AIAA SciTech Forum
23-27 January 2023, National Harbor, MD & Online
AIAA SCITECH 2023 Forum
10.2514/6.2023-1004

Domain-Decomposed Reduced-Order Modelling of Steady Aerodynamics for 2D Store Separation Analysis

Navdeep Pandey*, Rohit Tembhare†, Nisha Singh‡ and Aniruddha Sinha§
Dept. of Aerospace Engineering, Indian Institute of Technology Bombay, Mumbai, 400076, INDIA

Multi-body aerodynamic analysis is crucial whenever there is a store that separates from its parent aircraft. Certification of a store for use with an aircraft demands a meticulous analysis of the separation trajectory across a range of operating parameters (freestream conditions). The present work demonstrates an efficient albeit approximate technique that employs a reduced-order model (ROM) based on proper orthogonal decomposition (POD) for predicting the flow field around a store-aircraft dyad. The main innovation is a domain-decomposition approach which allows POD-based ROM to be used for the majority of the flow domain, keeping small regions that must be addressed by full-order simulations. Steady Reynolds-Averaged Navier Stokes (RANS) is used to generate the learning database for a particular set of operating conditions. The prediction of the lift and drag forces and pitching moment coefficients for a new set of operating conditions is performed using the proposed approach. Encouraging results are obtained in the validation that is pursued for simplicity on a two-dimensional problem; the approach applies to three-dimensional problems too.

Nomenclature

M_∞	=	freestream Mach number
α	=	angle of attack
β_{store}	=	pitch angle of store relative to airfoil's chord
X_{store}	=	x-position of the store's nose w.r.t. airfoil's LE, normalized by airfoil chord length
Y_{store}	=	y-position of the store's nose w.r.t. airfoil's LE, normalized by airfoil chord length
$\boldsymbol{\mu}$	=	operating parameter vector
$\mathbf{q}$	=	flow field variable vector (comprising of velocity vector, density, pressure, etc.)
$\boldsymbol{\phi}_n$	=	n^{th} POD mode
η_n	=	n^{th} POD coefficient

I. Introduction

Whenever a new aircraft and/or its stores (missiles, drop tanks, etc.) are developed, or an existing aircraft/store combination undergoes some modifications, the dyad has to undergo an extensive investigation to certify the safe-separation flight envelope. During the analysis, trajectories followed by the store under various operating conditions are predicted to ensure that it does not strike back at the aircraft itself. There are mainly three conventional approaches to conduct such investigations – flight testing, experiments in a wind tunnel, and simulation using computational fluid dynamics (CFD). All of them pose some challenges related to economic and practical viability; moreover, they are all extremely time-consuming. For example, flight testing is expensive in addition to being prone to serious accidents; both wind tunnel experiments and CFD are expensive – economically as well as time-wise.

Here, a method is proposed based on empirical reduced-order modelling (ROM) integrated with domain-decomposition to estimate the store's trajectory. The empiricism in the ROM implies the existence of a prior 'learning' database, which typically comes from a small set of preliminary CFD calculations that sparsely sample the parameter space. Such computations are referred to as full-order model (FOM) results in this work, to distinguish them

*Graduate student
†Graduate student
‡Graduate student
§Associate Professor; AIAA Member; Corresponding author: sinha.a@iitb.ac.in

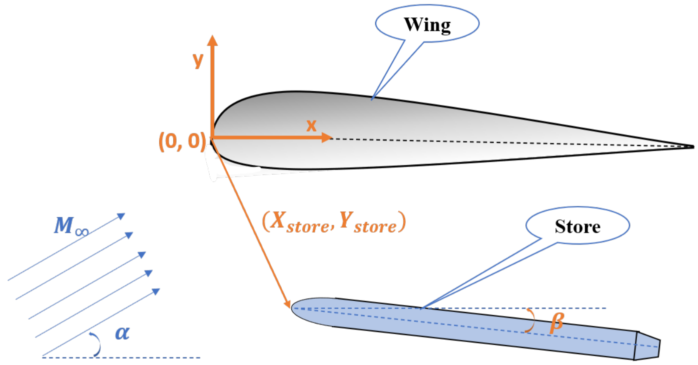


Fig. 1 Setup of the two-dimensional 2-body store-separation problem.

from the proposed ROM technique. Subsequently, the ROM predicts the flow and/or derived quantities (like lift/drag on external bodies) at unsampled parameter values. ROM is a well-known tool to predict the characteristics of the flow field very efficiently. The ability of ROMs to predict flows with a small turnaround time requiring minimal computational resources makes them suitable for applications where rapid design decisions have to be made – e.g., multi-disciplinary analysis and optimization. Evidently, the prediction of store-separation trajectory is a candidate for the employment of ROMs.

