The Aeronautical Journal (2022), **126**, pp. 1546–1567
doi:10.1017/aer.2022.4

REGULAR PAPER

Reduced-order model for efficient generation of a subsonic missile's aerodynamic database

A. Sinha[1], R. Kumar[2] and J. Umakant[2]

[1]Indian Institute of Technology Bombay, Dept. of Aerospace Engineering, Mumbai, India and [2]Defence Research and Development Laboratory, Hyderabad, India
Email: as@aero.iitb.ac.in

Received: 27 July 2021; **Revised:** 6 January 2022; **Accepted:** 7 January 2022

Keywords: ROM; Aerodynamics database generation; Missile aerodynamics

Abstract

This article reports on the development of a fast method for generating the aerodynamic database for subsonic flow over a missile. At present, this is typically achieved using RANS-based CFD, which is expensive for the complex missile geometries and the multiplicity of operating conditions to be evaluated. The presented reduced-order model (ROM) provides a reasonably accurate prediction of the aerodynamic coefficients of the missile (and, in fact, the full flow field around it) within half a minute. In particular, in the interpolative regime, prediction errors for all coefficients are typically less than 1% of their respective maximum values encountered in the database, with extrapolation incurring more error. The empirical approach 'learns' from the CFD solutions calculated for a few operating conditions, and then is able to make predictions for any other condition within a feasible set. The learning employs proper orthogonal decomposition (POD) to characterise the most important features of the flow. The prediction is posed as an optimisation problem that aims to find the flow solution as a linear combination of the above POD modes that minimises the residual of the governing equations. Innovations on the prevailing POD-ROM approach include novel implementation of boundary conditions, simplified computation of the aerodynamic coefficients, and a procedure for choosing modelling parameters based on extensive cross-validation. Challenges overcome in application to a problem of industrial relevance are discussed.

Nomenclature

$\boldsymbol{B}$	vector of boundary conditions that a flow field has to satisfy
C_A, C_S, C_N	axial, side and normal force coefficients of missile
C_l, C_m, C_n	roll, pitch and yaw moment coefficients of missile
D	diameter of missile
J	cost function in optimisation-based ROM
M	Mach number
$N^{c,R}$	number of grid cells that contribute to the ROM residual evaluation sub-domain
N_p	number of POD modes used for an expansion
N_s	number of 'snapshots' – steady flow solutions available in learning database
$\boldsymbol{R}, R_e$	vector of residuals of governing equations, and residual of the one denoted by e
$\widetilde{\boldsymbol{R}}, \widetilde{R}_e$	approximations of the above quantities using POD expansion
$\widetilde{\mathscr{R}}_e^c$	volume-integrated residual of e-equation over the cth grid cell
p	pressure; also signifies the $\mathcal{L}_p$-norm of a vector field (disambiguated by context)
$\boldsymbol{q}, q$	vector field of flow variables and any one of its scalar components
u, v, w	x, y and z components of flow velocity, respectively
Ω	flow domain
Ω_P	sub-domain over which the POD inner product is evaluated
Ω_R	sub-domain over which the ROM residual is evaluated

© The Author(s), 2022. Published by Cambridge University Press on behalf of Royal Aeronautical Society. This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (https://creativecommons.org/licenses/by/4.0/), which permits unrestricted re-use, distribution, and reproduction in any medium, provided the original work is properly cited.
https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

α	angle-of-attack of missile
ρ	density
$\boldsymbol{\mu}$	vector of operating condition parameters
$\eta^n, \boldsymbol{\eta}$	projection coefficient of a flow snapshot on to the nth POD mode, and vector of all such coefficients included in an expansion
ζ_s^n	weight of sth snapshot in learning database towards nth POD mode
λ^n	eigenvalue corresponding to the nth POD mode
ϕ	roll angle of missile
γ	ratio of specific heats at constant pressure and volume
$\overline{(\cdot)}$	mean of any flow quantity across all snapshots of the learning database
$(\cdot)'$	deviation of any flow quantity from its mean value in the learning database
$\widetilde{(\cdot)}^n$	POD eigenfunction (mode) number n corresponding to a flow quantity

1.0 Introduction

The design of a missile involves consideration of several variants of the geometry. A fundamental requirement of this process is a detailed aerodynamic database for each of these design choices, listing the forces and moments acting on the missile for a large range of conditions – e.g. various combinations of Mach numbers, angles of attack, and roll angles. Creation of such aerodynamic databases poses a significant cost to organisations involved in the development of novel configurations. With the availability of increasing computing power and improved numerical algorithms, most of this activity has shifted from physical flight tests and wind tunnel experiments to the virtual environment of computational fluid dynamics (CFD). The state-of-the-art CFD invariably involves a large-scale full-order model (FOM), a terminology that distinguishes this model from the more efficient, albeit approximate, reduced-order model (ROM) outlined subsequently. Depending on the accuracy required, the designer may seek quick Euler (inviscid) solutions or costlier Reynolds-averaged Navier-Stokes (RANS) solutions (that account for molecular and eddy viscosity). In either case, the state-of-the-art FOMs yield steady flow solutions because unsteady simulations are neither computationally feasible nor necessary for the engineering answers being sought. Unfortunately, for the complex missile configurations of practical relevance, the computational expense of generating large aerodynamic databases is still very high.

Over the years, a variety of methods have been explored for reducing the computational costs of these databases, with minimal attendant degradation in accuracy. The overall approach is termed model order reduction (MOR); Ref. [1] is an up-to-date review. All MOR methods involve empiricism; i.e. the designer must first create an FOM database (the so-called learning database) by sparsely sampling the freestream parameter domain (e.g. Mach number, angle-of-attack and roll angle) and/or other geometric parameters (like fin deflection angle) for a particular design configuration. These flow solutions are termed learning snapshots in this article. This step requires heavy computational effort – it represents the upfront cost of these empirical approaches. Subsequently, MOR promises to efficiently predict the flow solutions (or their salient properties, like aerodynamic coefficients) using the empirical database of snapshots. Note that fewer FOM snapshots need to be simulated in this approach with the rest being predicted at very low cost, so that the total cost incurred is much less than that in the generation of current state-of-the-art FOM databases.

Amongst MOR approaches, we focus on projection-based methods. In these, the flow solutions from the learning database are analysed to derive one or more empirical reduced-order bases (ROBs). Subsequently, the methods seek flow solutions that are projected on to these bases. Projection-based MOR methods fall in one of two broad categories – those based purely on interpolation and those that are based on minimising the residual of the governing equations. In the literature [2], these are also called non-intrusive vs. intrusive approaches. For the purpose of this article, an outcome of the projection-based residual minimisation MOR strategy will be exclusively called a reduced order model (ROM).

Interpolation methods, as the name suggests, involve interpolation of empirical snapshots based on their operating conditions to determine the flow solution for a new operating condition that was not

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

evaluated in the original database. These methods are very fast, and can deliver results of acceptable accuracy if the flow features are approximately similar between the new condition and those evaluated for the original database. A large variety of interpolation methods are reported in the literature. They may be classified based on the interpolated quantity. The simplest are those that directly interpolate the integrated aerodynamic quantities – viz. coefficients of lift, drag etc. Of intermediate complexity and accuracy are approaches that interpolate the pressure and shear stress on body surfaces [3]. The other end of the spectrum is occupied by methods that attempt to predict the solution over the entire flow domain for new parameter sets of interest, using interpolation of the projection coordinates of the solution with respect to an appropriate ROB [2, 4–6].

In discussing the projection-based residual-minimising MOR approaches for flows reported in the literature, we make a distinction between unsteady and steady ROMs. For problems dominated by time-varying flow fields, unsteady ROMs are required to approximately determine the essential flow features in a time-resolved manner. On the other hand, steady ROMs find utility in problems where knowledge of the steady (usually time-averaged) component of the flow field is sufficient for engineering purposes, as in the present case.

The steady ROM development starts from snapshots of the steady flow field for different values of a set of parameters of the problem, just like in the case of interpolation. The next step is the eduction of the ROB that represents the dominant flow features observed across all snapshots. In the literature, this is consistently pursued with proper orthogonal decomposition (POD) [5, 7–12]. POD is a statistical procedure that delivers an orthogonal transformation to convert a set of observations (snapshots) of possibly correlated variables (flow fields) into a reduced set of linearly uncorrelated variables (basis functions/modes) [13, 14]. The POD modes afford a reduced-order representation of the steady flow fields. Being an empirical method, the POD modes derived are entirely dependent on the richness of the snapshots – i.e. the snapshots should cover the range of flow fields that are expected to be encountered by the ROM.

The preceding steps constitute the offline component in the MOR procedure; they are performed once at the beginning. The final online step is the prediction of the steady flow solution for a particular parameter set that was not computed in the snapshot-creation step. In case of ROMs, this new flow solution is derived as a linear combination of the POD modes that optimally satisfies the steady governing equations of the flow along with appropriate boundary conditions [7–12]. The optimality is in terms of the $\mathcal{L}_p$-norm of the residual of the governing equations, where p is usually 1 or 2. The ROM method is more robust and versatile (albeit computationally expensive) than the interpolation method since it is guided by the governing equations. The ROM approach has been shown to be useful for predicting the steady aerodynamics of an aircraft wing/fin in both two-dimensional [7–9] and three-dimensional [10] problems; it has also been successfully applied to full aircraft configurations [5, 11, 12].

Here we report on the further development and application of the POD-ROM approach to the problem of predicting the aerodynamic characteristics of a missile. Although the missile's speed ranges from subsonic to supersonic, the scope of the present work is limited to a regime of subsonic missile flight where shocks are absent. Modelling shocks in a POD setting is complicated [6, 8, 15, 16], and will be pursued in the future. Moreover, we address variations in freestream conditions of the missile, but not in its geometric configuration, viz. fin deflection. Even with these restrictions, the ROM tool may partially replace expensive FOM calculations in aerodynamic characterisation, leading to large savings in computational time and cost. Indeed, we demonstrate how well the ROM can replicate the results of RANS CFD, at a very small fraction of the computational cost. Along the way, we also present on improvements in boundary condition formulation, aerodynamic coefficients evaluation and incorporation of physics-motivated efficiencies in the POD-ROM approach.

The article is organised as follows. Section 2 describes the missile aerodynamics problem under investigation. Proper orthogonal decomposition is briefly recapitulated in Section 3, prior to description of the ROM approach in Section 4. Results are presented and discussed in Section 5; these include description of the POD basis educed from the learning database, the choice of modelling options

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

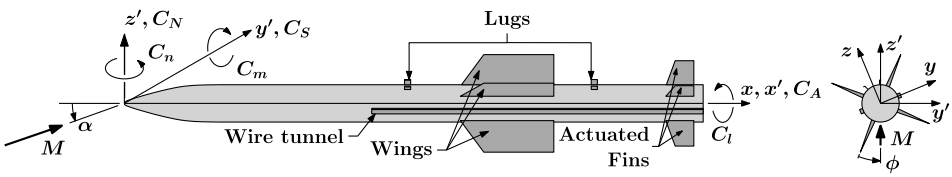

Figure 1. *Side and rear views of the typical missile configuration studied.*

based on cross-validation, validation with separate test cases, and a brief comparison with results from interpolation-based prediction. Final conclusions and outlook appear in Section 6.

