_x \times n_z}$, where $Q^\top Q = I_{n_z}$, such that the orthogonal projection of \tilde{x} on an n_z -dimensional subspace is given by $Q Q^\top \tilde{x}$. Equivalently, one can think that the original state is approximated by $\tilde{x} \cong Q \tilde{z}$ for some reduced-order state (or coefficient vector) $\tilde{z} \in \mathbb{R}^{n_z}$. This results in the following low-order state-space model

$$\begin{aligned} \tilde{z}_{k+1} &= (Q^\top A Q) \tilde{z}_k + (Q^\top B) \tilde{u}_k, \\ \tilde{y}_k &= (C Q) \tilde{z}_k + D \tilde{u}_k. \end{aligned} \quad (10)$$

The vector $\tilde{z} = Q^\top \tilde{x} \in \mathbb{R}^{n_z}$ can thus be interpreted as the state of the low-rank approximation of (2)

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \approx \begin{bmatrix} Q F Q^\top & Q G \\ H Q^\top & D \end{bmatrix} = \begin{bmatrix} Q & 0 \\ 0 & I_{n_y} \end{bmatrix} \begin{bmatrix} F & G \\ H & D \end{bmatrix} \begin{bmatrix} Q^\top & 0 \\ 0 & I_{n_u} \end{bmatrix}.$$

The projection matrix Q is constructed from the POD modes of X_0 , that is

$$Q = U_{n_z}, \quad (11)$$

where $X_0 \cong U_{n_z} \Sigma_{n_z} V_{n_z}^\top$.

The least-squares problem giving (F, G, H, D) is then

$$\min_{F, G, H, D} \left\| \begin{bmatrix} X_1 \\ Y_0 \end{bmatrix} - \begin{bmatrix} Q & 0 \\ 0 & I_{n_y} \end{bmatrix} \begin{bmatrix} F & G \\ H & D \end{bmatrix} \begin{bmatrix} Q^\top & 0 \\ 0 & I_{n_u} \end{bmatrix} \begin{bmatrix} X_0 \\ U_0 \end{bmatrix} \right\|_F^2, \quad (12)$$

whose solution is

$$\begin{bmatrix} F & G \\ H & D \end{bmatrix}_{opt} = \begin{bmatrix} Q^\top X_1 \\ Y_0 \end{bmatrix} \begin{bmatrix} Q^\top X_0 \\ U_0 \end{bmatrix}^\dagger, \quad (13)$$

where \dagger denotes the pseudo-inverse of a matrix. It is worth noting that the reduced-order model given by the IOROM algorithm is qualitatively similar to the one associated with DMDC. The main difference (besides the output equation, not considered in DMDC) is that the pseudo-inverse operation, which also amounts to an SVD decomposition and thus is conceptually similar to (4), is done here directly on the projected snapshot matrices. This is different than what is done in DMDC, where the SVD decomposition (4) is applied to X_0 and U_0 . A minor difference is also that the POD modes are computed here with respect to X_0 instead of X_1 .

In the parameter-varying case, the regression problem (13) is solved at each value of the parameter grid $\{\rho^j\}_{j=1}^{n_g}$ by taking the corresponding snapshot matrices $\{X_0(\rho^j), X_1(\rho^j), U_0(\rho^j), Y_0(\rho^j)\}_{j=1}^{n_g}$. By always using the same projection matrix Q when computing the low-order models at different ρ , state-consistency is automatically guaranteed because the orthogonal projection has the same range space. An LPV reduced-order model is then obtained by interpolating (13) across the parameter's range. That is

$$\begin{aligned} \tilde{z}_{k+1} &= F_{\rho_k} \tilde{z}_k + G_{\rho_k} \tilde{u}_k + (\bar{z}(\rho_k) - \bar{z}(\rho_{k+1})), \\ \tilde{y}_k &= H_{\rho_k} \tilde{z}_k + D_{\rho_k} \tilde{u}_k, \end{aligned}$$

where $(F_{\rho_k}, G_{\rho_k}, H_{\rho_k}, D_{\rho_k})$ are obtained by interpolating the corresponding matrices for the value of ρ at timestep k . Note that the low-order state is $\tilde{z}_k = z_k - \bar{z}(\rho_k)$, where the trim point $\bar{z}(\rho_k) = Q^\top \tilde{x}(\rho_k)$ can change as a function of ρ . The term $(\bar{z}(\rho_k) - \bar{z}(\rho_{k+1}))$ is added to correctly take into account this effect [32].

Since the choice of a fixed projection matrix is typically associated with less accuracy, two strategies are proposed in [32] to alleviate this issue. The first consists of using in the decomposition (11) a *fat* snapshot matrix X_0 obtained by stacking column-wise the snapshot matrices of multiple parameters. This matrix will feature $n_s n_g$ columns, which can result in computationally expensive calculations when this number is large. The second, less accurate but more practical in case several grid points are analyzed,

consists of iteratively building Q by incrementally processing the snapshot data from each grid point in a similar fashion to the Gram-Schmidt orthogonalization procedure. The former strategy is used here when showing results for the IOROM algorithm, together with a linear interpolation of the state-space matrices.

3. Balanced mode decomposition with oblique projection algorithm

This section presents the technical aspects of the Balanced Mode Decomposition (BMD) algorithm proposed in this paper. Section 3.1 clarifies the goals and the novelty of the contribution with respect to previous works. Section 3.2 presents the algorithm and Section 3.3 details the computation of the subspaces defining the oblique projection. Finally, Section 3.4 presents a version of the algorithm which can handle algebraic constraints and thus allows the analyses in Section 4 of the morphing wing with an unsteady aerodynamics model.

3.1. Novelty and connections with prior work

The main motivation for the proposal of the BMD algorithm for data-driven LPV low-order modeling is to address two limitations of recent extensions of the celebrated DMD method to input-output parameter-varying models. The first one concerns the use of Q (i.e. the subspace spanned by the most energetic POD modes according to the standard choice of inner product as unweighted scalar product) for the projection of the higher-order dynamics, which is suboptimal as also acknowledged by the authors of [32]. In the input-output context, a subspace typically providing lower input-output errors with respect to the others having same size n_z is the one where the system's state is in balanced coordinates [37]. This is indeed the rationale behind balanced truncation, which consists of removing the states corresponding to the smallest $n_x - n_z$ Hankel singular values [38]. The justification for this is that the sum of the Hankel singular values provides a lower bound, and for systems in balanced coordinates, an upper bound on the error of the approximation achieved by removing system's states. Even though not guaranteed to be optimal, balanced truncation is a very effective tool in model-based order reduction [14,39]. These ideas are used here to propose a new projection operator for the high-dimensional state.

Whereas the aspect mentioned above is relevant also for the case where the sought model is an LTI (i.e. when there is no parameter dependence), the second one is specific to the LPV setting. Precisely, the second limitation addressed by the BMD algorithm concerns how to handle state-consistency across the frozen models in order to estimate the system's response at intermediate points in the parameter grid. In the currently available approaches, this is addressed in two possible ways. When state-consistency is not fulfilled, all reduced-order models are run in parallel by interpolating directly the high-dimensional lifted state. This is the case of aD-MDC, and while it has the advantage that the projection operators are parameter-dependent (i.e. at each parameter's value one can use a different set of POD modes), an LPV model is not available and moreover computational efficiency might be compromised. On the other hand, a parameter-independent projection matrix for all frozen models can be used in order to guarantee state-consistency. This is the case for the IOROM algorithm, and it has the drawback that, whereas the orthonormal basis associated with the n_z most energetic modes will be in general different at each value of ρ , a fixed one that is common to all parameters is used.

The central idea to overcome both of the aforementioned issues is to replace the orthogonal projection employed in standard POD-based approaches by an oblique projection. Given $V \in \mathbb{R}^{n_x \times n_z}$ and $W \in \mathbb{R}^{n_x \times n_z}$, such that W is bi-orthogonal to V , i.e. $W^\top V = I_{n_z}$,

the oblique projection of \tilde{x} is given by $\Pi\tilde{x}$, where $\Pi = VW^\top$. As a result, the original state is approximated as $\tilde{x} \cong V\tilde{z}$ (where, as before, $\tilde{z} \in \mathbb{R}^{n_z}$ is the reduced-order state), and the component of \tilde{x} that is eliminated by the projection is in the nullspace of Π . As opposed to the orthogonal projection, which is characterized by a single subspace (the one spanned by the columns of Q), the oblique projection is defined by two subspaces: the basis space (spanned by the columns of V), such that the projection of \tilde{x} lies in the span of V ; and the test space (spanned by the columns of W), such that the projection $V\tilde{z}$ has zero error within it, i.e. $W^\top(\tilde{x} - V\tilde{z}) = 0$. Technically, the high-dimensional state vector is projected along the orthogonal complement of the subspace spanned by the columns of W onto a subspace spanned by the columns of V . In practice, this means that what is lost by projecting \tilde{x} (i.e. the nullspace of the projection) is orthogonal to W , and the state basis only depends on V . The two issues discussed above are then addressed by: computing V and W from the *empirical* controllability and observability Gramians of the system (which leads to a model-free balanced truncation); employing a fixed V and a parameter-dependent W . Since V by definition defines the basis of the vector space where the state of each model is defined, this basis will be common to all the local state-space models, and thus state-consistency is guaranteed.