The overall objective of this research is to efficiently predict the safe flight envelope by computing the trajectory of a store released from its parent aircraft under various operating conditions (e.g., freestream Mach number M_∞, angle-of-attack α of the aircraft, etc.). The setup comprises two bodies (see fig. 1), where body 2 (store) is under the influence of body 1 (aircraft) (and vice versa). For simplicity of the validation process, we apply the proposed method on a two-dimensional problem in this work. Moreover, we attempt to predict the forces and moments acting on the store for a given position of it, relative to the aircraft flying at a given (M_∞, α) condition. The actual calculation of the separated store's trajectory may be pursued as a straightforward augmentation, involving an additional 3-degree-of-freedom rigid body solver coupled with the method validated here.

The proposed ROM uses proper orthogonal decomposition (POD) to identify the underlying reduced-order topology of the flow. Most of such spatial modal analyses, including POD, necessitate that all the empirical flow data supplied are for the same geometry (and mesh). In the case of store trajectory prediction, the store changes its relative position with respect to the aircraft continuously after its release. The corresponding empirical database thus has arbitrary alteration of the flow geometry. Consequently, the usual POD-ROM approach for single-body aerodynamics [1] cannot be employed for the full-flow domain here. Considering this limitation, the proposed approach decomposes the full-domain into multiple sub-domains such that POD-ROM can be employed for the maximal portion of the flow domain; the remaining sub-domain that involves changing geometry/mesh is solved using the full-order model (FOM). The flow has to be matched at the interfaces of the sub-domains through a few iterations of the ROM(s) and FOM calculations. This is expected to yield a drastic decrease in the overall computational effort and time consumed, when compared to the prevailing FOM analysis applied to the full domain. We demonstrate here that very reasonable accuracy is achieved in the proposed domain-decomposed ROM (DDROM), by predicting the forces and moment coefficients for unsampled test cases.

II. Background

We start with a brief background of existing methods for trajectory prediction in Section II.A; subsequently, we discuss the previous applications of ROMs to similar problems in Section II.B.

A. Existing methods for trajectory prediction

The very first attempts to predict the trajectory of the stores were performed using flight testing. The store was repeatedly dropped from the aircraft flying at gradually increasing speeds, until the store was observed to come back closer to the aircraft after its release instead of separating monotonically. This method was based on trial-error, and in some cases, this led to the store hitting the aircraft itself. In the early 1960's, wind tunnel model testing for trajectory prediction was developed using the Captive Trajectory System [2, 3]. Since wind tunnel tests typically use small-scale

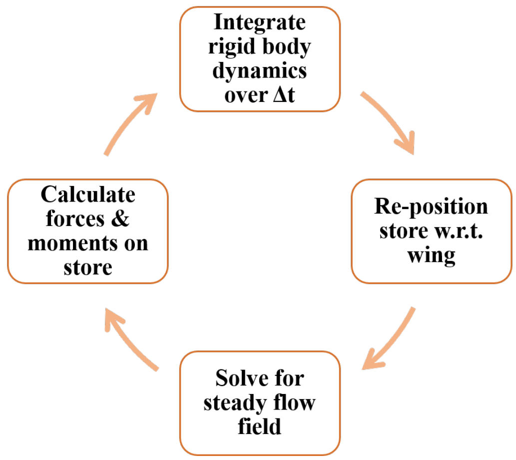


Fig. 2 Flow chart representing the iterative steps involved in store trajectory prediction with the FOM approach, using time integration with a step size of Δt.

models, large deviations were often observed in comparison with the more reliable reference flight test data.

After the advancement of CFD capabilities to accurately represent the flow around complex aircraft-store geometries, there was a shift towards CFD analysis from wind tunnel model testing [4]. In the 1980's, the Influence Function Method became popular [5], which used computer simulation for store separation studies. In this method, wind tunnel data was used to determine the influence coefficients of an aircraft in the vicinity of a reference store, and this database was used to determine the forces and moments for any other test store placed at different positions relative to the aircraft. The store trajectory was determined using a 6 degrees of freedom (DOF) rigid body dynamics solver in an iterative manner. With the consistent efforts and investments from US Air Force and Navy, an experimental database was created in the 1990's, based on generic wing-store configurations [6], which further helped researchers to validate CFD approaches developed for trajectory prediction [7–11].