2.0 Missile configuration and its FOM database

To motivate the exposition of the POD-ROM theory to follow, we first describe the air-launched missile configuration being studied; a sketch is shown in Fig. 1. This is a typical configuration with blunted tangent-ogive nose, cylindrical body, four wings, and all-movable in-line tail fins. The body has a slenderness ratio (length to diameter) 16.35, the nose has a fineness ratio of 1.75 with bluntness ratio of 20%, and the boat-tail has an angle of 8.6°. Although the fins are actuated for controlling the trajectory, they remain undeflected in the current work. In the nominal roll orientation, the wings and fins form an 'X' with respect to the plane containing the longitudinal axis and the relative wind vector. This also serves as the plane of mirror symmetry for a pair of external wire tunnels running along each side of the missile, and a set of four lugs near the top that anchor the missile underneath an airplane.

Two separate coordinate systems – xyz and $x'y'z'$ – are used to describe the missile. The origin of both are at the missile nose, and the coincident x- and x'-axes run along the longitudinal axis. In the body-fixed nominal coordinate system xyz, the y-axis is normal to the relative wind vector in the nominal roll orientation. In the industry, however, the total-angle-of-attack coordinate system $x'y'z'$ is prevalent, with the y'-axis being orthogonal to the relative wind vector. The two coordinate systems coincide in the nominal roll orientation. The convention for the roll angle ϕ is also specific to the industry. In particular, $\phi = 45°$ is the nominal roll orientation, with the relative wind vector lying in the first quadrant of the $x - z$ plane (and also the coincident $x' - z'$ plane). If the relative wind vector happens to be in the fourth quadrant of the $x - z$ plane, then we say that $\phi = -135°$. On the other hand, ϕ is 0° when the relative wind is in the plane of the bottom left wing/fin (as viewed from the rear); see Fig. 1. In summary, ϕ varies in the range $-180°$ to $+180°$, and the angle-of-attack α only takes positive values; usually $\alpha \leq 20°$.

Figure 1 also presents the sign convention for the aerodynamic force and moment coefficients, referred to the $x'y'z'$ coordinate system. The force coefficients are axial C_A, side C_S and normal C_N. The corresponding moment coefficients are roll C_l, pitch C_m and yaw C_n; the moment arm is measured from the missile nose. As usual, the characteristic length, area, speed and density for defining the coefficients are the missile diameter D, cross-sectional area $\pi D^2/4$, freestream speed and freestream density, respectively.

Table 1 presents the combinations of flight conditions comprising the learning database for the empirical ROM. There are two subsonic Mach numbers, six angles of attack and three roll angles; note that none of these cases displayed shocks in the flow field. All cases simulate flight at the same ambient conditions. Although all possible combinations should result in 36 snapshots, only 32 of these are unique since the roll angle ceases to matter at $\alpha = 0°$. If the minor protuberances of the two wire tunnels and four lugs were absent, then the missile would have had a four-fold rotational symmetry and a two-fold mirror symmetry. The minor symmetry-breaking features are indeed ignored in the industry for the purpose of generating the aerodynamic database, which explains the limited range of roll angles simulated. Direct predictions from our ROM will also be limited to this range; predictions at other roll angles can be obtained from symmetry arguments by ignoring the symmetry-breaking features, just as with the FOM database.

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

Table 1. *Freestream parameter combinations for missile aerodynamics learning database, and maxima of the six aerodynamic coefficients across the database*

Parameter	Values
Mach number, M	0.6, 0.8
Angle-of-attack, α (degrees)	0, 2, 5, 10, 15, 20
Roll angle, ϕ (degrees)	0, 22.5, 45
Maxima of C_A, C_S, C_N, C_l, C_m, C_n	0.45, 1.39, 10.18, 0.47, 102.95, 15.10

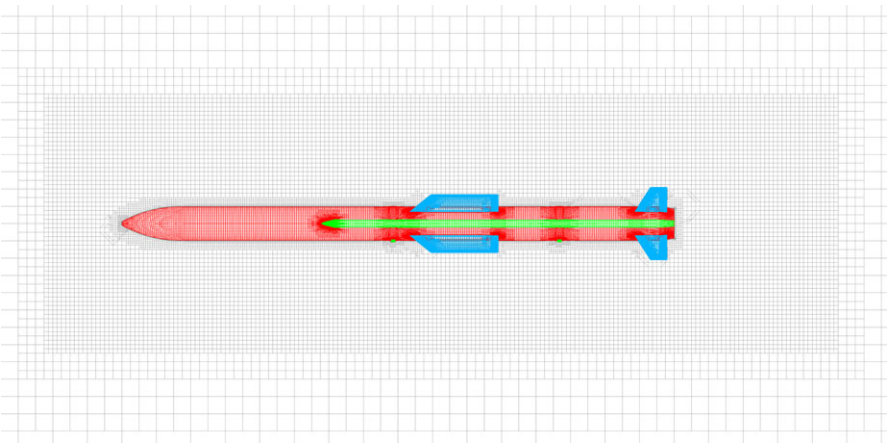

Figure 2. *Mesh used in CFD calculations on the missile surface and in the plane of mirror symmetry.*

The empirical database was generated using numerical simulation. The flow domain was a cube of side about 160 times the missile diameter D, centred on the nose. The edges of the cube were aligned with the body-fixed coordinate axes shown in Fig. 1. A three-dimensional hexahedral unstructured grid was generated in HEXPRESS (version 6.0; NUMECA International Company; Brussels, Belgium), with approximately 10 million cells (see Fig. 2). The mesh around the missile surface was refined to ensure a y^+ of about unity.

The flow simulations were performed using ANSYS Fluent (version 15.0) [17], solving the $k-\omega$ SST steady RANS equations using an implicit pressure-based coupled finite volume solver having second-order accuracy in space. Details of the solver can be found in the user manual. For future reference, we only note that the cell face values required in computing the convective terms were calculated with a second-order upwind scheme, using least-squares cell-based gradients.

A pressure far-field condition was specified on the inlet boundary consisting of the upstream face as well as the four side faces of the cubic flow domain. Zero gauge pressure and an absolute temperature of 303.15 K (sea level conditions) were consistently applied at the inlet for all snapshots. Only the Mach number and the flow direction were varied on the inlet per the specifications in Table 1. The incoming turbulence was specified by an intensity of 1% and a hydraulic diameter equal to D. The sixth face of the cube downstream of the missile constituted the outlet boundary on which a pressure outlet condition was specified. For this, the pressure, temperature and turbulence parameters were the same as for the inlet. The missile surface constituted a no-slip adiabatic wall.

3.0 A brief recapitulation of POD

What follows is a discussion of some of the aspects of Proper Orthogonal Decomposition [18] that are most relevant for the present work. Let $\boldsymbol{q}(\boldsymbol{x})$ represent the vector field of flow variables with the spatial

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

position coordinate $\boldsymbol{x}$ ranging over the flow domain, Ω, say. For the 3D Euler equations, one choice of $\boldsymbol{q}$ is $\left[\rho, \rho u, \rho v, \rho w, p\right]^{\mathrm{T}}$, where p and ρ are the pressure and density respectively, u, v and w are respectively the x, y and z Cartesian components of velocity, and $(\cdot)^{\mathrm{T}}$ is the matrix transpose operator. This particular choice is recommended in the literature [8, 10], and will be argued to be particularly appropriate subsequently. In the following, we use non-dimensional quantities. In particular, linear dimensions are non-dimensionalised by the diameter of the missile D, velocity by the ambient speed of sound c_∞, density by the ambient density ρ_∞, and pressure by $\rho_\infty c_\infty^2$. With this, the non-dimensional pressure in the freestream is γ^{-1}, γ being the ratio of the specific heats of air. Note that, per the discussion in Section 2, all the snapshots had the same values of D, c_∞ and ρ_∞, so that the above non-dimensional flow variables were directly comparable across the entire empirical database.

A particular steady operating condition of the missile (specified by, for example, the triad of its freestream Mach number, angle-of-attack and roll angle) is denoted by the vector $\boldsymbol{\mu}$; e.g. $\boldsymbol{\mu} = [M, \alpha, \phi]^{\mathrm{T}}$. Let such steady flow solutions be available for N_s operating conditions (for reference, $N_s = 32$ in our problem). Thus, the learning database of snapshots is $\{\boldsymbol{q}\,(\boldsymbol{x}; \boldsymbol{\mu}_s)\}_{s=1}^{N_s}$. Here, $\boldsymbol{\mu}_s$ denotes the operating condition of the sth snapshot. For notational convenience, we will sometimes denote this database as $\{\boldsymbol{q}_s(x)\}_{s=1}^{N_s}$.

Let the mean flow field across the database be $\overline{\boldsymbol{q}}(\boldsymbol{x}) := N_s^{-1} \sum_{s=1}^{N_s} \boldsymbol{q}_s(x)$. Then, the snapshots' fluctuations are given by $\boldsymbol{q}'(\boldsymbol{x}; \boldsymbol{\mu}) := \boldsymbol{q}(\boldsymbol{x}; \boldsymbol{\mu}) - \overline{\boldsymbol{q}}(\boldsymbol{x})$. In particular, the database of snapshot fluctuations becomes $\{\boldsymbol{q}'_s(x)\}_{s=1}^{N_s}$. In mean-subtracted vector POD, we assume that any flow vector field (not only those included in the learning database) can be approximated by a linear combination of N_p POD modal vector fields added to the above mean flow vector field (which is calculated on the learning database only):

$$\boldsymbol{q}(\boldsymbol{x}; \boldsymbol{\mu}) = \overline{\boldsymbol{q}}(\boldsymbol{x}) + \boldsymbol{q}'(\boldsymbol{x}; \boldsymbol{\mu}), \qquad \boldsymbol{q}'(\boldsymbol{x}; \boldsymbol{\mu}) \approx \sum_{n=1}^{N_p} \eta^n(\boldsymbol{\mu}) \widetilde{\boldsymbol{q}}^n(\boldsymbol{x}). \tag{1}$$

Here, $\widetilde{\boldsymbol{q}}^n$ is the nth POD mode; note that it is a *vector* field with the same components as $\boldsymbol{q}$. In particular, if p, say, is a component of $\boldsymbol{q}$, then the corresponding component of the POD mode $\widetilde{\boldsymbol{q}}^n$ is denoted by $\widetilde{p}^n$. Also, η^n is the nth POD coefficient; it is parameterised by the operating condition $\boldsymbol{\mu}$. For later use, we denote the vector of POD coefficients for operating condition $\boldsymbol{\mu}$ as $\boldsymbol{\eta}(\boldsymbol{\mu}) := \left[\eta^1(\boldsymbol{\mu}), \eta^2(\boldsymbol{\mu}), \cdots, \eta^{N_p}(\boldsymbol{\mu})\right]^{\mathrm{T}}$. Moreover, we denote the nth POD coefficient corresponding to the sth snapshot in the learning database by the shorthand $\eta_s^n := \eta^n(\boldsymbol{\mu}_s)$.

By contrast, in the mean-retained POD variant, the POD modes directly reconstruct the flow snapshots, and not their fluctuations. In scalar POD variant, each flow variable (p, ρv, etc. in the prevailing example) is decomposed into its own set of *scalar* POD modes, weighted by its respective *independent* set of POD coefficients. It has been shown conclusively that both mean-retained POD and scalar POD approaches degrade the utility of the subsequent ROM [5, 9].