The idea of using an oblique projection for LPV model-order reduction was first proposed in [29]. Therein, the setting where a model of the system is available (in the form of high-order state-space models) is considered, and thus both the construction of V and W , and the computation of the low-order model, is model-based. In the data-driven ROM literature, balancing concepts are used in two important techniques, namely Balanced POD (BPOD) [40] and the Eigensystem Realization Algorithm (ERA) [41]. The former is only partially equation-free: the controllability Gramian is computed from data, while for the observability Gramian an adjoint simulation model is needed. Additionally, the high-order state-space matrices are required for the balanced projection. For the case of ERA, a balanced model comes from impulse response simulations of the model in the spirit of system identification algorithms from realization theory [42]. The ERA algorithm is closely related to BPOD, as it can be interpreted as a data-driven balanced truncation. An important difference is that ERA provides only the reduced-order model and not the balancing transformation, namely the set of vectors known as balancing and adjoint modes in BPOD. These modes are the counterpart of the basis and test space in BMD, respectively, and are a desirable output of a ROM algorithm as they show the most important spatio-temporal structures in the dynamics. In an aeroservoelastic setting, this can provide insights into efficient design solutions. It is recalled that BPOD can be interpreted as a special case of POD when impulse responses are used to build the snapshot matrices and the observability Gramian is used as inner product [40]. Conceptually (because in practice the algorithm formulation is articulated in a different way), it can be helpful to think of BMD as a version of DMD that makes use of this special case of POD.

3.2. BMD regression problem

We consider first the frozen-parameter case and, by virtue of the previously discussed oblique projection, propose the following low-order LTI system model

$$\begin{aligned}\tilde{z}_{k+1} &= (W^\top AV)\tilde{z}_k + (W^\top B)\tilde{u}_k, \\ \tilde{y}_k &= (CV)\tilde{z}_k + D\tilde{u}_k,\end{aligned}\quad (14)$$

where the computation of the balancing basis V and test spaces W from system's trajectories will be detailed in Section 3.3. The vector $\tilde{z} = W^\top\tilde{x} \in \mathbb{R}^{n_z}$ can thus be interpreted as the state associated with the following low-rank approximation of (2)

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \approx \begin{bmatrix} VFW^\top & VG \\ HW^\top & D \end{bmatrix} = \begin{bmatrix} V & 0 \\ 0 & I_{n_y} \end{bmatrix} \begin{bmatrix} F & G \\ H & D \end{bmatrix} \begin{bmatrix} W^\top & 0 \\ 0 & I_{n_u} \end{bmatrix}.\quad (15)$$

The matrices (F, G, H, D) can then be obtained with the following least-squares problem

$$\min_{F,G,H,D} \left\| \begin{bmatrix} X_1 \\ Y_0 \end{bmatrix} - \begin{bmatrix} V & 0 \\ 0 & I_{n_y} \end{bmatrix} \begin{bmatrix} F & G \\ H & D \end{bmatrix} \begin{bmatrix} W^\top & 0 \\ 0 & I_{n_u} \end{bmatrix} \begin{bmatrix} X_0 \\ U_0 \end{bmatrix} \right\|_F^2,\quad (16)$$

which has solution

$$\begin{bmatrix} F & G \\ H & D \end{bmatrix}_{opt} = \begin{bmatrix} W^\top X_1 \\ Y_0 \end{bmatrix} \begin{bmatrix} W^\top X_0 \\ U_0 \end{bmatrix}^\dagger.\quad (17)$$

To build a low-order LPV model, snapshot matrices are first collected for the values of the parameter in the grid $\{\rho^j\}_{j=1}^{n_g}$, and the least-squares problem (16) is solved at each grid point. Crucially, the test space W is allowed to be a function of ρ . This leads to the following solution for the reduced-order models in the grid

$$\begin{bmatrix} F(\rho^j) & G(\rho^j) \\ H(\rho^j) & D(\rho^j) \end{bmatrix}_{opt} = \begin{bmatrix} W^\top(\rho^j)X_1(\rho^j) \\ Y_0(\rho^j) \end{bmatrix} \begin{bmatrix} W^\top(\rho^j)X_0(\rho^j) \\ U_0(\rho^j) \end{bmatrix}^\dagger,\quad (18)$$

where the dependence on ρ of the local quantities has been here explicitly reported in order to clearly point out that all the objects involved in the construction of the low-order state-space models are parameter-varying.

The BMD LPV reduced-order model is then obtained by interpolating the frozen matrices (18) across the parameter's range

$$\begin{aligned}\tilde{z}_{k+1} &= F_{\rho_k}\tilde{z}_k + G_{\rho_k}\tilde{u}_k + (\tilde{z}(\rho_k) - \tilde{z}(\rho_{k+1})), \\ \tilde{y}_k &= H_{\rho_k}\tilde{z}_k + D_{\rho_k}\tilde{u}_k,\end{aligned}\quad (19)$$

where, as in IOROM, $(F_{\rho_k}, G_{\rho_k}, H_{\rho_k}, D_{\rho_k})$ are obtained by interpolating the corresponding matrices for the value of ρ at timestep k , and the term $(\tilde{z}(\rho_k) - \tilde{z}(\rho_{k+1}))$ takes into account the fact that the equilibrium point associated with each ρ is in general different, and $\tilde{z}(\rho_k) = W^\top\tilde{x}(\rho_k)$. Note that, since V is fixed, the basis space is common to all the frozen models and thus the interpolation can be done at the state-matrices level (as in IOROM). However, the projection is parameter-dependent due to the use of a parameter-varying test space $W(\rho)$.

3.3. Basis and test spaces construction

In order to preserve the most important features of the input-output mapping of the system when projecting into the lower order subspace of dimension n_z , the matrices V and W are computed from the controllability and observability Gramians, respectively W_c and W_o . This ensures that the projection preserves the most observable and controllable states, enabling an approximate data-driven balanced truncation of the reduced-order LPV model.

Because the approach is fully data-driven, empirical Gramians are computed from data matrices consisting of appropriate state trajectories using known systems theoretical results. The empirical controllability Gramian can be obtained, following the definition [37,43], by impulse response simulations (one for each input channel). As for the empirical observability Gramian, if the model is linear and its adjoint is available, then it can be computed from impulse response simulations (one for each output channel) of the adjoint system, as done in balanced POD [40]. This computation is identical to the one giving the controllability Gramian, but is applied to the adjoint system (for an LTI model in state-space form,

Algorithm 1 Balanced Mode Decomposition with oblique projection.

Input: parameter grid points $\{\rho^j\}_{j=1}^{n_g}$; snapshot matrices $\{X_0(\rho^j), X_1(\rho^j), U_0(\rho^j), Y_0(\rho^j)\}_{j=1}^{n_g}$; empirical Gramians $\{W_c(\rho^j), W_o(\rho^j)\}_{j=1}^{n_g}$; desired model order n_z .
Output: test space projection matrices $\{W(\rho^j)\}_{j=1}^{n_g}$; fixed basis space projection matrix V ; reduced-order models at the grid points $\{F(\rho^j), G(\rho^j), H(\rho^j), D(\rho^j)\}_{j=1}^{n_g}$.

```

1: for  $j = 1, \dots, n_g$  do
2:    $L_c(\rho^j)L_c(\rho^j)^\top = W_c(\rho^j)$    Cholesky factorization of  $W_c$ 
3:    $L_o(\rho^j)L_o(\rho^j)^\top = W_o(\rho^j)$    Cholesky factorization of  $W_o$ 
4:    $H(\rho^j) = L_c(\rho^j)^\top L_o(\rho^j)$ 
5:    $(U, \star, \star) = \text{svd}(H(\rho^j))$ 
6:    $\tilde{U} = U(:, 1:n_z)$ 
7:    $(\tilde{U}, \star, \star) = \text{svd}(L_c(\rho^j)\tilde{U})$ 
8:    $\tilde{Q}(:, 1+n_z(j-1):n_zj) = \tilde{U}(:, 1:n_z)$ 
9: end for
10:  $(U_V, \star, \star) = \text{svd}(\tilde{Q})$ 
11:  $V = U_V(:, 1:n_z)$    Fixed basis space
12: for  $j = 1, \dots, n_g$  do
13:    $(\tilde{Q}, \tilde{R}) = \text{qr}(L_o(\rho^j)^\top V)$    Thin QR factorization
14:    $Q = \tilde{Q}(:, 1:n_z)$ 
15:    $R = \tilde{R}(1:n_z, :)$ 
16:    $W(\rho^j) = L_o(\rho^j)Q(R^\top)^{-1}$    Parameter-varying test space
17:
18: end for

```

$$\begin{bmatrix} F(\rho^j) & G(\rho^j) \\ H(\rho^j) & D(\rho^j) \end{bmatrix} = \begin{bmatrix} W^\top(\rho^j)X_1(\rho^j) \\ Y_0(\rho^j) \end{bmatrix} \begin{bmatrix} W^\top(\rho^j)X_0(\rho^j) \\ U_0(\rho^j) \end{bmatrix}^\dagger$$

solution of the BMD regression problem

this is obtained by replacing A and B by A^\top and C^\top). If the above does not hold, for example in case one has only access to the system's trajectories and not to the model's matrices, the approach developed in [44], valid also for nonlinear systems and not requiring an adjoint model, can be used. In this method the data matrices used for the Gramian computation consist of state trajectories obtained from unforced (zero input) simulations (one for each state) obtained by perturbing the initial condition of each state. Since these are unforced responses, when the system is sufficiently damped, it will be generally sufficient to observe only the initial time-steps and thus this calculation can be parallelized and efficiently implemented to reduce the computational time.