Almost all the CFD approaches proposed by researchers provided results in good agreement with the experimental database. This paved the way for the adoption of inviscid, quasi-steady analysis for trajectory prediction (see fig. 2); the method provided sufficient accuracy in cases of stores dropped from steady flights. Basically, the flow around the aircraft-store configuration is solved assuming it to be steady; the resulting aerodynamic forces and moments on the store are supplied to a rigid-body motion solver to calculate its changed position after a small time step, the steady flow is solved again for this new two-body configuration, and so on. In spite of its success, researchers were dubious regarding its accuracy in some complicated cases – e.g., when multiple stores are released from the aircraft, or when the aircraft makes sudden manoeuvres, etc. In late 2010's researchers performed unsteady Euler calculations for store separations [12, 13] and reconfirmed that quasi-steady analysis provides sufficiently accurate results for the store dropped from steady flight and there is no added advantage in performing unsteady, inviscid analysis.

B. Application of ROM to similar problems

Applications of reduced-order modelling in flow problems can be broadly divided into three categories – steady (parameter-varying) ROM, unsteady (time-varying) ROM for a single parameter set, and a combination of the two. In problems dealing with time-varying flow characteristics, unsteady ROM is employed to predict the flow behaviour at different time instants. Steady ROM finds its applications in the problems where steady (or time-averaged) flow features (for a particular parameter in a set) are sufficient for the engineering purpose at hand. In the present work, we are interested in steady ROMs for the reason described in section II.A.

The application of parameter-varying steady ROM to single-body aerodynamics can be found in several works [1, 14–18]. As mentioned previously, the first step in the development of a steady ROM is the generation of the 'learning database' comprising of 'snapshots' of the steady flow field for a sufficiently rich yet sparsely-sampled set of operating parameters (e.g., Mach number, angle of attack, side slip angle, etc.). This is typically obtained by solving the FOM – i.e., doing Euler or RANS (Reynolds-averaged Navier Stokes) CFD simulations; rarely do we get this empirical data from experiments. This is the one-time cost of the ROM, albeit a major one. The next step involves the reduction of the order of the problem by identifying the underlying simplicity in the flow features present in the empirical database.

Mathematically, we compute a minimal set of basis functions or modes of the data such that maximal information is captured. The most common tool used for this is proper orthogonal decomposition (POD) [19, 20]. The last step is the prediction of flow field behaviour for a new set of parameters. This step can be performed in two ways – (i) flow field prediction based on interpolation (or extrapolation) of the basis functions [e.g. 21], or (ii) computation of flow behaviour as a linear combination of the basis modes using optimized coefficients that satisfy the steady governing equations as well as boundary conditions [17, 18], called steady ROM. The steady ROM approach is more robust but computationally expensive compared to the simple interpolation approach. In the present work, we are interested in the application of steady ROM in the more complicated 2-body problem.

The present approach primarily builds on top of two of our previous works – an initial attempt at solving the present 2D 2-body problem by decomposing the flow domain into two sub-domains [22], and the development of the aerodynamic database of a missile (i.e., a single body) using (single-domain) POD-ROM [1]. The former effort established the basic philosophy of the approach. However, it failed to achieve any substantial saving in computational time for reasons that will be evident later. The latter paper, although concerning a simpler problem, presented some valuable improvements in the POD-ROM approach itself that are of relevance here.

III. Theory and approach

A. POD

Let $\mathbf{q}(\mathbf{x};\boldsymbol{\mu})$ represent the vector field of relevant flow variables in a steady problem, with $\mathbf{x} := (x, y)$ in 2D Cartesian coordinates, and $\boldsymbol{\mu}$ being the vector of operating parameters (e.g., freestream Mach number M_∞ and angle of attack α of the airfoil in a 2D problem). In case of a 2D problem governed by Euler's equations, the vector field $\mathbf{q}$ may be $[\rho, \rho u, \rho v, p]^T$, where ρ is the density, u & v are the x- & y-components of velocity respectively, and p is the pressure.