To be able to assess the efficacy of the above approximation, the POD procedure requires an inner product to be defined *a priori*. As an example, in the present work, the inner product of two flow vector fields $\boldsymbol{q}$ and $\boldsymbol{r}$ is defined as

$$\langle \boldsymbol{q}(\cdot), \boldsymbol{r}(\cdot) \rangle := \int_{\Omega_P} \left[q_\rho r_\rho + q_{\rho u} r_{\rho u} + q_{\rho v} r_{\rho v} + q_{\rho w} r_{\rho w} + q_p r_p \right] \mathrm{d}\Omega_P, \tag{2}$$

where $q_{\rho v}$ refers to the $y-$component of momentum in the flow field $\boldsymbol{q}$, and so on, and $\Omega_P \subseteq \Omega$. As the individual variables are already non-dimensional, the above definition is dimensionally consistent.

The snapshot POD method [14] starts from the original approach [13] by noting that the modes must be in the space spanned by the fluctuation snapshots. That is,

$$\widetilde{\boldsymbol{q}}^n(\boldsymbol{x}) = \sum_{s=1}^{N_s} \zeta_s^n \boldsymbol{q}'_s(\boldsymbol{x}), \qquad \forall n \in [1, N_p]. \tag{3}$$

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

The POD modes are determined [14, 18] indirectly by calculating the weights ζ_s^n from the eigenvalue problem $N_s^{-1}\sum_{\ell=1}^{N_s}\langle \boldsymbol{q}'_\ell, \boldsymbol{q}'_s\rangle \zeta_\ell^n = \lambda^n \zeta_s^n, \forall s \in [1, N_s]$, which arises when attempting to minimise the error in the representation of Equation (3) in the sense of the norm induced by the inner product defined in Equation (3). The kernel of the snapshot POD eigenvalue problem is evidently symmetric and positive semi-definite. Therefore, the eigenvalues λ^n are guaranteed to be real and non-negative. The eigenvalues are ordered in decreasing order; i.e. $\lambda^1 \geq \lambda^2 \geq \cdots \lambda^{N_p} \geq 0$. With this sequencing, $\widetilde{\boldsymbol{q}}^n$ is called the nth POD mode; this also reflects the order of salience of their features in the flow database. One can conclude from the formulation of the snapshot method that $N_p \leq N_s$; physically, N_p should also be smaller than N_s so that the POD may filter out the inessential flow features or noise instead of over-fitting the data. The POD modes are also called POD eigenvectors or POD eigenfunctions. Owing to the special properties of the POD kernel mentioned above, the POD modes form a mutually orthogonal set. For ease of use, they are always unitised to obtain an orthonormal basis. Then, it follows from the POD expansion of Equation (3) that the POD coefficients are orthogonal projection coefficients: $\eta^n(\boldsymbol{\mu}) = \langle \boldsymbol{q}'(\cdot;\boldsymbol{\mu}), \widetilde{\boldsymbol{q}}^n(\cdot)\rangle$.

Remark 1. *The POD modes are spatial basis functions that are applicable to all choices of $\boldsymbol{\mu}$. As such, the $\boldsymbol{q}$ data for different $\boldsymbol{\mu}$'s must be presented to the POD algorithm on the same geometry and grid. Thus, without additional modifications, it is impossible to handle cases where the tail fins of the missile are deflected, and we omit such situations in this work.*

Remark 2. *A discontinuous function requires infinitely many basis functions for accurate reconstruction. Thus, without additional sophistication [6, 8, 15, 16], a POD basis cannot represent a set of flow fields with shocks at different locations. Here we remain strictly in the subsonic flow regime to avoid such issues.*

Remark 3. *The inner product underlying the POD method need not be evaluated over the entire flow domain. Instead, critical regions of the flow, say Ω_P, may be identified where the snapshots display the greatest variations [8]. The POD modes thus obtained may yet be used to represent the flow over the entire snapshot domain. This flexibility is leveraged here for computational efficiency at negligible sacrifice of accuracy.*

Remark 4. *The POD modes need not be computed in a global manner using all the available snapshots in the database. Instead, locally valid POD databases may be more useful when attempting to span large parameter domains [10]. This was not necessary in the problem under investigation.*

4.0 POD-ROM approach for steady aerodynamics

The basic methodology for creating a ROM for steady aerodynamics prediction of a missile is adopted from the approach of Ref. [7], with subsequent refinements reported in Refs [5, 8, 9, 10, 11, 12]. With the preceding notation of the vector field of flow variables $\boldsymbol{q}$ and the vector operating condition of the missile $\boldsymbol{\mu}$, the problem is formally stated as

$$\left(\frac{\partial \boldsymbol{C}}{\partial t} + \boldsymbol{R}\right)\boldsymbol{q}(\boldsymbol{x}, t; \boldsymbol{\mu}) = 0 \quad \forall \boldsymbol{x} \in \Omega,\ t \in \mathbb{R}^+, \qquad \text{subject to } \boldsymbol{B}(\boldsymbol{q}(\boldsymbol{x}, t; \boldsymbol{\mu})) = 0 \quad \forall \boldsymbol{x} \in \partial\Omega,\ t \in \mathbb{R}^+.$$

Here, $\boldsymbol{C}$ is the vector operator that maps the flow variable vector $\boldsymbol{q}$ to the conserved flow variable vector, and $\boldsymbol{R}$ is the residual operator that accounts for the terms other than the time-derivative in the conservation equations. The elements of $\boldsymbol{R}$ are denoted by $\{R_e\}_{e\in\mathscr{E}}$. As an example, for the 3-D Euler equations we have $\mathscr{E} := \{\rho, u, v, w, E\}$, signifying the conservation equations for mass, the three components of momentum, and energy, respectively. These vector conservation equations must be satisfied over the flow domain Ω at all times. The vector of conditions that must be satisfied at all times on the domain boundary $\partial\Omega$ are represented by the $\boldsymbol{B}$ operator. Since we are interested in the *steady* solution of the governing equations for a particular choice of $\boldsymbol{\mu}$, we need to find the vector field $\boldsymbol{q}(\boldsymbol{x};\boldsymbol{\mu})$ such that $\boldsymbol{R}(\boldsymbol{q}(\boldsymbol{x};\boldsymbol{\mu}))$ vanishes everywhere in Ω, subject to the boundary conditions.

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

With the approximate POD expansion of Equation (1), we have the steady residual as

$$\boldsymbol{R}(\boldsymbol{q}(\boldsymbol{x};\boldsymbol{\mu})) \approx \boldsymbol{R}\left(\overline{\boldsymbol{q}}(\boldsymbol{x}) + \sum_{n=1}^{N_p} \eta^n(\boldsymbol{\mu})\widetilde{\boldsymbol{q}}^n(\boldsymbol{x})\right) =: \widetilde{\boldsymbol{R}}(\boldsymbol{x};\boldsymbol{\eta}(\boldsymbol{\mu})),$$

since $\overline{\boldsymbol{q}}(\boldsymbol{x})$ and $\{\widetilde{\boldsymbol{q}}^n(\boldsymbol{x})\}_{n=1}^{N_p}$ are constant in the procedure. Similarly, we also approximate $\boldsymbol{B}(\boldsymbol{q}(\boldsymbol{x};\boldsymbol{\mu}))$ using the POD expansion, and denote the outcome as $\widetilde{\boldsymbol{B}}(\boldsymbol{x};\boldsymbol{\eta}(\boldsymbol{\mu}))$. Evidently, $\widetilde{\boldsymbol{R}}$ and $\widetilde{\boldsymbol{B}}$ depend on $\boldsymbol{\mu}$ through $\boldsymbol{\eta}$ only; they also depend implicitly on N_p.

Since the truncated POD expansion is an approximation, in general we will not be able to find $\boldsymbol{\eta}(\boldsymbol{\mu})$ such that $\widetilde{\boldsymbol{R}}$ vanishes exactly, simultaneously throughout the domain. Thus, we look to *minimise* the pth power of the p-norm of $\widetilde{\boldsymbol{R}}$; the minimum will correspond to the closest approximation to the desired solution that is possible with the retained POD modes. The optimisation problem is thus formulated as

$$\min_{\boldsymbol{\eta}} J(\boldsymbol{\eta}(\boldsymbol{\mu})) := \min_{\boldsymbol{\eta}} \left|\left|\widetilde{\boldsymbol{R}}(\cdot;\boldsymbol{\eta}(\boldsymbol{\mu}))\right|\right|_{\mathcal{L}_p}^p \qquad \text{for a given } \boldsymbol{\mu}. \tag{4}$$

Note that the boundary conditions are implicitly applied in the evaluation of the residual. Omitting the dependence of $\boldsymbol{\eta}$ on $\boldsymbol{\mu}$ for notational brevity, the discretised implementation is given by

$$J(\boldsymbol{\eta}) = \sum_{e\in\mathscr{E}} \int_{\Omega_R} \left|\widetilde{R}_e(\boldsymbol{x};\boldsymbol{\eta})\right|^p \mathrm{d}\Omega_R \approx \sum_{e\in\mathscr{E}} \sum_{c=1}^{N^{c,R}} \forall_c^{1-p} \left|\widetilde{\mathscr{R}}_e^c(\boldsymbol{\eta})\right|^p, \qquad \widetilde{\mathscr{R}}_e^c(\boldsymbol{\eta}) := \int_{\forall_c} \widetilde{R}_e(\boldsymbol{x};\boldsymbol{\eta})\mathrm{d}\forall_c. \tag{5}$$

Here, $\Omega_R \subseteq \Omega$ is the domain on which the residual is evaluated (see Section 4.2 for more details), $N^{c,R}$ is the number of finite-volume cells that constitute this domain, and $\forall_c$ is the volume of the cth cell. The volume-integrated residual of the e-equation on the cth cell is denoted by $\widetilde{\mathscr{R}}_e^c$. The task of finding the steady state solution $\boldsymbol{q}$ for the particular operating condition $\boldsymbol{\mu}$ is thus transformed into an optimisation problem for $\boldsymbol{\eta}$.

4.1 Computation of the cost function

Although the flow solutions in the learning database generally come from high-fidelity RANS, the residual may be computed using a low fidelity Euler solver [5, 8–10]. This is because the lower-order POD modes, being representative of global behaviour, cannot resolve the near-wall localised viscous/turbulent effects in any case. However, they indirectly account for these effects since they are derived from the RANS results that incorporate the no-slip condition at the walls. The conservative form of the non-dimensional steady Euler equations for a calorically perfect gas in Cartesian coordinates (with $\boldsymbol{u} := [u, v, w]^{\mathrm{T}}$ and $V := \sqrt{u^2+v^2+w^2}$) are

$$\begin{aligned} &\nabla\cdot(\rho\boldsymbol{u}) = 0, \qquad \nabla\cdot(\rho\boldsymbol{u}u) + \frac{\partial p}{\partial x} = 0, \qquad \nabla\cdot(\rho\boldsymbol{u}v) + \frac{\partial p}{\partial y} = 0, \\ &\nabla\cdot(\rho\boldsymbol{u}w) + \frac{\partial p}{\partial z} = 0, \qquad \nabla\cdot\left(\rho\boldsymbol{u}\left(\frac{\gamma p}{(\gamma-1)\rho} + \frac{V^2}{2}\right)\right) = 0. \end{aligned} \tag{6}$$

We start evaluating the cost function by integrating the above equations over each grid cell. Gauss' divergence theorem is used to convert the cell-volume integral to an integral over the surface of the cell. For example, the volume-integrated residuals of the mass, $x-$momentum and energy equations over the cth cell become (see definition in Equation (5))

$$\widetilde{\mathscr{R}}_\rho^c = \sum_{f=1}^{N_f^c} \left[\rho\boldsymbol{u}\cdot\hat{\boldsymbol{n}}A\right]_f^c, \quad \widetilde{\mathscr{R}}_u^c = \sum_{f=1}^{N_f^c} \left[(\rho\boldsymbol{u}\cdot\hat{\boldsymbol{n}}u + p\hat{n}_x)A\right]_f^c, \quad \widetilde{\mathscr{R}}_E^c = \sum_{f=1}^{N_f^c} \left[\rho\boldsymbol{u}\cdot\hat{\boldsymbol{n}}\left(\frac{\gamma p}{(\gamma-1)\rho} + \frac{V^2}{2}\right)A\right]_f^c, \tag{7}$$

where $[\cdot]_f^c$ denotes the surface integral over the fth face of the cth cell, and N_f^c is the number of faces of the cth cell. Also, A represents surface area, and $\hat{\boldsymbol{n}}$ is the outward unit normal, with Cartesian components $\hat{n}_x$, $\hat{n}_y$ and $\hat{n}_z$.