Once W_c and W_o are available, a procedure based on [29, Proposition 2] is employed to compute the test and basis spaces. This construction is reported in the first part of the pseudocode given in Algorithm 1, which summarizes input, output, and main steps of the BMD algorithm (MATLAB notation is used for matrix operations and rows/columns selection). For a fixed value of ρ , the construction proposed in [29] is an equivalent procedure, but more numerically stable for large-scale systems, to the well known square root algorithm for balanced truncation [43]. Indeed, it can be noted (see lines 5–11) that the subspace V is taken as a basis for $\text{span}(L_c\tilde{U})$, where L_c (L_o) is a Cholesky factor of W_c (W_o) and \tilde{U} consists of the first n_z left singular vectors of $H = L_c^\top L_o$. The singular values of H are the Hankel singular values of the system and the SVD decomposition of H plays a fundamental role in balanced reduction [37,43]. As for the subspace W , it can be shown that the expression in line 16 is equivalent to $W = W_o V (V^\top W_o V)^{-1}$, but it is computed with improved numerical robustness [45] by making use of a thin QR factorization (line 13). These choices of V and W are shown [29, Proposition 2] to provide the same balancing projection operator used in the square root algorithm. The desired order n_z is indicated as an input of Algorithm 1, because one might want to obtain a low-order model with a specified size. This is also the approach used in the analyses presented in Section 4. Alternatively, n_z can be indirectly defined as in aDMDC and IOROM by setting a threshold on the singular values of a data matrix. Whereas in those two methods this is done with respect to singular values of the snapshot matrix of the state (X_1 in Eq. (5) and X_0 in Eq. (11)), in the BMD algorithm the matrix H should be considered. That is, a threshold on the Hankel singular values of the system can be selected, as in standard balanced reduction. This threshold can be conveniently expressed as percentage of the

largest Hankel singular value. Note that for each parameter in the grid there is a different matrix H (line 4), and thus, given a threshold, the number of truncated singular values might differ at each grid point. Because obviously a common value of n_z for all the local models should be selected, a remedy for this is e.g. to choose as n_z the maximum number of truncated singular values across the grid.

The output $\{F(\rho^j), G(\rho^j), H(\rho^j), D(\rho^j)\}_{j=1}^{n_g}$ provided by the BMD algorithm is a grid LPV model. After an interpolation algorithm to evaluate the matrices' entries for any value of ρ inside the considered range has been chosen, this model can be used for simulation and control design. Note also that recently proposed robust analysis methods for linear-time varying (LTV) systems [46,47] can be applied to this model, e.g. to investigate specific aircraft manoeuvres. Indeed, by fixing a particular trajectory for ρ the LPV system is transformed into an LTV one. Moreover, the parameter-varying test space $W(\rho^j)$ can be useful to gain insights into the aeroservoelastic modes which have been eliminated and those that have been kept in the projection, while the parameter-independent basis space can be used to recover at each time-step k the high-dimensional state via the transformation $\tilde{x}_k = V\tilde{z}_k$.

As noted in the introduction, the algorithm provides an approximate balanced truncation. Approximation is related to the use of empirical Gramians, which are only finite-time approximations of the true ones (for this reason, also called finite-time Gramians) since their computation is trajectory-based. As a result, they only provide in principle a finite-time balanced realization [14], whereas the theoretical order reduction error bounds are only available for infinite-time balanced realizations. This source of error can however be made arbitrarily small by using long enough data sequences for constructing the Gramians. The slowest decay rate of the system's impulse responses is the key parameter to consider when choosing the length of the trajectory [48].

3.4. Extension to handle algebraic constraints

Since the BMD algorithm will be applied in Section 4 to a wing described by an FSI solver which implements the algebraic constraints described in Section 2.2, an extension to handle this instance is presented here. For a fixed value of ρ , the model structure for the high-order model becomes

$$\begin{aligned} \tilde{x}_{k+1} &= A\tilde{x}_k + B\tilde{u}_k + R\tilde{u}_{k+1}, \\ \tilde{y}_k &= C\tilde{x}_k + D\tilde{u}_k + P\tilde{u}_{k+1}, \end{aligned}$$

where a potential effect of the algebraic constraints in the output equation is also considered via the matrix P (in the analyses of the morphing wing this matrix was, as expected, always found to be zero). Therefore, the low-order approximation (for each value of ρ in the parameter grid) becomes

$$\begin{aligned} \begin{bmatrix} A & B & R \\ C & D & P \end{bmatrix} &\approx \begin{bmatrix} VFW^T & VG & VL \\ HW^T & D & P \end{bmatrix} \\ &= \begin{bmatrix} V & 0 \\ 0 & I_{n_y} \end{bmatrix} \begin{bmatrix} F & G & L \\ H & D & P \end{bmatrix} \begin{bmatrix} W^T & 0 & 0 \\ 0 & I_{n_u} & 0 \\ 0 & 0 & I_{n_y} \end{bmatrix}. \end{aligned}$$

The new objective function to be minimized is

$$\min_{F,G,L,H,D,P} \left\| \begin{bmatrix} X_1 \\ Y_0 \end{bmatrix} - \begin{bmatrix} W & 0 \\ 0 & I_{n_y} \end{bmatrix} \begin{bmatrix} F & G & L \\ H & D & P \end{bmatrix} \begin{bmatrix} W^T & 0 & 0 \\ 0 & I_{n_u} & 0 \\ 0 & 0 & I_{n_y} \end{bmatrix} \begin{bmatrix} X_0 \\ U_0 \\ U_1 \end{bmatrix} \right\|_F^2,$$

and the new optimal solution is

$$\begin{bmatrix} F & G & L \\ H & D & P \end{bmatrix} = \begin{bmatrix} W^T X_1 \\ Y_0 \end{bmatrix} \begin{bmatrix} W^T X_0 \\ U_0 \\ U_1 \end{bmatrix}^\dagger.$$

The BMD LPV reduced-order model with algebraic constraints is then, in analogy to (19), given by

$$\begin{aligned} \tilde{z}_{k+1} &= F_{\rho_k} \tilde{z}_k + G_{\rho_k} \tilde{u}_k + L_{\rho_k} \tilde{u}_{k+1} + (\tilde{z}(\rho_k) - \tilde{z}(\rho_{k+1})), \\ \tilde{y}_k &= H_{\rho_k} \tilde{z}_k + D_{\rho_k} \tilde{u}_k + P_{\rho_k} \tilde{u}_{k+1}. \end{aligned} \quad (20)$$

The IOROM algorithm has also been extended to the algebraic-differential case in order to allow its application to the test case considered in Section 4. This can be done in a similar way to what has been shown above for BMD. Specifically, starting from (12), the new least-squares problem for the IOROM reduced-order model is

$$\min_{F,G,L,H,D,P} \left\| \begin{bmatrix} X_1 \\ Y_0 \end{bmatrix} - \begin{bmatrix} Q & 0 \\ 0 & I_{n_y} \end{bmatrix} \begin{bmatrix} F & G & L \\ H & D & P \end{bmatrix} \begin{bmatrix} Q^T & 0 & 0 \\ 0 & I_{n_u} & 0 \\ 0 & 0 & I_{n_y} \end{bmatrix} \begin{bmatrix} X_0 \\ U_0 \\ U_1 \end{bmatrix} \right\|_F^2,$$

which has the following solution

$$\begin{bmatrix} F & G & L \\ H & D & P \end{bmatrix} = \begin{bmatrix} Q^T X_1 \\ Y_0 \end{bmatrix} \begin{bmatrix} Q^T X_0 \\ U_0 \\ U_1 \end{bmatrix}^\dagger. \quad (21)$$

Eq. (20), when the interpolated matrices are taken from (21), provides the IOROM LPV reduced-order model with algebraic constraints.

4. Results

This section presents and discusses results obtained by applying the three algorithms aDMDc, IOROM, and BMD to the flexible and highly cambered morphing wing depicted in Fig. 1. The wing is made of composite material, and the trailing edges are able to morph and, by doing so, to increase or decrease the camber, thus replacing conventional ailerons. The reader is referred to the related previous works for details on the wing design [49] and its investigation with fluid-structure interaction tools [34].

4.1. Summary of the wing's FSI model

The high-fidelity FSI model of the morphing wing is presented in detail in [33,50]. The structural model is based on a combination of linear plate and beam elements. The external skin of the lifting surface and the Voronoi-based internal structure are modelled using plates, while the stringers are modelled with beam elements. The stiffness and mass matrices are obtained with the commercial software Nastran [51]. From these, a modal decomposition is performed to extract the structural modes of the wing, which are coupled via thin plate spline and inverse distance weighting [52] with the aerodynamic model. The aerodynamic model is based on a 3D unsteady panel method [53]. The unsteadiness of the flow is represented by shedding at each time step a new row of vortex ring singularities after the trailing edge. All the other wake nodes are then moved, via a second-order Runge–Kutta integration scheme, using the local velocity of the flow. The aerodynamic forces on the surface are computed with the coefficient of pressure on each panel, considering the far field velocity, the induced velocity by the wing itself, and the induced velocity by all the wake panels.