We assume that the flow fields available in the learning database may be subjected to an efficient linear modal decomposition such that

$$\mathbf{q}(\mathbf{x};\boldsymbol{\mu}) \approx \overline{\mathbf{q}}(\mathbf{x}) + \sum_{n=1}^{N} \eta_n(\boldsymbol{\mu})\boldsymbol{\phi}_n(\mathbf{x}). \tag{1}$$

Here, $\overline{\mathbf{q}}$ represents the base flow variable vector field, which is typically obtained by averaging across all the snapshots of the learning database. The deviation of each solution (snapshot) from the base flow is assumed to be well approximated by the linear combination of N spatial 'modes' (or basis flow fields) $\{\boldsymbol{\phi}_n(\mathbf{x})\}_{n=1}^N$ weighted by the coefficients $\{\eta_n(\boldsymbol{\mu})\}_{n=1}^N$. For an efficient order reduction, N is typically much smaller than the number of grid points needed to represent the flow domain for a converged CFD simulation. For convenience, we will write the vector of weight coefficients as $\boldsymbol{\eta}(\boldsymbol{\mu}) := (\eta_1(\boldsymbol{\mu}), \eta_2(\boldsymbol{\mu}), \ldots, \eta_N(\boldsymbol{\mu}))^T$.

In this work, the above basis flow fields are obtained using the very well-established approach of POD [19, 20]. For brevity, we omit the details here; further discussion of the variant of POD that is most relevant for the present work can be found in Sinha et al. [1].

B. ROM

The (approximate) prediction of the flow field for a new parameter set outside the learning database, say $\boldsymbol{\mu}_0$, is pursued using the ROM. Rather than simply interpolating the parameter space, a more robust and accurate approach is to invoke the underlying governing equations of the flow, or a simplification thereof. It is based on the reduced order modal decomposition of the flow field, i.e., the POD. Equation (1) reveals that this comes down to determining the new set of (POD) coefficients $\boldsymbol{\eta}(\boldsymbol{\mu}_0)$. The basic methodology of POD-based ROM for single-body analysis was proposed by LeGresley and Alonso [14], which was further refined over the subsequent years [1, 17, 18].

Let the vector of governing *unsteady* conservation equations be represented as

$$\frac{d}{dt}(\mathscr{C}(\mathbf{q})) = \mathscr{R}(\mathbf{q}), \quad \forall \mathbf{x} \in \Omega. \tag{2}$$

Here, $\mathscr{C}$ is the operator that maps the chosen set of flow variables $\mathbf{q}$ to the conserved flow variables, $\mathscr{R}(\mathbf{q})$ is a shorthand notation for the terms other than the local time derivative, and Ω represents the relevant flow domain. These equations are supplemented by the vector of boundary conditions, formally represented as

$$\mathscr{B}(\mathbf{q}) = 0, \quad \forall \mathbf{x} \in \delta, \tag{3}$$

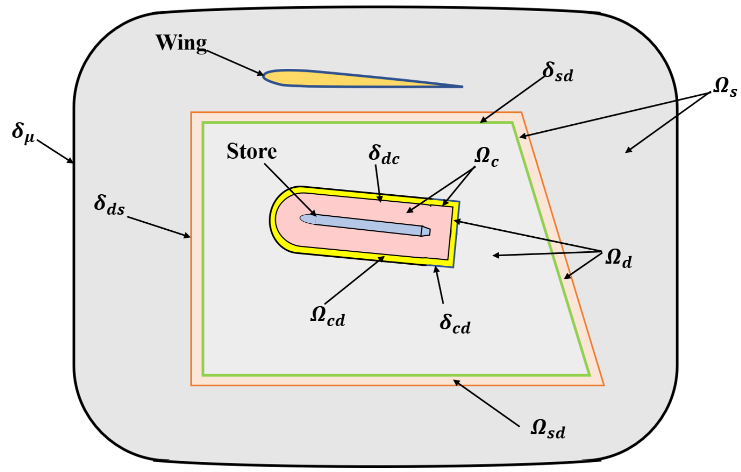


Fig. 3 Three domain-decomposed ROM approach for the two-body problem of fig. 1.

where δ denotes the boundary of the flow domain. Since the *steady* solution is desired for $\boldsymbol{\mu}_0$, we would ideally have $\mathbf{q}(\mathbf{x}, \boldsymbol{\mu_0})$ such that $\mathscr{R}(\mathbf{q}(\mathbf{x}, \boldsymbol{\mu_0})) = 0$ along with $\mathscr{B}(\mathbf{q}(\mathbf{x}, \boldsymbol{\mu_0})) = 0$. Indeed, the FOM attempts to satisfy these constraints at a multitude of grid points/cells over the flow domain and boundary.