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

It is evident from the above expressions that we need the flow variables on the faces of each cell, say at the face centroids. Moreover, the face-integrated mass flow $[m]_f^c := \left[\rho \boldsymbol{u} \cdot \hat{\boldsymbol{n}} A\right]_f^c$ appears in every residual term, and thus must be reconstructed as accurately as possible. One option that is adopted here is to rely on the fact that mass conservation is satisfied in the above form for each steady FOM solution in the database. Since the POD modes (as well as the mean flow field) are a linear combination of the FOM solutions (see Equation (3)), each POD mode (and the mean flow field), as well as the solutions reconstructed therefrom, will satisfy mass conservation automatically if ρu, ρv and ρw on the cell faces are chosen as POD variables (as is the case here). The other two independent variables in the POD are density and pressure; the choice of the latter is particularly desirable for ease of computing the aerodynamic coefficients (see Section 4.4). In summary, to facilitate accurate evaluation of the residual, the POD is performed on face-based data alone.

Given a candidate set of optimisation variables (POD coefficients), we reconstruct the face-based pressure, density and three components of momentum vector using the POD eigenfunctions and the mean flow field. The mass flow $[m]_f^c$ through the fth face of the cth cell is calculated as the dot product of the reconstructed momentum vector $[\rho \boldsymbol{u}]_f^c$ and the pre-computed face area vector $\left[\hat{\boldsymbol{n}} A\right]_f^c$. Then, for example, the $\left[\rho \boldsymbol{u} \cdot \hat{\boldsymbol{n}} u A\right]_f^c$ term in the x-momentum equation is computed as $[m]_f^c \, [\rho u]_f^c \, / \, [\rho]_f^c$.

The literature on steady ROM aerodynamics [8, 10, 19] has a discussion on the choice of the p-norm. The $\mathcal{L}_1$-norm has been recommended rather than the more prevalent $\mathcal{L}_2$-norm, so as to avoid giving greater weightage to the residuals in regions of the solution domain where the FOM incurred greater errors; we verified this too. Thus, specialising Equation (5), the final expression for the cost function is

$$J(\boldsymbol{\eta}(\boldsymbol{\mu})) = \sum_{c=1}^{N^{c,R}} \left[\left| \widetilde{\mathscr{R}}_u^c(\boldsymbol{\eta}(\boldsymbol{\mu})) \right| + \left| \widetilde{\mathscr{R}}_v^c(\boldsymbol{\eta}(\boldsymbol{\mu})) \right| + \left| \widetilde{\mathscr{R}}_w^c(\boldsymbol{\eta}(\boldsymbol{\mu})) \right| + \left| \widetilde{\mathscr{R}}_E^c(\boldsymbol{\eta}(\boldsymbol{\mu})) \right| \right]. \tag{8}$$

4.2 Hyper-reduction

One of the prominent recommendations from the literature [8, 10, 11, 20, 21] is that the residual need not be evaluated over the entire domain. This is because the ROM problem is already severely over-determined, with the residual having to be minimised on many more cells than there are degrees of freedom (the number of POD mode coefficients – η's). Thus, critical regions of the flow may be determined *a priori* and the residual may be evaluated in those regions alone. That is, the domain Ω_R of evaluation of the ROM residual in Equation (5) may be a subset of the flow domain Ω; recall that the POD inner product may also have been evaluated over a limited domain $\Omega_P \subseteq \Omega$. If Ω_R is chosen properly, then the accuracy of the ROM may be preserved while significantly reducing the computational overhead of the cost function evaluation in the optimisation. This approach to model order reduction is called hyper-reduction [1, 22].

In aerodynamics applications, the appropriate Ω_R is usually identified heuristically. For example, as we will show later, regions of the flow near the missile surface (but not too near) may be most useful for the ROM. Such physics-based choices have been found to be fruitful earlier too [8, 11, 20]. The alternative to these heuristic approaches are algorithmically rigorous methods that select the subset of cells in Ω that best satisfy a mathematical criterion. Examples include the missing point estimate [23], the discrete empirical interpolation method [24], the Gauss-Newton with approximated tensors approach [25], and the least-squares collocation method [21]. Unfortunately, for large-scale aerodynamics problems, these methods often become computationally infeasible in the offline model-development phase itself, forcing one back to the heuristic approaches.

4.3 Implementing boundary conditions

Let us consider generic Dirichlet and Neumann boundary conditions, and analyse how they translate in the POD expansion. For this, we will consider two sub-cases. *Constant boundary condition* refers to

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

a case where the same numerical value of the boundary condition applies to all the snapshots (in the learning database as well as in the test cases). *Varying boundary condition* implies that different numerical values of the boundary condition prevail in the various snapshots. The first sub-case is exemplified by no-slip (Dirichlet) condition at a wall, and zero pressure gradient (Neumann) condition at an outlet. Varying boundary condition is exemplified by various freestream velocities (Dirichlet) being imposed on the inlet boundary to simulate different operating conditions of a missile.

Constant Dirichlet condition:

Suppose a particular variable, say q, is constrained to be $\widehat{q}$ on (a part or the entirety of) the boundary (wall, inlet, outlet, etc.) in all snapshots. Then, $q = \widehat{q}$ on that surface in the mean flow field, and $q = 0$ thereat in all the POD modes. Since all POD-reconstructed candidate solutions are linear combinations of the POD modes plus the mean flow field, they automatically satisfy this boundary condition without needing any explicit constraints in the optimisation. Of course, the POD modes cannot be used to compose the solution if the boundary condition value is changed from $\widehat{q}$.

The above analysis applies only when $\widehat{q}$ is one of the variables whose POD modes are being computed. Consider the example of a specified mass flux boundary condition imposed at a choked nozzle throat. If momentum is one of the POD variables, then the above statements hold. However, if density and velocity are the POD variables, then the throat mass flux corresponding to a POD-reconstructed flow is no longer a linear combination of the throat mass fluxes in the individual POD modes, and therefore this simplification is rendered invalid.

As an example of particular relevance, the no-slip condition at a wall is automatically satisfied by all solutions, as long as all components of velocity or momentum are expanded in a linear combination of their POD modes (and the mean field). As another example, if the freestream pressure and/or temperature are the same for all supplied snapshots of missile aerodynamics (as in our learning database), then there is no need to impose an explicit freestream pressure/temperature condition in the POD-ROM. The latter example of implementation of a constant but non-zero Dirichlet boundary condition demonstrates a benefit of subtraction of the snapshot mean in the POD. Otherwise, the constant freestream pressure/density (for example) would have to be reconstructed from the POD modes, necessitating explicit implementation.

Constant Neumann condition:

Following the above arguments, this boundary condition also does not give rise to any additional constraint on the POD modes, since all of them will have zero gradient at the particular boundary, and the mean flow field will have the specified (possibly non-zero) constant gradient at the boundary. For example, if outlet pressure gradient is constrained to be zero in flow simulations, then this is implicitly satisfied by all POD-ROM solutions also.

Varying Dirichlet condition:

This case will be discussed with a relevant example – the solution of the POD-ROM for a particular freestream velocity. The latter condition of a missile is usually specified in terms of its freestream Mach number, angle-of-attack, and roll angle. However, the primitive variables in the POD are the three components of momentum vector, along with density and pressure. Thus, recalling the caveat mentioned in the case of imposing a *constant* Dirichlet condition, we must first determine the freestream (i.e. the inlet boundary) momentum vector of a snapshot, denoted here by $\widehat{\rho \boldsymbol{u}}$. Since the normalised density is unity at the inlet for all snapshots, this is actually indistinguishable from $\widehat{\boldsymbol{u}}$ in our problem.

Suppose that the momentum takes on boundary values of $\{\widehat{\rho \boldsymbol{u}}_s\}_{s=1}^{N_s}$ respectively in the N_s snapshots constituting the learning database underlying the POD. Then, the mean flow field will have the average freestream momentum $\overline{\widehat{\rho \boldsymbol{u}}} = N_s^{-1} \sum_{s=1}^{N_s} \widehat{\rho \boldsymbol{u}}_s$ at all points of the inlet boundary. We next determine the fluctuation boundary values of the snapshots $\{\widehat{\rho \boldsymbol{u}}'_s\}_{s=1}^{N_s}$, where $\widehat{\rho \boldsymbol{u}}'_s := \widehat{\rho \boldsymbol{u}}_s - \overline{\widehat{\rho \boldsymbol{u}}}$. Then, referring to Equation

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

(3), the freestream momentum value of the nth POD mode can be directly found as $\widehat{\widetilde{\rho \boldsymbol{u}}}^n = \sum_{s=1}^{N_s} \zeta_s^n \widehat{\rho \boldsymbol{u}}_s'$, instead of actually evaluating the POD mode. Finally, to obtain the freestream momentum value of $\widehat{\rho \boldsymbol{u}}_0$ corresponding to a particular operating condition $\boldsymbol{\mu}_0$ requires the satisfaction of the following set of linear equations by the vector POD coefficients $\boldsymbol{\eta}(\boldsymbol{\mu}_0)$ during the optimisation:

$$\overline{\widehat{\rho \boldsymbol{u}}} + \sum_{n=1}^{N_p} \eta^n(\boldsymbol{\mu}_0) \widehat{\widetilde{\rho \boldsymbol{u}}}^n = \widehat{\rho \boldsymbol{u}}_0.$$

The advantage of this novel implementation is two-fold: (i) we enforce the crucial freestream boundary condition exactly, and (ii) we do not have to evaluate the POD-reconstructed flow solution at the inlet boundary.

Varying Neumann condition:

Such boundary conditions, if present, can also be addressed in a manner similar to the preceding varying Dirichlet conditions. We simply expand the set of simultaneous equality constraints to be satisfied during the optimisation.

4.4 Evaluation of aerodynamic coefficients

The total surface force vector on a body immersed in a turbulent flow is given by

$$\boldsymbol{F} = \int_A \left(-p\hat{\boldsymbol{n}} + \mathbb{T}\cdot\hat{\boldsymbol{n}}\right) \mathrm{d}A =: \boldsymbol{P} + \boldsymbol{T}, \qquad \mathbb{T} = (\mu + \mu_t)\left(\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^{\mathrm{T}} - \frac{2}{3}\nabla\cdot\boldsymbol{u}\mathbb{I}\right).$$

Here, $\hat{\boldsymbol{n}}$ is the local unit normal on the surface, $\mathbb{T}$ is the combined viscous and turbulent stress tensor, μ is the (laminar) viscosity, μ_t is the turbulent eddy viscosity, $\mathbb{I}$ is the identity tensor, and A is the body's surface area. Note that p and $\boldsymbol{u}$ are the Reynolds-averaged pressure and velocity, as in our snapshots. Pressure-based forces are accounted for by $\boldsymbol{P}$, whereas $\boldsymbol{T}$ gives the contribution of viscous and turbulent stresses, both shear and normal. Below we propose a very simple method for calculating the exact $\boldsymbol{P}$ and an approximation of $\boldsymbol{T}$.