The state of the system x consists of the total number of structural modes of the wing and the doublet strengths (from the 3D panel method solver), with $n_x=618$. The input vector u of size $n_u=6$ is given by

$$u = [\alpha; p; q; r; F_s; F_{as}], \quad (22)$$

where α is the angle of attack, p , q , and r are the roll, pitch, and yaw rotation rates, and F_s and F_{as} are the (normalized) symmetric and anti-symmetric morphing actuation inputs. Their value is associated with a camber deformation and is thus related to a trailing edge deflection: specifically, the amount of upwards (negative value) or downwards (positive value) deflection.

As for the output channels, we will consider both single output and multiple output models. Emphasis will be given to the case where the output is the first bending mode of the wing ($n_y=1$), since this is usually the one associated with dynamic instabilities and large deformations, and thus it is of particular interest for active control tasks [54,55]. The flight speed will be considered as the time-varying parameter ($n_\rho=1$).

The training phase, common to both the algorithms and consisting of generating the snapshot matrices in Eq. (3), is carried out by exciting the system with a series of impulses deployed in random order in all input channels, following the same procedure adopted in [30]. The amplitude of these signals has been chosen so that nonlinear effects due to the wake's evolution are not excited. The same precaution is used for the computation of the empirical gramians. Trajectories are of length $n_s=500$ and are recorded with sampling time 0.006 s.

4.2. Fixed-parameter models

In this first set of tests, the accuracy of the different models at fixed values of the flight speed V is assessed by means of sinusoidal inputs. The frequencies, different for each channel, are expressed in terms of the aircraft reduced frequency $f_r =: \frac{V}{\bar{c}}$, where $\bar{c}=0.29$ m is the mean chord of the wing. The first, third, and fifth input channels are excited with sinusoids $\frac{1}{5\pi} f_r$, $\frac{1}{10\pi} f_r$, and $\frac{1}{20\pi} f_r$, respectively, while the other channels are set to zero. For reference, the first bending mode of the wing has a frequency of approximately 10 Hz [33]. This test is performed for 3 flight speeds in the range of operating conditions of interest, namely $V=30$ m/s, $V=40$ m/s, and $V=50$ m/s. To quantify the accuracy as a function of the order of the model n_z , the Euclidean norm of the error signal

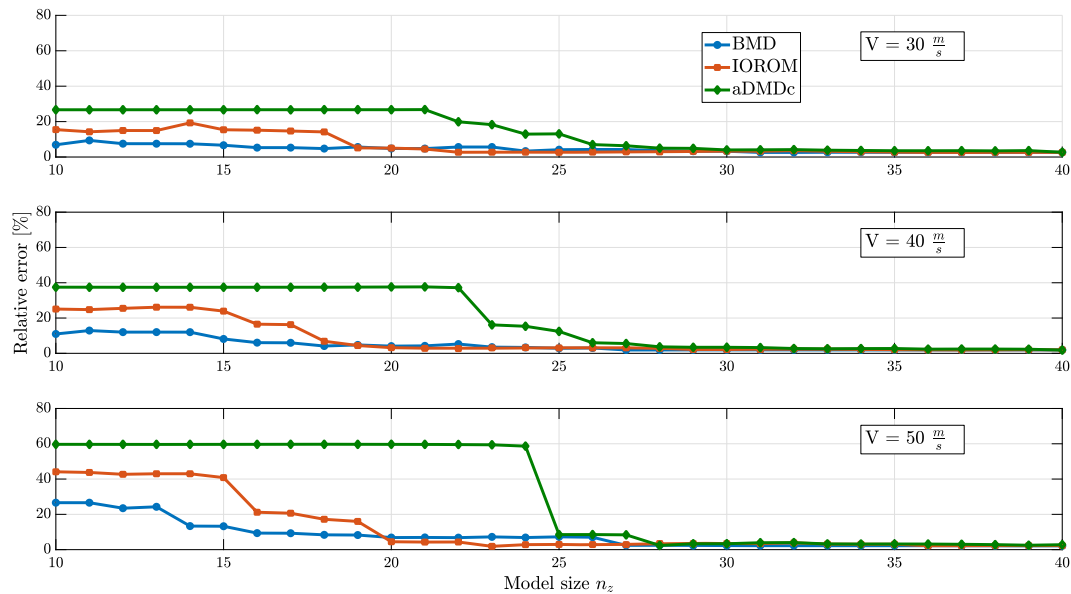


Fig. 2. Relative error on the prediction of the system's output (bending mode) for three different values of the flight speed as a function of the model order.

between the first bending mode amplitude provided by the high-fidelity FSI and each of the predictions obtained with the three ROM algorithm is computed. The results in terms of relative error (with respect to the predicated signal norm) are shown in Fig. 2.

It is clear that for all speeds BMD provides the smallest error for a very low-order approximation of the full dynamics, as expected in view of the choice of low-dimensional subspace where the high-dimensional data are projected. As the size n_z of the system increases, the difference between the algorithms is less noticeable and, for high enough orders, the algorithms tend to give same results.

It is also noted that the results obtained with the aDMDC algorithm showed great sensitivity, in the range of n_z displayed in Fig. 2, to the SVD truncation order r employed in Eq. (4). Using the hard threshold criterion from [36] did not provide good results as it resulted in a very large r (therefore the truncation included very low singular values deteriorating the approximation). Since fine-tuning the value of r to optimize the results obtained with aDMDC would have required trying several values of r for each different value of n_z , this was not pursued here. Instead, the heuristic choice $r=n_z+10$ was implemented and proved to provide reasonable results. Even though, because of this possibly suboptimal choice, the resulting gap in performance with the other two methods observed in Fig. 2 can be also ascribed to numerical inaccuracies associated with the decomposition (4), the need to optimally choose r can be considered as a disadvantage of the aDMDC problem formulation. It is finally observed that, in the extensive analyses performed, aDMDC typically showed worse performance than BMD irrespective of the choice of r .

4.3. Parameter-varying models

In the second set of tests, the accuracy during parameter-varying manoeuvres is tested. A manoeuvre of 3 s where the flight speed linearly increases from $V=20$ m/s to $V=50$ m/s is analyzed. Unless otherwise specified, the reduced-order models are obtained using snapshot matrices obtained gridding the flight speed range every 2 m/s and thus using 16 different speeds ($n_g=16$). A linear interpolation will be used to evaluate quantities for values of ρ that are not in the grid.

4.3.1. Sinusoidal excitation

The same sinusoidal input signals used in Section 4.2 are considered here. In Fig. 3, the bending mode amplitude response obtained with the FSI solver (FSI) is compared with the predictions of the three algorithms when the order of the models is fixed at $n_z=14$. The goal is to compare the prediction accuracy of the algorithms when a very low number of states (in comparison with the order of the original system) is employed. It is noted that the same observations can be gathered when values of n_z in the same range are considered. All the signals are normalized by the largest value of the bending amplitude measured in the FSI simulation, which for the analyzed case was 0.7. Note that a unitary value of the first bending mode corresponds to a wingtip displacement of 4.6 cm. In this case, since only one value of n_z was considered, the aDMDC model was here obtained by fine tuning the threshold value r in order to provide the best results.

The plot confirms, also in the LPV setting, that the BMD algorithm guarantees the smallest error when a low-rank approximation of the system is desired. In this simulation, aDMDC outperforms IOROM, possibly due to the fact that it uses a parameter-varying set of POD modes. However, aDMDC does not provide a family of interpolated low-order models, and interpolates directly the high-order states, thus requiring parallel simulations of the low-order models. The better performance of BMD, despite the fact that a part of the projection (the one related to the basis space) is constant, is ascribed to the improved selection of subspace for the projection compared to the standard POD one, common to the other two methods. In addition to the improvement in the accuracy, the BMD algorithm is also capable of providing, like the IOROM algorithm, a family of consistent LTI models with the advantages for LPV control design and in general real-time applications.

4.3.2. Effect of the input signals

This section investigates the accuracy of the reduced-order models for different types of input signals. The Euclidean norm of the error signal between the first bending mode amplitude provided by the high-fidelity FSI and the prediction obtained with each of the three ROM algorithms is again used as metric to assess the quality of the approximation. Three classes of inputs are considered: *Sine* coincides with the signal tested so far and already

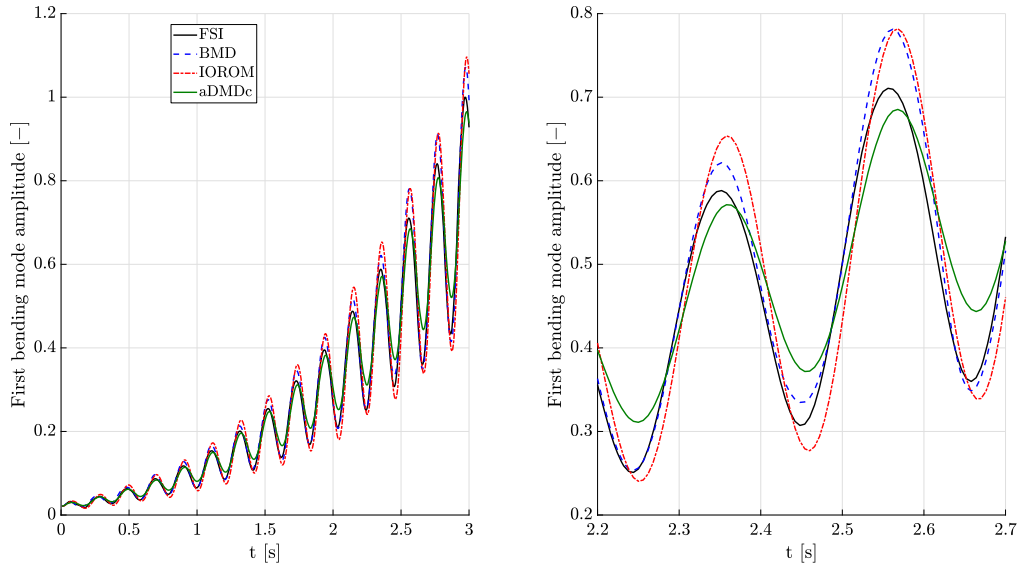


Fig. 3. Comparison of the normalized output (bending mode) for parameter-varying simulations (flight speed linearly increasing from 20 m/s to 50 m/s) of models with size 14.