In the POD-based ROM, we substitute the approximate expansion of eqn. (1) in the above governing equations. Since the base flow field and the POD modes are known from the learning database, the residual and boundary conditions are now approximated as

$$\mathscr{R}(\mathbf{q}(\mathbf{x}, \boldsymbol{\mu}_0)) \approx \tilde{\mathscr{R}}(\mathbf{x}; \boldsymbol{\eta}(\boldsymbol{\mu}_0)), \qquad \mathscr{B}(\mathbf{q}(\mathbf{x}, \boldsymbol{\mu}_0)) \approx \tilde{\mathscr{B}}(\mathbf{x}; \boldsymbol{\eta}(\boldsymbol{\mu}_0)). \tag{4}$$

Moreover, due to the preceding approximation, one cannot expect the above vector fields to vanish exactly on their respective domains. Instead, we recast the given problem as the following optimization problem:

$$\min_{\boldsymbol{\eta}} J(\boldsymbol{\eta}(\boldsymbol{\mu}_0)), \quad J(\boldsymbol{\eta}(\boldsymbol{\mu}_0)) := \left(\left\|\tilde{\mathscr{R}}(\cdot; \boldsymbol{\eta}(\boldsymbol{\mu}_0))\right\|_p\right)_\Omega \quad \text{subject to } \left(\left\|\tilde{\mathscr{B}}(\cdot; \boldsymbol{\eta}(\boldsymbol{\mu}_0))\right\|_p\right)_\delta < \epsilon. \tag{5}$$

Here, $||\cdot||_p$ denotes the $\mathscr{L}^p$ norm of a vector field, and ϵ denotes a suitable tolerance specified for approximately satisfying the boundary conditions. In the present work, we have used the $\mathscr{L}^1$ norm of the residue, and it is computed over the entire domain Ω. Note that for a reasonably small number of retained POD modes, the above represents a very small optimization problem, that constitutes a significant saving over the corresponding FOM.

C. Domain decomposition

Reconsider the two-body problem of fig. 1 with the store being allowed to take up any arbitrary position (i.e., location and orientation) relative to the aircraft. Evidently, the problem geometry as well as the mesh must change as the store is traversing its trajectory. POD cannot represent the flow field for such cases involving changes in the geometry and/or mesh. We propose the following domain decomposition strategy so as to deploy POD-based ROM for solving the flow problem over the largest possible portion of the flow domain, along with a FOM solution restricted to the remaining small sub-domain. This is an extension of the approach presented by Sinha and Garg [22].

As shown in fig. 3, the full domain is decomposed into three sub-domains as follows:

- *Capsule*, Ω_c – consists of the region immediately surrounding the store, which may comprise a grid that does not change as the store moves. Evidently, its extent is limited by the configuration of nearest approach between the store and aircraft.
- *Dropbox*, Ω_d – consists of the entire region enveloping the capsule where the capsule is expected to go in the course of the store-separation trajectory, while still remaining under the influence of the aircraft. Basically, this is the only sub-domain of the flow where the mesh needs to change in the problem.
- *Staticzone*, Ω_s – consists of the remaining flow domain that remains unchanged in the problem. It envelopes the aircraft as well as the dropbox, and stretches all the way to the far-field boundary.

We also provide for two narrow overlap regions – one between the capsule and the dropbox (denoted Ω_{cd}), and the other between the dropbox and the staticzone (denoted Ω_{sd}). Keeping in mind the typical store-separation trajectories, the