Let the integrated surface pressure force for the sth snapshot in the original database be $\boldsymbol{P}_s$. This is known from the pressure contribution to the aerodynamic force coefficients (calculated by the CFD software, say) multiplied by $\rho_\infty U_\infty^2 A_{ref}/2$. Here ρ_∞ and U_∞ are the freestream density and speed, respectively, and A_{ref} is the reference area (usually square of the missile diameter). Note that, if the database is computed for various missile speeds and/or flight altitudes, the above multiplier will be different for each snapshot. Let the mean pressure force on the missile across the database be $\overline{\boldsymbol{P}}$, and the consequent snapshot-wise fluctuation pressure forces be $\{\boldsymbol{P}'_s\}_{s=1}^{N_s}$. Since pressure is one of the POD variables, $\boldsymbol{P}$ for any condition is linear in the POD coefficients $\boldsymbol{\eta}$ as

$$\boldsymbol{P} = -\int_A \left(\overline{p} + \sum_{n=1}^{N_p} \eta^n \widetilde{p}^n\right) \hat{\boldsymbol{n}} \mathrm{d}A = \overline{\boldsymbol{P}} + \sum_{n=1}^{N_p} \eta^n \widetilde{\boldsymbol{P}}^n, \quad \widetilde{\boldsymbol{P}}^n := -\int_A \widetilde{p}^n \hat{\boldsymbol{n}} \mathrm{d}A = \sum_{s=1}^{N_s} \zeta^n{}_s \boldsymbol{P}'_s.$$

Evidently, $\widetilde{\boldsymbol{P}}^n$ is the pressure force corresponding to the nth POD mode. As shown by the second equation above, it can be computed at no extra expense with the known $\{\boldsymbol{P}'_s\}_{s=1}^{N_s}$ and the snapshot weight vectors towards POD modes $\{\boldsymbol{\zeta}^n\}_{n=1}^{N_p}$ (see Equation (3)). The pressure moment vector on the missile can also be calculated from an analogous expression.

On the other hand, the viscous and turbulent force $\boldsymbol{T}$ not only involves the eddy viscosity μ_t that is omitted from POD, but is also nonlinear as $\rho\boldsymbol{u}$ and ρ are the decomposed variables instead of $\boldsymbol{u}$. Moreover, even μ is a nonlinear function of temperature, which in turn is nonlinearly related to the POD variables density and pressure. Thus, the above straightforward linear relation between the POD coefficients $\boldsymbol{\eta}$ and $\boldsymbol{P}$ should not be extended to $\boldsymbol{T}$ (and hence to $\boldsymbol{F}$). However, we now argue for just that. For the subsonic database being considered, the wall temperature is not expected to vary significantly.

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

This not only means that μ may be assumed to be constant at the wall, but it also implies that pressure and density should be almost proportional. Now, the wall-normal pressure gradient is known to approximately vanish in a boundary layer, so that the wall-normal density gradient should also not be strong. But it is this gradient that appears in $\boldsymbol{T}$, when we expand $\boldsymbol{u}$ in terms of POD modes (that have ρ and $\rho\boldsymbol{u}$ variables). Finally, μ_t is not even retained in our POD, and thus there is no good justification in neglecting its surface variations between snapshots, other than the subsequent demonstration through our results.

In summary, with partial rationalisation, we assume that total forces and moments (not just the pressure components) are linear functions of the POD coefficients. That is,

$$\boldsymbol{F} \approx \overline{\boldsymbol{F}} + \sum_{n=1}^{N_p} \eta^n \widetilde{\boldsymbol{F}}^n, \qquad \text{with} \quad \widetilde{\boldsymbol{F}}^n := \sum_{s=1}^{N_s} \zeta^n{}_s \boldsymbol{F}'_s \quad \text{and} \quad \overline{\boldsymbol{F}} := \frac{1}{N_s} \sum_{s=1}^{N_s} \boldsymbol{F}'_s, \tag{9}$$

with the meaning of $\overline{\boldsymbol{F}}$, $\boldsymbol{F}'$, and $\widetilde{\boldsymbol{F}}$ respectively carrying over from $\overline{\boldsymbol{P}}$, $\boldsymbol{P}'$, and $\widetilde{\boldsymbol{P}}$ above. The advantage is that we obtain a procedure for evaluating the aerodynamic coefficients of a reconstructed solution without having to do the actual reconstruction and the surface integration. Of course, if one finds this approximation to be tenuous in a certain application, then it is easy to revert to surface integration to get the exact aerodynamic coefficients.

It should be remarked that the above forces and moments are necessarily obtained in a body-fixed coordinate system (that which remains anchored in the consistent grid for all snapshots). If one desires the aerodynamic coefficients in the total-angle-of-attack reference frame, then the appropriate transformation has to be made separately. Evidently, the force and moment data for the snapshots must also be presented in the body frame to the above algorithm.

5.0 Results and discussion

5.1 POD results

5.1.1 POD modes

Proper orthogonal decomposition has two main ingredients – the selection of the snapshots is the obvious one; the definition of the underlying inner product is the other choice for the designer. As pointed out in the literature (10), the domain of integration Ω_P in the inner product can be a limited sub-domain in the vicinity of the missile. It need not include the very near-surface region, since the local flow structures thereat will be filtered out by the global nature of POD in any case. Also, the domain need not include regions far from the missile, as they will be irrelevant for predicting the forces and moments on the missile.

Two separate sets of POD (and ROM) calculations were performed with the set of 32 learning snapshots. The integration domain in the first POD analysis was the full flow domain, which is a cube of sides approximately 160 times the missile diameter D. In the second case, the integration domain was an annular region encasing the missile (see Fig. 3(a) and also Fig. 4). It consisted of the cells of the unstructured grid whose centroids were more than $0.2D$ but less than $2\,D$ away from any point on the surface of the missile – a 30-fold reduction in the number of grid cells. This choice will be explained subsequently; here we will evaluate if there is a penalty to be paid for this benefit.

Figure 3(b) presents the fractional eigenspectra of the two POD cases. It appears that the first three eigenvalues are many orders-of-magnitude larger than the fourth and subsequent ones. Also, the major difference in the two eigenspectra is the quantum of decrease of the eigenvalues between the third and fourth POD modes.

The above observations can be explained with reference to Fig. 4 that presents the mean flow field and the first few POD eigenfunctions for the reduced-domain POD. The $\overline{\rho u}$ field appears to be around 0.7 over the entire domain, except very close to the missile. This is understandable since half the snapshots were at Mach 0.6 and the other half were at Mach 0.8. Note that the mean velocity at the missile surface is zero in every snapshot, so that the near-missile mean velocity should indeed be smaller than the mean freestream

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

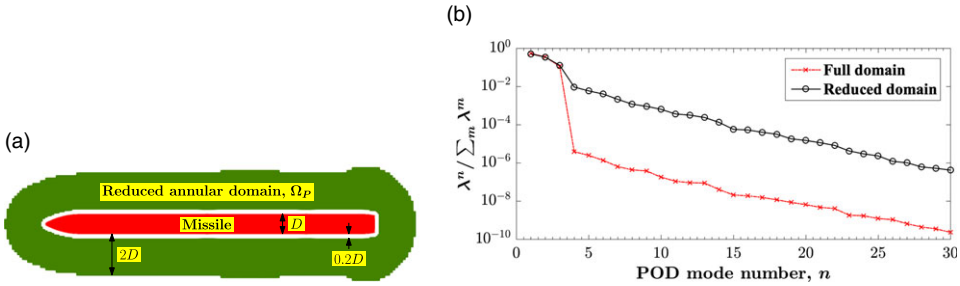

Figure 3. *(a) Reduced annular domain Ω_P around the missile for POD, shown in a longitudinal section. (b) POD eigenspectra from two separate calculations using domains Ω and Ω_P.*

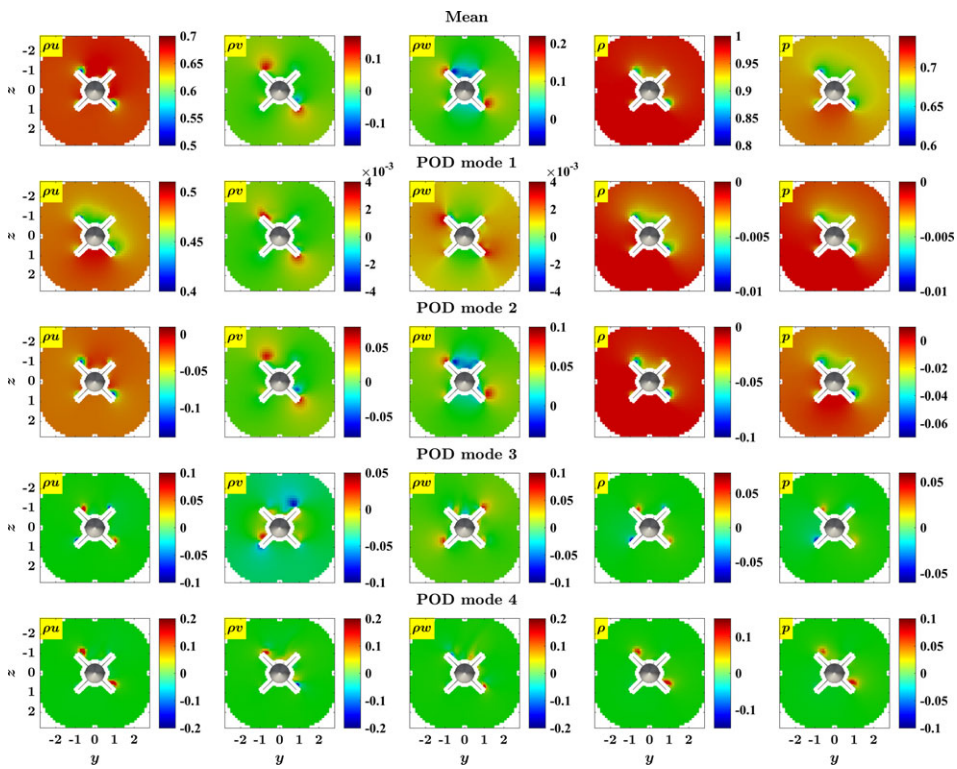

Figure 4. *The five components (along columns) of the mean flow field and of the first four POD eigenfunctions (along rows), at a section through the main fins of the missile.*

velocity. Also, the above Mach numbers correspond to the velocity magnitude, which is slightly larger than the mean x-component being portrayed in the plot. The learning database has snapshots with roll angles of 45°, 22.5° and 0°, that have increasing y-components of freestream velocity starting from zero. This explains the slight positive $\overline{\rho v}$ field, with higher values at the tips of the two wings around which the flow accelerates. Also, the choices of angles of attack and the roll angles in the learning database means that the z-component of the freestream flow is either zero (in case $\alpha = 0$) or positive in the snapshots. This is reflected in the overall positive $\overline{\rho w}$ field, with higher values at the same wing tips as in the $\overline{\rho v}$ case, and lower values where there might be flow retardation or even separation. The $\overline{\rho u}$ field is also reflected in the mean density and pressure fields; they relax to unity and $\gamma^{-1} = 0.714$, respectively, far

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

away from the missile. Notice that towards the outer edge of the reduced annular domain depicted, the mean flow field tends to be uniform.