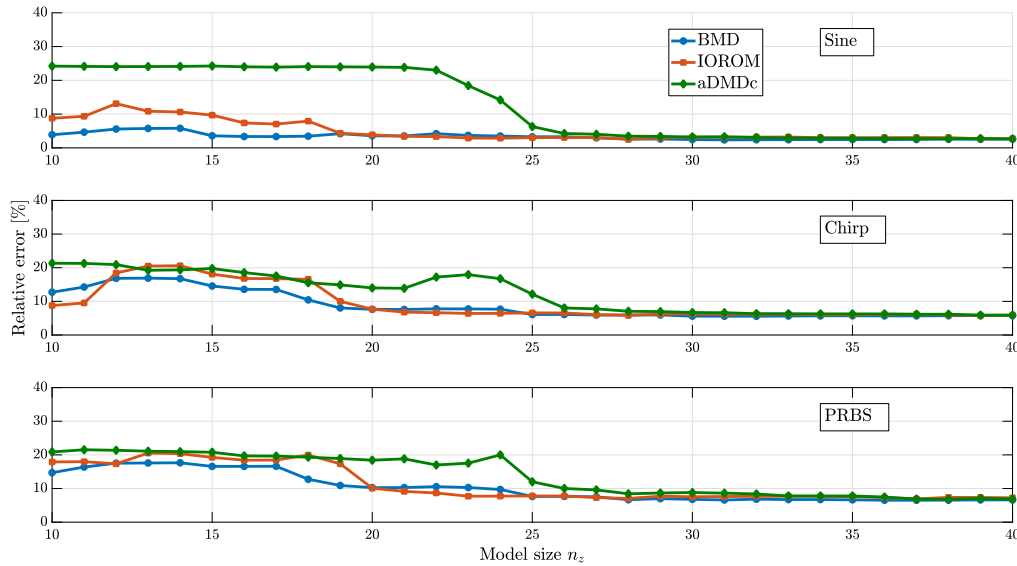


Fig. 4. Relative error of the prediction of the system's output (bending mode) for speed-varying manoeuvres with three different types of input signals as a function of the model order.

investigated in [30]; *Chirp* excites the system by injecting in all 6 input channels defined in (22) a chirp signal with frequency linearly varying from $\frac{1}{50\pi} f_r$ to $\frac{2}{15\pi} f_r$; *PRBS* excites the system by injecting in all 6 input channels a PRBS-9 sequence. This last input, namely a Pseudo-Random Binary Signal (PRBS) [56], is a deterministic signal with white-noise-like properties. It is very well known in the system identification and experiment design fields since it has the favourable property of equally distributing energy across all the frequencies in the input spectrum. In this way, information on the models in different frequency ranges can be extracted. Although not a common input in aeroservoelastic applications, it has been used in this spirit here, since the previously adopted sets of input will only give information on the behaviour of the reduced models around the aircraft reduced frequency f_r . Results are shown in Fig. 4.

The plots confirm the advantage in using the BMD approach when seeking a low-order model capturing parameter variations.

These results are important considering that they are obtained by exploring different frequency ranges of the system's response.

An interesting aspect observed in Fig. 4 is that none of the algorithms exhibit a monotonic improvement of the model's accuracy (measured here by the relative prediction's error) as a function of the system's order. Indeed, in a very few cases, a small deterioration can be observed between two consecutive values of n_z , before the curve keeps decreasing as n_z increases. Even though this might seem surprising at first glance, there is no theoretical guarantee that such a monotonic improvement is achieved in this type of algorithms. The reason is that adding one mode to the low dimensional subspace where the dynamics is described might (in principle) deteriorate the model's approximation, if that mode alone does not add meaningful information. This is for example the case when pairs of modes describe relevant features of the system (e.g. modes of vibration), and so only when both of them are used in the projection there is an improvement. In practice, nu-

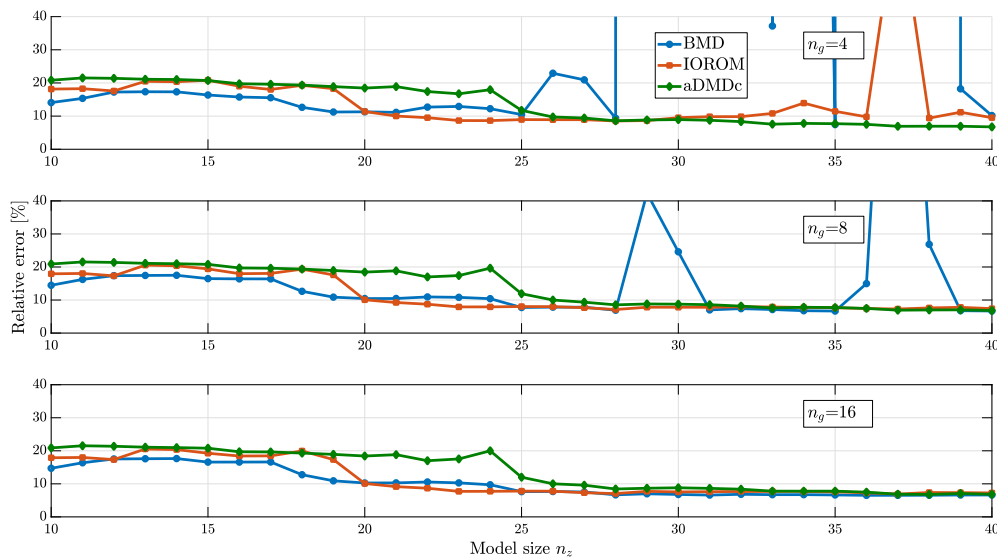


Fig. 5. Relative error on the prediction of the system's output (bending mode) for speed-varying manoeuvres with values of n_g as a function of the model order.

merical errors due to the SVD truncation and the interpolation of the models across the parameter's grid can also have an effect on these results. However, it is noted that the number of occurrences and the entity of this deterioration are limited.

4.3.3. Effect of the parameter grid

In this section we analyze the effect of the flight speed grid where the reduced-order LTI models are computed, i.e. the selection of the parameter n_g . This is an important aspect, which is known to influence both the accuracy of the LPV models and the quality of the control design based on them. Three cases are compared in Fig. 5: $n_g=4$ where the grid includes one plant every 10 m/s; $n_g=8$ where the grid includes one plant every 4 m/s from $V=20$ m/s to $V=48$ m/s and then $V=50$ m/s; $n_g=16$ which is the grid used so far. PRBS inputs are used to excite the system in these tests.

From the analyses it can be gathered that aDMDC is more robust than the other algorithms to the value of n_g . In particular, both IOROM and BMD present poor performance for a few reduced order models in the range of n_z between 30 and 40 when the flight speed grid is coarser. The reason for this behaviour is due to the interpolation approaches employed by the three ROM schemes. Whereas IOROM and BMD interpolate the low-order state-space matrices obtained at the grid points, aDMDC interpolates directly the high-order vector states which are obtained by lifting the low-order states \tilde{z} from the local models (8) running in parallel. Interpolating every entry of the state-space matrices therefore makes the choice of the grid a more delicate aspect in IOROM and BMD. The unstable behaviours resulting in very high (out of the plot) errors are indeed ascribed to numerical inaccuracies in this interpolation. It has been observed that the entries of the matrices are overall bigger as the order n_z is increased, hence justifying why these outliers take place within the aforementioned range of model's orders. While there does not seem to be a fundamental reason to explain it, it is apparent that the sensitivity of BMD to the coarseness of the parameter grid is more accentuated. A possible explanation is that, because at each grid point the computation of the empirical Gramians is required, dealing with coarse grid exacerbates the numerical inaccuracies associated with the projections on low-order models. Improved interpolation schemes, not considered in this work, could be employed to ameliorate this issue. Except for these isolated numerical problems, BMD shows better performance even when very coarse grids are employed.