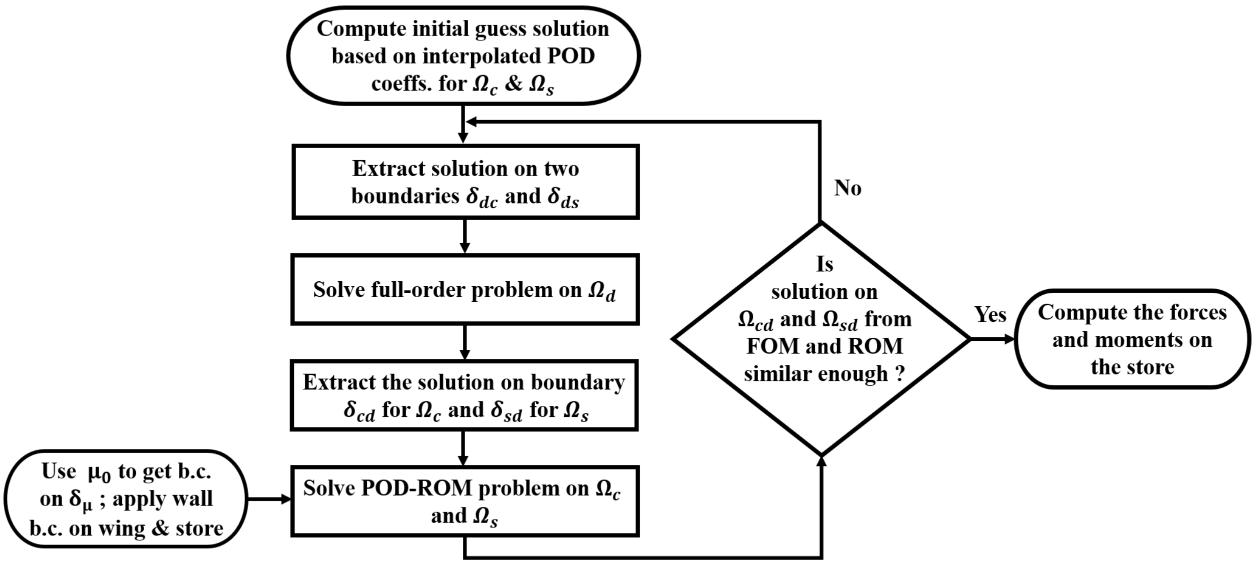


Fig. 4 Flow chart of steps involved in three domain-decomposed ROM for 2-body flow analysis.

dropbox is made trapezoidal in shape. Evidently, the aircraft will cease to have an influence on the store's motion once the latter is far enough away from it, thereby naturally providing a bottom boundary for the dropbox. The trajectory calculation may be safely discontinued when the capsule touches this boundary.

The capsule and staticzone are sub-domains where the mesh can remain same across all possible positions of the store, as well as all choices of the operating parameters (viz. freestream conditions). Thus, we pursue POD-based ROM in these two sub-domains. On the other hand, the dropbox is the only sub-domain where the mesh must change across various possible positions of the store. Hence, POD-ROM is inapplicable here, and we have to apply FOM calculations instead. The flow solution must be matched at the two interfaces. To enable the matching of the lower-order derivatives of the solution also, the interfaces are designed as narrow overlap regions.

The staticzone comprises the majority of the flow domain. On the other hand, the capsule is a very small sub-domain. However, when RANS analysis is pursued, then the mesh must be refined close to the solid boundaries of both the aircraft and the store. Thus, the majority of the mesh cells will be accounted for by the staticzone and capsule sub-domains. By solving the flow here using the very efficient POD-ROM, we should obtain significant computational savings. On the other hand, the remaining dropbox region can typically have a relatively coarser mesh, so that the FOM calculations are not very expensive thereat either.

The actual DDROM solution procedure is iterative, as outlined in fig. 4. A start is made by calculating the POD coefficients for the two POD-ROM domains by interpolating the learning database in the parameter space. This yields the initial guess of flow solutions in Ω_s and Ω_c. From these, we extract the flow variables on the two boundaries of the dropbox δ_{ds} and δ_{dc} (which are internal to the respective ROM domains Ω_s and Ω_c due to the existence of the overlap regions). These flow information now serve as the sole (Dirichlet) boundary conditions for the FOM to be solved on the dropbox sub-domain Ω_d. Once the latter solution is in hand, we extract the flow variables on the interfacing boundaries of the two ROM domains – viz. δ_{sd} for Ω_s and δ_{cd} for Ω_c, both of which are internal to Ω_d due to the overlap regions. Now, the ROM is solved on Ω_s using the above information as the interfacing boundary condition on δ_{sd}, along with the far-field and aircraft wall boundary conditions. Similarly, the ROM is solved on Ω_c using the information on δ_{cd} along with the store wall boundary condition. At this stage, we have the first iteration of the flow solution on the two overlap regions – viz. Ω_{sd} and Ω_{cd} – from both the ROMs as well as the FOM. If these are too different, we can iterate the above procedure, starting again by extracting the flow variables on the boundaries of the dropbox. On the other hand, if these are close enough, then we can cease iterating, and compose the overall solution. In particular, the desired force and moment coefficients on the store may be obtained from the final POD-ROM solution in the capsule sub-domain.