The first POD mode is dominated by the ρu field, which is two orders-of-magnitude higher than the other components of the momentum. This mode captures the major variation in the database between the Mach 0.6 and Mach 0.8 snapshots, which is an increased freestream u. (The precise magnitude of around 0.5 away from the missile is an artifact of the normalisation of the POD eigenfunction; one should only focus on the relative values of the POD eigenfunctions.) Note that the density and pressure components of the first POD mode faithfully track the ρu component, but they tend to zero far from the missile since all snapshots have the same freestream density and pressure that are already accounted for in the mean flow field.

The second POD mode is dominated by the ρw field; this is to be expected since the next major variation in the database (after the one captured by the first POD mode) is that due to different angles of attack from 0° to 20°. The third POD mode has non-zero ρv away from the missile; it captures the remaining distinctive variation of the roll angles in the database. The fourth POD mode has non-trivial values only towards the inner edge of the annular domain, and this trend persists in all higher-order modes that are not shown. All through these four modes, the behaviour of the ρu, density and pressure components appear to be very similar.

A clear explanation of the observations in Fig. 3(b) is obtained from the results and discussion of Fig. 4. The full domain differs from the reduced annular domain in its inclusion of the very near-surface region and the outer region. However, the latter sub-region is much larger in volume (and hence, weightage in the inner product) than the former. Moreover, the flow field is more or less uniform in the outer region omitted in the reduced domain. This explains why the relative magnitudes of the first three POD eigenvalues (that capture the far-surface flow variations in the database) are the same with the two domain choices. It also explains why the remaining parts of the POD eigenspectra are similar in the two cases, but only of much smaller relative magnitude in the case of the full domain. Basically, both choices display similar flow features, and both can only capture the very near-surface flow phenomena in very high order of modes.

5.1.2 POD-reconstruction of aerodynamic coefficients of cases in learning database

The pressure component of the aerodynamic forces and moments on a missile is not significantly affected by flow features that are extremely close to the surface. The friction component is apparently affected by flow in the boundary layer; however, this too is driven by the outer flow, and the boundary layer physics are also included in the snapshots already. This suggests that although POD is very inefficient at capturing the very near-surface flow features, it may still be useful for predicting the aerodynamic coefficients of a missile. This will indeed be validated here.

Before using the POD basis in predicting new cases, we evaluate its effectiveness in reproducing the aerodynamic coefficients of the cases in the learning database. In particular, Fig. 5 presents the error in reconstructing the aerodynamic force and moment coefficients of all the learning cases using the first 25 POD modes calculated on the reduced domain (depicted in Fig. 3(a)). Here and subsequently, the errors are normalised by the respective maxima across the learning database, as indicated in Table 1. In general, the maximum error occurs for lower angles of attack. This may be a consequence of the very localised flow features that occur at lower α's, which are only captured by the very highest order POD modes. The axial force coefficient C_A presents the lowest fractional error. This is interesting since it is dominated by shear stresses, which are actually nonlinear functions of the POD coefficients (see Section 4.4), but are approximated as linear functions here. The normal force and pitching moment coefficients present very low fractional errors. Their cross-sectional counterparts – viz. the side force and yawing moment coefficients – present the highest errors (although these are still less than 5%); this is probably an artifact of the order-of-magnitude smaller maximal value used in the normalisation of these errors. Referring to Fig. 1, for the slender missile body, the normal force is primarily responsible for the pitching moment, and the yawing moment is mostly due to the side force. This explains why the two pairs appear to go

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

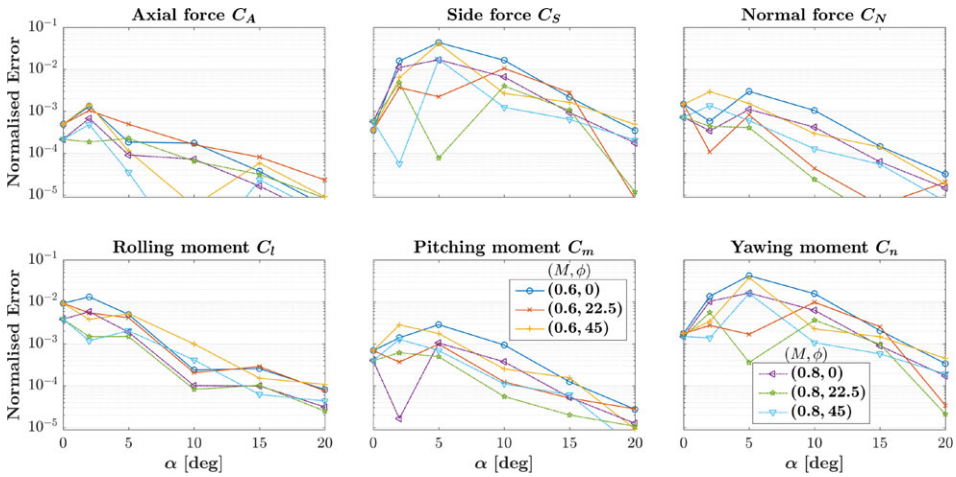

Figure 5. *Errors in reconstruction of aerodynamic coefficients with 25 POD modes.*

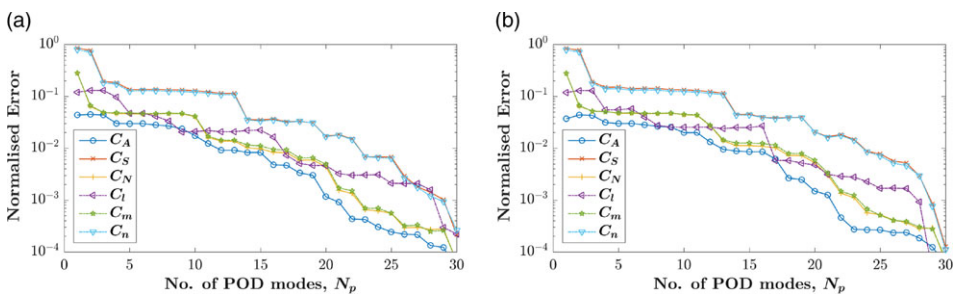

Figure 6. *Database-averaged error in POD reconstruction of aerodynamic coefficients.*

hand in hand across all the cases in Fig. 5. The rolling moment error is intermediate in magnitude, but is usually still less than 1%. It is in fact very difficult to predict accurately even in FOM analyses, owing to its insignificance compared to the other two moment components. In the interest of brevity, we will not show the moment coefficients in most of the subsequent results, concentrating instead on the force coefficients only.

The above error analysis was performed for the case of the 25-mode POD basis on the reduced annular domain in the vicinity of the missile surface; we now extend it to all possible numbers of retained POD modes. For this, we first compact the information presented for any single aerodynamic coefficient in Fig. 5 into a single number, viz. the simple average of all the normalised errors. The resulting six numbers, one for each aerodynamic force and moment coefficient, are reported in Fig. 6 versus increasing number of POD modes retained, for both the choices of POD domains.

Overall, it is observed that the reconstruction error indeed reduces with increasing N_p; however, the decay is not as rapid as observed for the POD eigenvalues in Fig. 3(b). This is not surprising because, although POD guarantees the best reconstruction (with a given number of basis modes) of the flow field *over the entire domain*, it need not be efficient in reproducing the flow field on a boundary like the missile surface. However, it is very encouraging that POD still delivers an acceptable reconstruction of all the coefficients, with less than 2% average error for $N_p = 20$ and less than 1% average error for $N_p = 25$. Figure 6(a) also bears out the overall trends observed in Fig. 5 regarding the relative magnitudes of coefficient prediction errors.

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

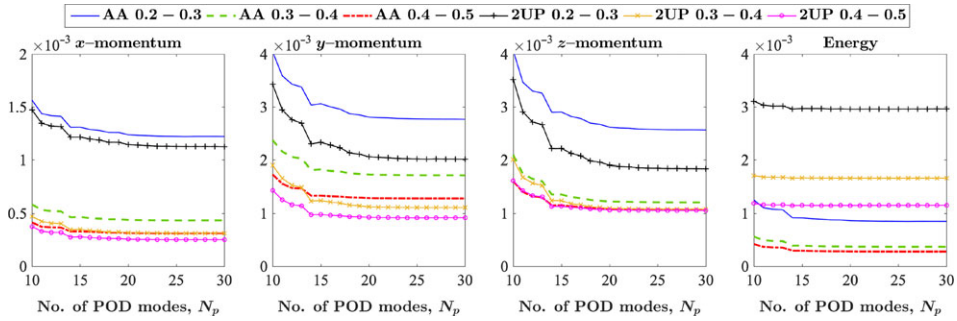

Figure 7. *Database-averaged POD-reconstructed residuals of momenta and energy equations. Legend indicates arithmetic average (AA) or 2nd-order upwind (2UP) interpolation used in data pre-processing, and alternative extents of Ω_R.*

We end this discussion on error analysis by noting that the results from the full-domain POD shown in Fig. 6(b) are very similar to the reduced-domain POD results of Fig. 6(a) being discussed above. This further corroborates the preservation of information in the cost-effective domain reduction process that was first noted in the context of their POD eigenspectra presented in Fig. 3(b). *Given that the significantly higher cost associated with the full-domain POD does not result in any appreciable benefit compared to the reduced-domain POD, we will implicitly use the latter in all the subsequent results presented, unless mentioned otherwise.*

5.1.3 Euler residuals computed from POD reconstruction of cases in learning database

We will subsequently use the Euler equation residuals defined in Equations (7) and (8) to determine the optimal ROM solution. Figure 7 presents the magnitudes of the residuals of the three momentum equations and the energy equation; recall that the continuity equation is satisfied to machine precision in the present context. Specifically, the $\mathcal{L}_1$-norm of the residuals are calculated from the flow fields of the learning database cases reconstructed with increasing number of POD modes N_p, and their average over the database are shown. The residuals asymptote uniformly with N_p.

This analysis affords an assessment of some of the options available in the POD-ROM approach. Firstly, the calculated residuals depend on the specific method used to interpolate the cell-centre FOM data to the faces, prior to the POD step. Figure 7 presents results from the simplest arithmetic average (AA) approach, as well as the second-order upwind (2UP) method used in the FOM solver [17]. We find that the former delivers significantly lower energy residuals, but somewhat higher momentum residuals. This result suggests that the much simpler arithmetic-average method may give acceptable predictions. All the POD results presented previously use the arithmetic average method in the pre-processing; essentially identical graphs (not shown) are obtained with the second-order upwind method also.

The second option evaluated here is the choice of the spatial domain on which the residual is calculated. The domain has already been reduced once prior to the POD (see Fig. 3(a) and its associated discussion); here we consider even smaller sub-domains of this. Specifically, we assess three annular regions that are successively farther removed from the missile surface. The annular regions are of $0.1D$ width, and they start from 0.2, 0.3 and $0.4D$ away from the missile surface, respectively. As expected, the Euler residuals decrease uniformly with increasing distance of the sub-domain of evaluation from the missile surface. Even though the nearest sub-domain results in two to three times higher residual values, it will be shown later to be the most suitable for the ROM.