4.3.4. Prediction capability for other signals

The capability of the models to predict other quantities of interest, such as for example aerodynamic coefficients depending on the system's states, is investigated. In particular, we test the accuracy when these coefficients are added to the vector of output (this is done for BMD) or computed directly from the states (this is done for IOROM and aDMDC). In the latter case, the low-order states are lifted to the high-order ones, which are then used to compute the coefficients using their known relationship to the states. While this is the only possible way of reconstructing the system's signals for aDMDC, in IOROM and BMD this can alternatively be done by simply adding the desired quantities to the vector of outputs. This would probably be the preferred approach if the signals are used for control (either because they represent measurements fed to the controller or because they are performance measures to be optimized). The different choice done here for BMD and IOROM is for the sake of exploring different models, and results showed that whether the signals were computed from output channels or retrieved from the states had a very minor impact on the predictive accuracy.

Fig. 6 shows the normalized lift (C_L), pitch (C_M), and drag (C_D) coefficients for the same constant acceleration manoeuvre considered in the previous sections and with a sinusoidal excitation. Normalization is performed, as done earlier in Fig. 3, by dividing each signal by the largest value of the corresponding signal in the FSI simulation.

The same observations gathered earlier with respect to the trajectory of the bending mode are confirmed here. It is particularly interesting to observe that, even though these coefficients are not outputs of the model, and thus the balancing projection is not aimed directly at capturing them, the BMD algorithm is still able to perform better than the others. Fig. 7 shows the same analyses when chirp signals are used as input to excite the model. Similar conclusions can be drawn.

4.4. Reduced-order models for model predictive control

In this section, a control application of the morphing wing's low-order models is investigated. Specifically, model predictive control (MPC) [57] is considered, given its well established use in the AWE field [58]. Two distinct reference tracking problems are examined, where pre-defined lift and first bending mode amplitude profiles are tracked while flying trajectories over a range

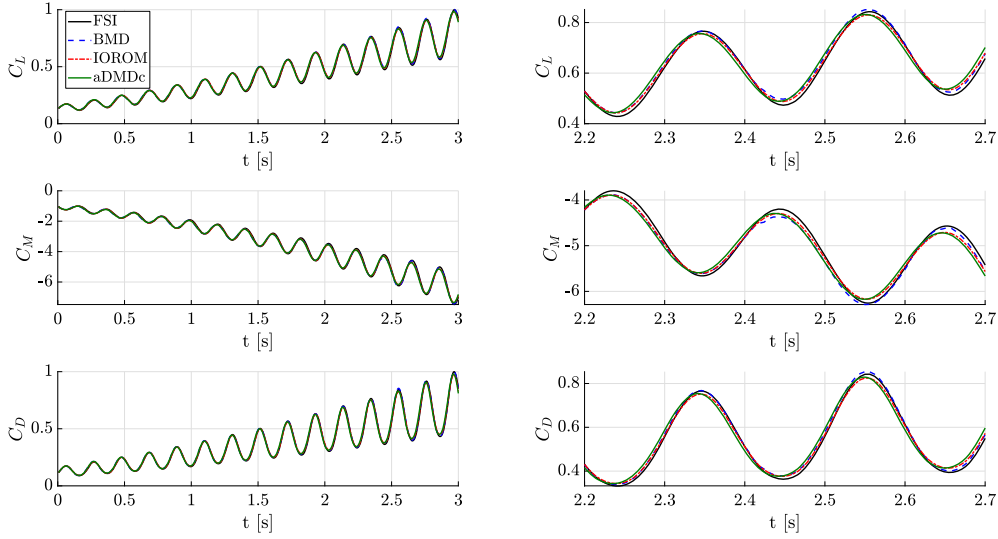


Fig. 6. Comparison of the normalized lift, pitch, and drag coefficients for a parameter-varying simulation with $n_z=14$ and sinusoidal inputs.

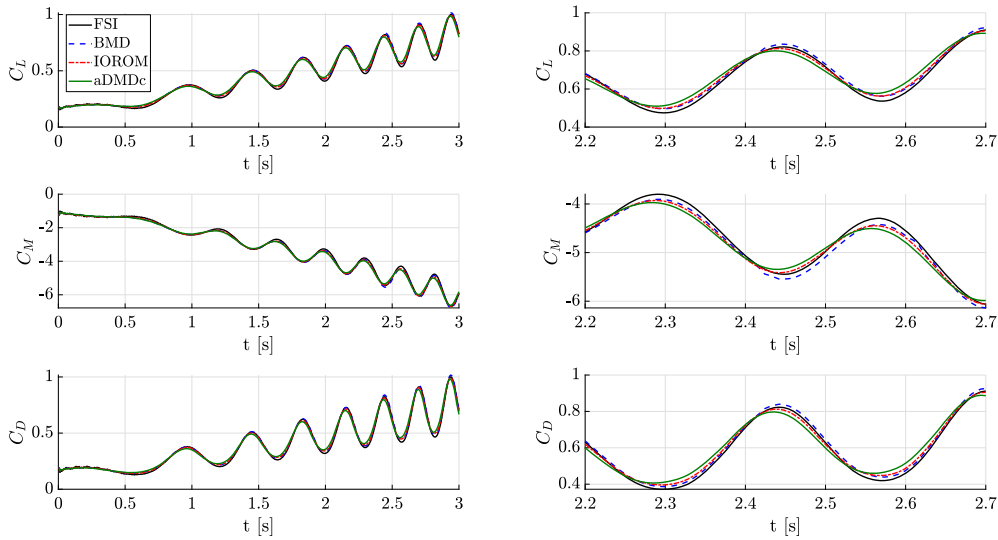


Fig. 7. Comparison of the normalized lift, pitch, and drag coefficients for a parameter-varying simulation with $n_z=14$ and chirp inputs.

of different flight speeds and in the presence of turbulence and gusts. The analysis of these manoeuvres is motivated by the interest in using active control to guarantee a safe operation for the AWE system (with respect to some of its critical components such as the wing or tether) by keeping indicators of the structural integrity close to desired, and possibly pre-optimized, values. This can avoid passive remedies such as reducing the load transmitted to the ground station, which in turn decreases the amount of wind energy harvested. Having effective and reliable control laws to guarantee the integrity of the AWE system can represent an important enabler for this technology [59].

In its basic form, model predictive control repeatedly solves a finite-horizon optimal control problem of length N_c subject to input and state-constraints. At each instant, a model of the system is employed to predict its response and thus select the control sequence $(u_i)_{i=0}^{N_c-1}$ which minimizes the cost

$$J_{MPC} = \sum_{k=0}^{N_c-1} \left(\|\tilde{y}_k - r_k\|_N^2 + \|\tilde{u}_k\|_M^2 + \|\Delta\tilde{u}_k\|_{M_\Delta}^2 \right), \quad (23)$$

where r is the reference trajectory, and for a vector x , we denote by $\|x\|_p$ the weighted l_2 -norm $(x^T P x)^{\frac{1}{2}}$. Besides the terms penalizing deviation of the output \tilde{y} from r (with the weighting matrix $N \in \mathbb{R}^{n_y \times n_y}$) and control effort (with the weighting matrix $M \in \mathbb{R}^{n_u \times n_u}$), the cost in (23) also penalizes fast changes in the input via the term $\Delta\tilde{u}_k = \tilde{u}_k - \tilde{u}_{k-1}$ (e.g. to avoid actuator rate saturation).

The following optimization problem will be solved to obtain the optimal input sequence

$$\text{minimize}_{(u_i)_{i=0}^{N_c-1}, (y_i)_{i=0}^{N_c-1}} J_{MPC}(\tilde{u}, \tilde{y}), \quad (24a)$$

$$\text{subject to } \tilde{y}_i = f(\tilde{u}_i, \tilde{u}_{i+1}), \quad (24b)$$

$$(\tilde{u}_i)_{i=0}^{N-1} \in \mathcal{U}, \quad (24c)$$

where: the cost function (24a) is defined in (23); the constraint set \mathcal{U} (24c) enforces minimum and maximum values for the input; and (24b) enforces the dynamic constraint that relates the sequence of input to the output via an input-output model of the system f . Precisely, f will be formulated here by using the reduced-order aDMDC, IOROM, and BMD models. The goal of the

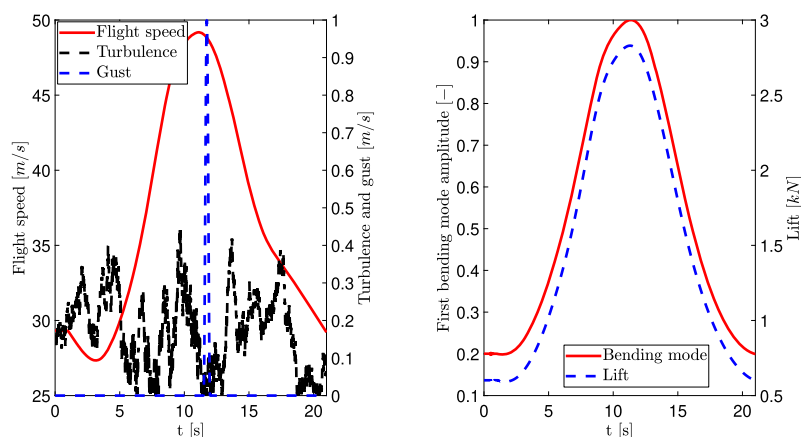


Fig. 8. Speeds variations during the manoeuvres (left plot) and reference tracking profiles (right plot).

analyses is to compare the associated closed-loop cost J_{MPC}^{CL} , that is the cost (23) incurred by the true system (simulated here by the high-dimensional FSI solver) when this is regulated by the inputs optimized solving problem (24). Since the model is used to predict the system's output, and thus select the input sequence, any mismatch between model and system can result in a degradation of the controller performance.