The geometry of grid cells chosen for evaluating the residuals is crucial. In a 3-D unstructured mesh, some non-triangular faces of cells may be non-planar due to errors in meshing. Such faces do not have a unique normal direction, and the fluxes calculated on them by the FOM solver presumably depend on the particular method adopted to have a consistent normal. Due to the inherent redundancy in our ROM

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

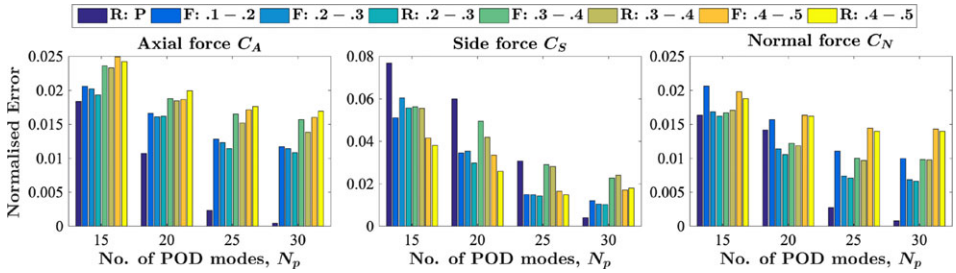

Figure 8. *Database-averaged cross-validation errors in ROM-prediction of force coefficients. In the legend, R and F refer to use of Ω_P vs. Ω for POD; choices of Ω_R are also indicated. The P indicates the result of projection, instead of ROM.*

optimisation problem, we chose to eliminate all cells with non-planar faces from Ω_R. This reduced the cell count from $\sim$70K to $\sim$3600 in the annular zone from 0.2 to 0.3D, for example. The ROM was degraded if this pruning step was skipped.

5.2 ROM results

5.2.1 Cross-validation of the ROM reusing the learning database

The POD is guaranteed to be the most efficient at reconstructing the flow field of a snapshot in the learning database. However, this is only in an average sense, and also in the sense of reconstruction accuracy evaluated over the domain of definition of the inner product. On the other hand, we wish to use the POD basis in the ROM for predicting the forces and moments on the missile surface, and that too for a case that is not included in the learning database. Before testing the POD-ROM approach on new cases, we can assess it by reusing the learning database itself in a cross-validation strategy. This is useful in ascertaining the optimal values of some of the parameters of the approach itself for the specific problem at hand, and incurs no extra FOM simulation cost.

Snapshots are available for N_s ($=32$) operating conditions in the learning database. We form N_s separate POD bases, each omitting one of the original snapshots. Then, each of these N_s POD basis sets is used in the POD-ROM approach to predict the aerodynamic coefficients of the respective omitted snapshot only. The full learning database of course has the truth values of these aerodynamic coefficients, so that the predictive effectiveness of the model can be assessed readily.

Figure 8 presents the results of this validation exercise. As in the results shown in Fig. 6, we assess the performance of the approach by averaging the fractional error across all the N_s predictions, each for a snapshot that was omitted in calculating the respective POD basis. This method allows us to evaluate three parameter choices: (a) the effect of using a reduced domain in the POD instead of the full FOM domain, (b) the appropriate number of POD modes to retain, and (c) the choice of the annular region around the missile on which to evaluate the Euler residual in the ROM. Each of these options have been partially assessed in Section 5.1, but without exercising the ROM.

Before describing the ROM prediction errors, we discuss the average errors incurred in estimation of the aerodynamic coefficients when simply 'projecting' a learning database snapshot onto the POD basis formed by leaving it out. This is denoted by R: P in Fig. 8. These results may be compared with their counterparts in Fig. 6(a), where the projection was onto the POD basis created by including the one being projected. Evidently, the errors in the present case are an order of magnitude higher than those found earlier. This reflects the difficulty in estimating surface flow conditions from POD-reconstruction targeting the overall flow domain.

The ROM prediction error decreases almost monotonically with increasing N_p, but saturates beyond $N_p \approx 25$. While the initial improvement is expected from the discussion of Fig. 6, the final saturation behaviour may not have been anticipated. However, there is an obvious difference in the two scenarios that explains this. Earlier we were looking at the vanishing of the error over the learning database when the POD basis spanned the same database. On the other hand, here we are assessing the effectiveness

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

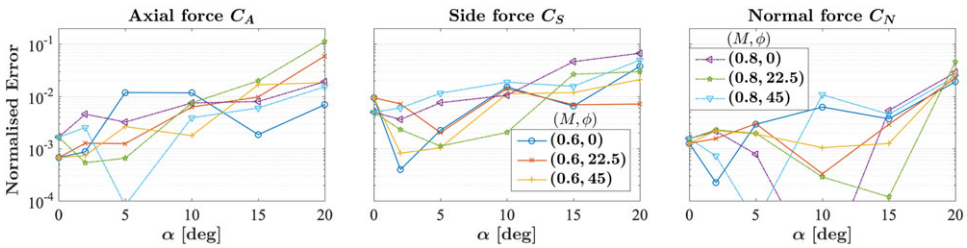

Figure 9. *Error in cross-validation of force coefficients with the 25 POD-mode R:* $0.2 - 0.3$ *ROM. The scheme of presentation follows Fig. 5.*

of the POD basis in reconstructing the aerodynamic coefficients of a case that is outside its span. Thus, the present observation is in line with the idea of over-fitting – there is an optimal number of degrees of freedom for the most robust fit of data.

There appears to be an optimal offset of the annular region on which the ROM residual is evaluated; overall, the most effective choice appears to be the region between $0.2D$ and $0.3D$ from the missile surface. This trend is not followed in the case of prediction of the side force coefficient using the smaller values of N_p. The probable reason for the overall trend was surmised in Section 5.1.1. Namely, Euler residuals evaluated on viscous-dominated regions close to the surface are a poor indicator of the actual accuracy of the solution; on the other hand, regions too far from the missile have diminishing correlation with the surface forces and moments. In fact, it is based on the above result that the POD itself was performed on the snapshots abstracted to the reduced domain extending between $0.2D$ and $2.0D$ off of the missile surface, as pictured in Fig. 3(a). Indeed, Fig. 8 demonstrates that the subsequent ROM delivers almost identical results as the case where the POD is performed on the full FOM domain. This equivalence is particularly advantageous since usage of the snapshots abstracted on the smaller domain drastically reduces the cost and time expense of the POD-ROM implementation. The choice of the outer extent (i.e. $2.0D$) of the reduced domain for POD was not optimised further. Since the grid coarsens rapidly away from the missile surface, this choice does not materially affect the computational efficiency of the POD-ROM.

It may appear that the imposition of the far-field boundary condition is impossible if the POD is performed on snapshots abstracted on a near-surface domain. This apparent paradox is resolved by recalling that the freestream boundary condition is actually evaluated indirectly with the strategy delineated in Section 4.3. Basically, we are using an inner product whose induced norm is positive semi-definite (i.e. defined over a part of the domain), so that the POD modes are notionally defined over the entire FOM domain that extends to the far-field boundary.

Figure 9 presents a detailed view of the aerodynamic coefficient prediction error with the most pertinent model, viz. using the first 25 POD modes computed on the reduced $0.2D - 2.0D$ domain, and evaluating the Euler residual in the ROM on the $0.2D - 0.3D$ annular domain. One finds fairly reliable prediction in most of the cases with fractional errors limited to 2%, but the errors do increase significantly at the highest angles of attack – viz. $\alpha = 20^\circ$, and to some extent at $\alpha = 15^\circ$. It is to be noted that cases with $\alpha = 20^\circ$ represent an extrapolation of the learning database. Also, cases with $\alpha \geq 15^\circ$ have massive flow separation that makes the aerodynamic coefficient vs. angle-of-attack curve nonlinear. Given these challenges, it is quite encouraging to note that even in these extreme conditions, the predictions are accurate to about 10%. Of course, a direct juxtaposition with the POD interpolation results in Fig. 5 shows these ROM results in poor light, but the comparison is not a fair one as the former approach did not have any predictive value.

The preceding presentations of aerodynamic coefficients' prediction errors were normalised by their respective maximum values observed in the database (as listed in Table 1); this provides a sense of the accuracy of the method in *absolute* terms. However, from a design perspective, it is more pertinent to

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

Table 2. *Computational expense and timing of optimisation ROM runs, averaged over all model evaluations; the variability is indicated in terms of the standard deviation*

No. of POD modes, N_p	20	25	30
Run time [seconds]	9.0 ± 1.6	15.3 ± 2.7	23.0 ± 4.0
No. of iterations	40.1 ± 7.1	50.8 ± 9.0	56.8 ± 10.0
No. of function evaluations	856.2 ± 151.4	1323.3 ± 233.8	1763.1 ± 310.9

Table 3. *Top half: parameter combinations of cases on which POD-ROM approach is validated. Bottom half: the respective normalized prediction errors (%) of the A/D coefficients with the 25 POD-mode R:* $0.2 - 0.3$ *model*

Case no.	1	2	3	4	5	6	7	8	9
α [°]	7.5	8	8	10	10	10	13	18	18
M	0.6	0.65	0.8	0.6	0.7	0.8	0.8	0.75	0.8
ϕ [°]	22.5	15	22.5	11.25	22.5	11.25	22.5	30	22.5
C_A	0.6	2.6	1.3	1.1	1.8	0.0	1.5	0.2	3.6
C_S	0.7	0.6	0.1	0.6	2.2	1.4	1.2	3.8	4.1
C_N	0.4	0.4	0.0	0.2	0.0	0.2	0.2	2.9	1.5
C_l	0.6	0.2	0.2	0.6	0.0	0.4	0.0	22.3	11.9
C_m	0.4	0.4	0.0	0.2	0.0	0.2	0.2	4.1	2.1
C_n	1.0	0.6	0.0	0.3	2.5	1.1	0.4	4.1	4.6

understand the prediction errors associated with the POD-ROM approach *relative* to their respective individual truth values (as found from the FOM). In the industry, it is typically deemed acceptable if wind-tunnel/flight data reveal up to 5% error in the FOM prediction of C_S, C_N, C_m and C_n, up to 10% in C_A, and up to 20% in C_l. Although not presented here, the relative errors corresponding to the data in Fig. 9 are well within these bounds. The only exceptions are: (a) cases where the coefficients themselves are very small – e.g. C_S and C_n at low angles of attack for $\phi \neq 45°$, (b) the rolling moment coefficient for which maximum relative errors approach 30%, and (c) extrapolative cases – e.g. when angle-of-attack is 20°. Even in the latter instances, the relative errors were around 6% in C_m and 11% in C_A. These results demonstrate that the proposed POD-ROM approach for missile data generation meets the industry standards for acceptability.

Table 2 gives an idea of the computational expense and time consumed for the ROM runs. The algorithm was implemented in Python, and the calculations were performed on a single processor in serial mode. The expense is primarily a function of the number of POD modes included in the optimisation, with the time taken being at most half a minute for $N_p = 30$. This run time for the ROM is almost negligible compared to an FOM calculation that is performed on a grid with approximately 10 million cells.

5.2.2 Validation of the ROM on new cases

The validity of the ROM can be tested on other unlearnt snapshots, as long as they are free of shocks. Table 3 presents the operating condition details of the validation cases employed in the present work. It will be observed that the cases differ in at least one freestream parameter from nearby points in the learning database matrix.

The result of the validation tests is also presented in Table 3 using the most appropriate ROM coming out of the cross-validation exercise in Section 5.2.1, viz. R: $0.2 - 0.3$. Flow separation is minimal for $\alpha \leq 13°$, and we find that the errors in the predicted aerodynamic coefficients are very reasonable, being less than 3% overall, and in fact less that 1% in most cases. Even in the last two cases of $\alpha = 18°$ where flow separation is severe, errors are generally less than 5%. The only outlier is the rolling moment in these

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

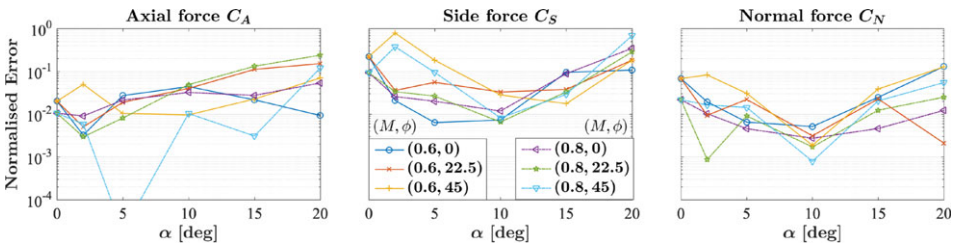

Figure 10. *Errors in force coefficients predicted using TPS-RBF interpolation with the first 25 POD modes defined on the reduced domain. The scheme of presentation follows Fig. 5.*

two cases, which display large errors. It was mentioned earlier that rolling moment is indeed difficult to predict, and it is also small compared to the other two moments. Overall the proposed POD-ROM approach is delivering encouraging estimates of all the aerodynamic force and moment coefficients.

5.3 Comparison of the ROM results with those from interpolation

A computationally inexpensive alternative to the optimisation ROM is the interpolation approach based on POD [4] discussed in Section 1; here we assess how it performs vis-à-vis the more expensive POD-ROM method. Recall that this method avoids a direct interpolation of the aerodynamic coefficients. Instead, the first N_p POD coefficients of the empirical database are interpolated to estimate those corresponding to a novel case, followed by evaluation of the aerodynamic coefficients therefrom using the approach described in Section 4.4. There are various interpolation options available, but instead of weighing their relative benefits, we follow earlier references [5] to employ thin-plate-spline (TPS) radial basis functions (RBF) using the `interpolate.Rbf` class of Scipy [26]. The learning database resides in a 3-D domain of operating parameters – viz. M, α and ϕ – that is scaled to the unit hypercube prior to interpolation.

Figure 10 presents the result of the cross-validation exercise (as described in Section 5.2.1) in terms of the fractional errors in interpolation-based prediction of the three aerodynamic force coefficients using the first 25 POD modes, evaluated on the database of 32 cases. The interpolation fares quite poorly in comparison with the corresponding ROM results shown in Fig. 9, with errors that are almost an order of magnitude greater. This attests to the inherent difficulty of the modelling problem undertaken in this work.

6.0 Conclusions

In this work, we demonstrate the application of a reduced-order model (ROM) approach to the prediction of aerodynamic force and moment coefficients of a subsonic missile. It leverages an existing RANS-based full-order model (FOM) database, where flow solutions have been calculated using ANSYS Fluent for certain missile operating conditions consisting of combinations of freestream parameters – viz. Mach number, angle-of-attack and roll angle. The ROM developed in this project learns from the FOM database, and is subsequently able to predict the aerodynamic coefficients at any other operating condition that falls within the range of conditions evaluated in the learning database. Limited excursions outside this range (i.e. extrapolations) are also possible.

The ROM approach makes use of proper orthogonal decomposition (POD) to extract the essential flow features from the learning database. It subsequently solves an optimisation problem, wherein a flow solution is sought that satisfies the boundary conditions corresponding to a new operating condition, all the while ensuring that the solution produces the minimum residual with respect to the governing equations. The main contributions of this work are:

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

- Application of model order reduction approach to an industrial problem involving three-dimensional steady subsonic aerodynamic flow over a missile at a range of relevant freestream conditions, including angle-of-attack up to 20°.
- Introduction of a reformulation of freestream boundary conditions in external aerodynamics problems as a linear equality constraint that can be imposed exactly.
- Demonstration of a physically motivated hyper-reduction approach for the problem.
- Introduction of a strategy for the approximate evaluation of aerodynamic coefficients from a reduced-order solution directly, without having to reconstruct the full solution.
- Obtaining aerodynamic coefficients from the ROM within half a minute on a single processor, whereas the FOM necessitates RANS simulation on $\sim 10^7$ cells – a matter of hours or days.
- Pursuit of an extensive cross-validation programme to understand the trade-offs involved at various stages of the model order reduction process.
- Demonstration of accuracy of all six predicted aerodynamic coefficients to within 5% of their respective maximum values (attained in the learning database) across nine different validation cases. The only exception is the difficult-to-predict rolling moment, which is off by up to 20% in one case, and by 10% in another. In terms of the *relative* errors, the accuracy of the POD-ROM method proves to be quite acceptable by industry standards.

The effectiveness of the POD-ROM approach is compared with a database interpolation method based on thin-plate spline radial basis functions, that is extant in the literature. For routine interpolations, this method is seen to result in around 10% errors, but the errors approach 100% in some interpolation cases, and definitely in cases involving extrapolation. This reinforces the superior performance of the proposed ROM approach.

ACKNOWLEDGEMENTS. AS gratefully acknowledges funding from Defence Research and Development Laboratory (DRDL) of India under grant DRDL/24/42P/17/0453/42932/CMM-II.

References

[1] Yano, M. Model reduction in computational aerodynamics, In P. Benner, S. Grivet-Talocia, A. Quarteroni, G. Rozza, W. Schilders, and L.M. Silveira (eds.), *Model Order Reduction. Volume 3: Applications.* De Gruyter, 2021.

[2] Fossati, M. and Habashi, W.G. Multiparameter analysis of aero-icing problems using proper orthogonal decomposition and multidimensional interpolation, *AIAA J.*, 2013, **51**, (4), pp. 946–960.

[3] Lorente, L.S., Vega, J.M. and Velazquez, A. Generation of aerodynamics databases using high-order singular value decomposition, *J. Aircr.*, 2008, **45**, (5), pp. 1779–1788.

[4] Bui-Thanh, T., Damodaran, M. and Willcox, K.E. Proper orthogonal decomposition extensions for parametric applications in compressible aerodynamics, In *21st Applied Aerodynamics Conference, AIAA Paper* 4213, 2003.

[5] Zimmermann, R. and Görtz, S. Improved extrapolation of steady turbulent aerodynamics using a non-linear POD-based reduced order model, *Aeronaut. J.*, 2012, **116**, (1184), 1079–1100.

[6] Franz, T., Zimmermann, R., Görtz, S. and Karcher, N. Interpolation-based reduced-order modelling for steady transonic flows via manifold learning, *Int. J. Comput. Fluid Dynam.*, 2014, **28**, (3–4), pp. 106–121.

[7] LeGresley, P.A. and Alonso, J.J. Investigation of non-linear projection for pod based reduced order models for aerodynamics, In 39th Aerospace Sciences Meeting, *AIAA Paper 926*, 2001.

[8] Alonso, D., Vega, J.M. and Velazquez, A. Reduced-order model for viscous aerodynamic flow past an airfoil, *AIAA J.*, 2010, **48**, (9), pp. 1946–1958.

[9] Zimmermann, R. and Görtz, S. Non-linear reduced order models for steady aerodynamics, *Proc. Comput. Sci.*, 2010, **1**, (1), pp. 165–174.

[10] Alonso, D., Vega, J.M., VelÁzquez, A. and de Pablo, V. Reduced-order modeling of three-dimensional external aerodynamic flows, *J. Aerosp. Eng.*, 2012, **25**, (4), pp. 588–599.

[11] Vendl, A. and Faßbender, H. Projection-based model order reduction for steady aerodynamics, In *Computational Flight Testing*, pp. 151–166. Springer, 2013.

[12] Washabaugh, K.M., Zahr, M.J. and Farhat, C. On the use of discrete nonlinear reduced-order models for the prediction of steady-state flows past parametrically deformed complex geometries, In 54th AIAA Aerospace Sciences Meeting, *AIAA Paper 1814*, 2019.

[13] Lumley, J.L. The structure of inhomogeneous turbulent flows. In A.M. Yaglom and V.I. Tatarsky (eds.), *Atm. Turb. and Radio Wave Prop.*, pp. 166–178. Nauka, Moscow, 1967.

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press

[14] Sirovich, L. Turbulence and the dynamics of coherent structures, Parts I-III, *Q. Appl. Math.*, 1987, XLV, (3), 561–590.
[15] LeGresley, P.A. and Alonso, J.J. Dynamic domain decomposition and error correction for reduced order models, In *41st Aerospace Sciences Meeting*, *AIAA Paper* 250, 2003.
[16] Nair, N.J. and Balajewicz, M. Transported snapshot model order reduction approach for parametric, steady-state fluid flows containing parameter-dependent shocks, *Int. J. Numer. Methods Eng.*, 2019, **117**, (12), pp. 1234–1262.
[17] *ANSYS Fluent 15 User's Guide*. ANSYS, Inc., Lebanon, NH 03766, USA, 2015.
[18] Holmes, P., Lumley, J.L., Berkooz, G. and Rowley, C.W. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge University Press, 2012.
[19] Alonso, D., Velazquez, A. and Vega, J.M. Robust reduced order modeling of heat transfer in a back step flow, *Int. J. Heat Mass Transf.*, 2009, **52**, (5), pp. 1149–1157.
[20] Vendl, A., Fassbender, H., Görtz, S., Zimmermann, R. and Mifsud, M. Model order reduction for steady aerodynamics of high-lift configurations, *CEAS Aeronaut. J.*, 2014, **5**, (4), 487–500.
[21] Washabaugh, K.M. *A Scalable Model Order Reduction Framework for Steady Aerodynamic Design Applications*. PhD thesis, Stanford University, 2016.
[22] Ryckelynck, D. A priori hyperreduction method: An adaptive approach, *J. Comput. Phys.*, 2005, **202**, (1), pp. 346–366.
[23] Astrid, P., Weiland, S., Willcox, K. and Backx, T. Missing point estimation in models described by proper orthogonal decomposition, *IEEE Trans. Automat. Cont.*, 2008, **53**, (10), pp. 2237–2251.
[24] Chaturantabut, S. and Sorensen, D.C. Nonlinear model reduction via discrete empirical interpolation, *SIAM J. Sci. Comput.*, 2010, **32**, (5), 2737–2764.
[25] Carlberg, K., Farhat, C., Cortial, J. and Amsallem, D. The GNAT method for nonlinear model reduction: Effective implementation and application to computational fluid dynamics and turbulent flows, *J. Comput. Phys.*, 2013, **242**, pp. 623–647.
[26] Jones, E., Oliphant, T., Peterson, P., *et al.* Scipy: Open source scientific tools for Python, 2001. http://www.scipy.org/

Cite this article: Sinha A., Kumar R. and Umakant J. (2022). Reduced-order model for efficient generation of a subsonic missile's aerodynamic database. *The Aeronautical Journal*, **126**, 1546–1567. https://doi.org/10.1017/aer.2022.4

https://doi.org/10.1017/aer.2022.4 Published online by Cambridge University Press