The analysis considers the case where the morphing wing (Fig. 1, I) flies a trajectory with flight velocities ranging between $V=27$ m/s and $V=50$ m/s (Fig. 8-left). Additionally, the wing is subject to a gust at the maximum flight speed with gust length 0.5 s corresponding to a 1 degree deflection of the angle of attack α , and to turbulence generated with a Dryden filter (Fig. 8-left). The right plot in Fig. 8 depicts the lift and the first bending mode's amplitude profiles tracked by the MPC algorithm. Recall from the previous discussion that a unitary value of the first bending mode corresponds to a wingtip displacement of 4.6 cm.

The scenario considered here has only a symmetric morphing actuation input, i.e. $\tilde{u} = F_s$. This normalized input is constrained to be in the interval $[-3, 3]$ at each time-step. This is associated with allowable deflections of the trailing edge in the range ± 9 mm. Problem (24) is solved using the MATLAB implementation provided in [60], where the application of MPC with models obtained via DMDc was investigated.

The output \tilde{y} is either the first bending mode, or the lift force generated by the wing, depending on the case. The control horizon is $N_c=10$ and the weights used in the cost are: $N=1300$, $M=10$, and $M_\Delta=0.1$ (lift tracking) and $N=13000$, $M=10$, and $M_\Delta=0.1$ (bending tracking). The penalty on the output deviation is increased in the latter case due to the difference in magnitude of the two tracked quantities (recall Fig. 8-right). Fig. 9 shows the comparison of the closed-loop cost J_{MPC}^{CL} resulting from closing the true plant (simulated by means of the high-dimensional FSI solver) with the MPC controller generated using the low-order models. For the sake of clarity, the closed-loop cost J_{MPC}^{CL} has been normalized in each case by dividing it by the corresponding value obtained with the BMD algorithm when $n_z=40$.

The observations gathered in the previous sections regarding the better prediction performance achieved with the BMD algorithm when low-order approximations are considered are confirmed here in the context of control applications. Both plots show that, while for higher orders the closed-loop costs have very similar values, when the size of the model is decreased the BMD gives in general the lowest cost. Another interesting observation is that the lift tracking problem is quite robust to the use of low-order models. Indeed, the closed-loop costs are always within two times of the lowest cost achieved at $n_z=40$ except for the case of

aDMDc at $n_z=10$). On the other hand, the bending tracking problem is shown to be more challenging when low-order representations are employed. Whereas no attempt to further optimize the MPC problem tuning was made (all the design parameters were kept the same independent of n_z), this motivates further work on the use of low-order models for control of coupled flexible structures like those encountered in AWE applications. In the real-time control setting, it is important to stress that by using BMD and IOROM models an order of magnitude computational speed-up was achieved with respect to the cases where aDMDc models were used. This is because the aDMDc models have the requirement of running several models in parallel.

5. Conclusion

The paper proposes the Balanced Mode Decomposition with oblique projection algorithm, a novel data-driven algorithm for constructing low-order LPV models from system's trajectories. Two recent algorithms from the literature, aDMDc and IOROM, are considered for comparison since they both have connections with the newly proposed approach. Technical details on the BMD algorithm are given in order to clearly point out the innovations, and the advantages with respect to previous work. The performance of the BMD algorithm is assessed on a morphing wing for airborne wind energy applications. The results, proposed both for the fixed parameter and, more extensively, for the parameter-varying case, confirm the theoretical advantages discussed in the technical part of the paper. When seeking low-order model representations, the BMD approach achieves generally, among the tested algorithms, the lowest prediction error and best control performance when used as model for an on-line MPC scheme. The improved accuracy is ascribed to the use of a projecting subspace that balances the low-order states (this element is of interest also in a fixed-parameter setting), and to the use of a parameter-varying projection operator (which can thus be enriched with parameter-dependent features, instead of being fixed throughout the range). This has the advantageous feature of being achieved while guaranteeing state-consistency. Owing to these appealing features, it is envisaged the application of BMD for tasks such as off-line and real-time control design, and in multi-disciplinary optimization tool chains, where typically low-order representations are employed as surrogate models.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

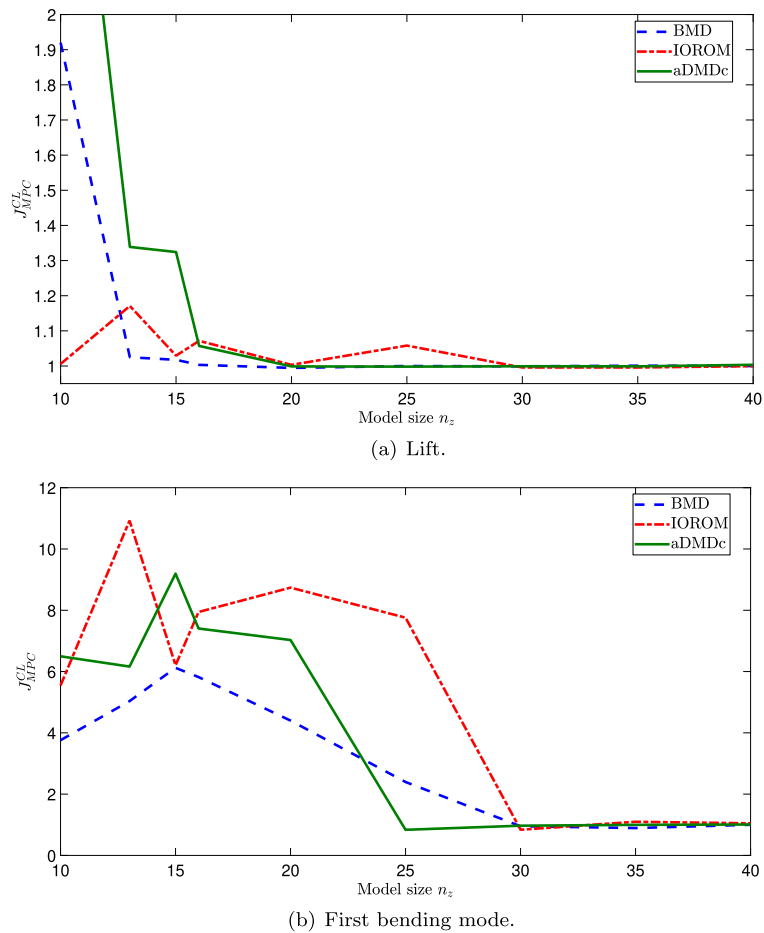


Fig. 9. Normalized closed loop costs for the two MPC tracking problems as a function of the model order.

Acknowledgements

This work is supported by the Swiss National Science Foundation under grant no. 200021_178890. The authors wish to thank Nivethan Yogarajah and Eva Ahbe for helpful discussions.

References

- [1] S.L. Brunton, B.R. Noack, P. Koumoutsakos, Machine learning for fluid mechanics, *Annu. Rev. Fluid Mech.* 1 (2) (2020) 391–421.
- [2] P. Holmes, J.L. Lumley, G. Berkooz, Turbulence, Coherent Structures, Dynamical Systems and Symmetry, Cambridge Monographs on Mechanics, Cambridge University Press, 1996.
- [3] P. Schmid, Dynamic mode decomposition of numerical and experimental data, *J. Fluid Mech.* 656 (2010) 5–28.
- [4] J.H. Tu, C.W. Rowley, D.N. Luchtenburg, S.L. Brunton, J.N. Kutz, On dynamic mode decomposition: theory and applications, *J. Comput. Dyn.* 52 (1) (2014) 477–508.
- [5] G. Berkooz, P. Holmes, J.L. Lumley, The proper orthogonal decomposition in the analysis of turbulent flows, *Annu. Rev. Fluid Mech.* 25 (1) (1993) 539–575.
- [6] P. Schmid, D. Henningson, Stability and Transition in Shear Flows, Springer-Verlag, New York, 2001.
- [7] A. Iannelli, R.S. Smith, The role of the state in model reduction with subspace and POD-based data-driven methods, in: IEEE American Control Conference, 2021.
- [8] M. Morzyński, W. Stankiewicz, B.R. Noack, R. King, F. Thiele, G. Tadmor, Continuous mode interpolation for control-oriented models of fluid flow, in: Active Flow Control, Springer Berlin Heidelberg, 2007.
- [9] B.R. Noack, W. Stankiewicz, M. Morzyński, P.J. Schmid, Recursive dynamic mode decomposition of transient and post-transient wake flows, *J. Fluid Mech.* 809 (2016) 843–872.
- [10] S. Le Clainche, D. Rodríguez, V. Theofilis, J. Soria, Flow around a hemisphere-cylinder at high angle of attack and low Reynolds number. Part II: POD and DMD applied to reduced domains, *Aerosp. Sci. Technol.* 44 (2015) 88–100.
- [11] C. Hu, C. Yang, W. Yi, K. Hadzic, L. Xie, R. Zou, M. Zhou, Numerical investigation of centrifugal compressor stall with compressed dynamic mode decomposition, *Aerosp. Sci. Technol.* 106 (2020).
- [12] R. Huang, H. Li, H. Hu, Y. Zhao, Open/closed-loop aeroservoelastic predictions via nonlinear, reduced-order aerodynamic models, *AIAA J.* 53 (7) (2015) 1812–1824.
- [13] R. Huang, H. Liu, Z. Yang, Y. Zhao, H. Hu, Nonlinear reduced-order models for transonic aeroelastic and aeroservoelastic problems, *AIAA J.* 56 (9) (2018) 3718–3731.
- [14] A.C. Antoulas, Approximation of Large-Scale Dynamical Systems, Society for Industrial and Applied Mathematics, 2005.
- [15] C.P. Moreno, P.J. Seiler, G.J. Balas, Model reduction for aeroservoelastic systems, *J. Aircr.* 51 (1) (2014) 280–290.
- [16] J.L. Proctor, S.L. Brunton, J.N. Kutz, Dynamic mode decomposition with control, *SIAM J. Appl. Dyn. Syst.* 15 (1) (2016) 142–161.
- [17] R. Toth, Modeling and Identification of Linear Parameter-Varying Systems, Springer, 2010.
- [18] A. Packard, Gain scheduling via linear fractional transformations, *Syst. Control Lett.* 22 (2) (1994) 79–92.
- [19] S. Bannani, D.M.C. Willemsen, C.W. Scherer, Robust control of linear parametrically varying systems with bounded rates, *J. Guid. Control Dyn.* 21 (6) (1998) 916–922.
- [20] T. He, A.K. Al-Jiboory, G.G. Zhu, S.S.-M. Swei, W. Su, Application of ICC LPV control to a blended-wing-body airplane with guaranteed H_∞ performance, *Aerosp. Sci. Technol.* 81 (2018) 88–98.
- [21] A. Hjartarson, P. Seiler, A. Packard, LPVTools: a toolbox for modeling, analysis, and synthesis of parameter varying control systems, in: 1st IFAC Workshop on Linear Parameter Varying Systems LPVS, 2015.
- [22] P. Benner, S. Gugercin, K. Willcox, A survey of projection-based model reduction methods for parametric dynamical systems, *SIAM Rev.* 57 (4) (2015) 483–531.
- [23] C. Pousot-Vassal, C. Roos, Generation of a reduced-order LPV/LFT model from a set of large-scale MIMO LTI flexible aircraft models, *Control Eng. Pract.* 20 (9) (2012) 919–930.
- [24] G. Ferreres, Computation of a flexible aircraft LPV/LFT model using interpolation, *IEEE Trans. Control Syst. Technol.* 19 (1) (2011) 132–139.

- [25] A.K. Al-Jiboory, G. Zhu, S.S.-M. Swei, W. Su, N.T. Nguyen, LPV modeling of a flexible wing aircraft using modal alignment and adaptive gridding methods, *Aerosp. Sci. Technol.* 66 (2017) 92–102.
- [26] Q. Zhang, L. Ljung, R.M. Pintelon, On local LTI model coherence for LPV interpolation, *IEEE Trans. Autom. Control* (2019).
- [27] C.D. Villemagne, R.E. Skelton, Model reductions using a projection formulation, *Int. J. Control* 46 (6) (1987) 2141–2169.
- [28] P. van Overschee, L. de Moor, *Subspace Identification for Linear Systems: Theory, Implementation, Applications*, Kluwer Academic Publishers, 1996.
- [29] J. Theis, P. Seiler, H. Werner, LPV model order reduction by parameter-varying oblique projection, *IEEE Trans. Control Syst. Technol.* 26 (3) (2018) 773–784.
- [30] N. Fonzi, S.L. Brunton, U. Fasel, Data-driven nonlinear aeroelastic models of morphing wings for control, *Proc. R. Soc. A, Math. Phys. Eng. Sci.* 2020 (2020).
- [31] J. Murua, R. Palacios, J.M.R. Graham, Applications of the unsteady vortex-lattice method in aircraft aeroelasticity and flight dynamics, *Prog. Aerosp. Sci.* 55 (2012) 46–72.
- [32] J. Annoni, P. Seiler, A method to construct reduced-order parameter-varying models, *Int. J. Robust Nonlinear Control* 27 (4) (2017) 582–597.
- [33] U. Fasel, P. Tiso, D. Keidel, G. Molinari, P. Ermanni, Reduced-order dynamic model of a morphing airborne wind energy aircraft, *AIAA J.* 57 (8) (2019) 3586–3598.
- [34] U. Fasel, D. Keidel, G. Molinari, P. Ermanni, Aeroservoelastic optimization of morphing airborne wind energy wings, in: *AIAA Scitech Forum*, 2019.
- [35] A. Iannelli, U. Fasel, N. Yogarajah, R. Smith, A balanced mode decomposition approach for equation-free reduced-order modeling of LPV aeroservoelastic systems, in: *AIAA SciTech Forum*, 2021.
- [36] M. Gavish, D.L. Donoho, The optimal hard threshold for singular values is $4/\sqrt{3}$, *IEEE Trans. Inf. Theory* 60 (8) (2014) 5040–5053.
- [37] B. Moore, Principal component analysis in linear systems: controllability, observability, and model reduction, *IEEE Trans. Autom. Control* 26 (1) (1981) 17–32.
- [38] K. Glover, All optimal Hankel-norm approximations of linear multivariable systems and their L_∞ -error bounds, *Int. J. Control* 39 (6) (1984) 1115–1193.
- [39] P. Benner, C. Himpe, Cross-gramian-based dominant subspaces, *Adv. Comput. Math.* 45 (5) (2019) 2533–2553.
- [40] C.W. Rowley, Model reduction for fluids, using balanced proper orthogonal decomposition, *Int. J. Bifurc. Chaos* 15 (03) (2005) 997–1013.
- [41] Z. Ma, S. Ahuja, C.W. Rowley, Reduced-order models for control of fluids using the eigensystem realization algorithm, *Theor. Comput. Fluid Dyn.* 25 (1) (2011) 233–247.
- [42] M. Viberg, Subspace-based methods for the identification of linear time-invariant systems, *Automatica* 31 (12) (1995) 1835–1851.
- [43] A. Laub, M. Heath, C. Paige, R. Ward, Computation of system balancing transformations and other applications of simultaneous diagonalization algorithms, *IEEE Trans. Autom. Control* 32 (2) (1987) 115–122.
- [44] S. Lall, J.E. Marsden, S. Glavaški, A subspace approach to balanced truncation for model reduction of nonlinear control systems, *Int. J. Robust Nonlinear Control* 12 (6) (2002) 519–535.
- [45] G. Golub, C. Van Loan, *Matrix Computations*, Johns Hopkins University Press, 2013.
- [46] P. Seiler, R. Moore, C. Meissen, M. Arcak, A. Packard, Finite horizon robustness analysis of LTV systems using integral quadratic constraints, *Automatica* 100 (2019) 135–143.
- [47] A. Iannelli, P. Seiler, A. Marcos, Worst-case disturbances for time-varying systems with application to flexible aircraft, *J. Guid. Control Dyn.* 42 (6) (2019) 1261–1271.
- [48] I. Markovsky, J.C. Willems, P. Rapisarda, B.L. De Moor, Algorithms for deterministic balanced subspace identification, *Automatica* 41 (5) (2005) 755–766.
- [49] D. Keidel, U. Fasel, L. Baumann, G. Molinari, P. Ermanni, Experimental validation of a morphing wing for airborne wind energy applications, in: *8th International Conference on Adaptive Structures and Technologies*, 2017.
- [50] U. Fasel, Reduced-order aeroelastic modeling of morphing wings for optimization and control, Ph.D. thesis, ETH, Zurich, 2020.
- [51] W.P. Rodden, E.H. Johnson, *User Guide V 68 MSC/NASTRAN Aeroelastic Analysis*, 1994.
- [52] D. Shepard, A two-dimensional interpolation function for irregularly-spaced data, in: *Proceedings of the 1968 23rd ACM National Conference*, 1968, pp. 517–524.
- [53] A.P.J. Katz, *Low-Speed Aerodynamics*, Cambridge University Press, 2001.
- [54] S. Waitman, A. Marcos, \mathcal{H}_∞ control design for active flutter suppression of flexible-wing unmanned aerial vehicle demonstrator, *J. Guid. Control Dyn.* 43 (4) (2020) 656–672.
- [55] J. Theis, H. Pfifer, P. Seiler, Robust modal damping control for active flutter suppression, *J. Guid. Control Dyn.* 43 (6) (2020) 1056–1068.
- [56] L. Ljung, *System Identification: Theory for the User*, Prentice Hall PTR, 1999.
- [57] J. Rawlings, D. Mayne, *Model Predictive Control: Theory and Design*, Nob Hill Pub., 2009.
- [58] L. Fagiano, M. Milanese, D. Piga, Optimization of airborne wind energy generators, *Int. J. Robust Nonlinear Control* 22 (18) (2012) 2055–2083.
- [59] R. Schmehl, *Airborne Wind Energy: Advances in Technology Development and Research*, Springer, Singapore, 2018.
- [60] E. Kaiser, J.N. Kutz, S.L. Brunton, Sparse identification of nonlinear dynamics for model predictive control in the low-data limit, *Proc. R. Soc. A, Math. Phys. Eng. Sci.* 474 (2018) 20180335.