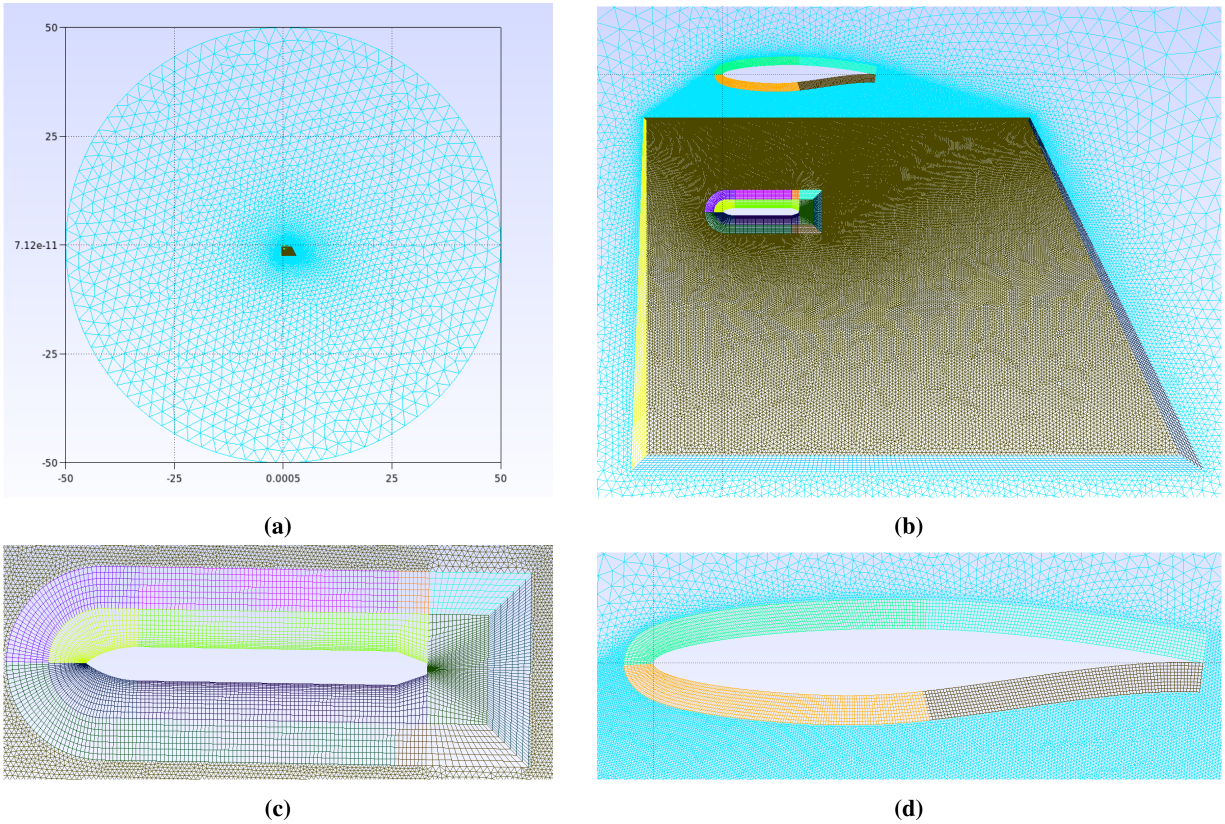


Fig. 5 Mesh topology for airfoil-store in 2D. (a) Overall domain. (b) Zoomed view of dropbox sub-domain. (c) Zoomed view of store with body-fitted structured mesh as capsule sub-domain. (d) Zoomed view of airfoil with body-fitted structured mesh.

IV. Results and discussion

We evaluate the proposed domain-decomposed POD-based ROM methodology on a two-dimensional problem involving a store-like body in the influence region of a wing-like body (airfoil in 2D), as depicted in figs. 1 and 3.

A. Geometry and mesh

The airfoil is the standard RAE-2822 geometry. The 'store' is a slender rectangle with a nose and a tail. Its overall length is 50% of the airfoil chord, and its thickness d is 10% of its length. The nose is a circularly-blunted tangent ogive with tip radius of $0.05d$ and length of $1.5d$. The boat-tail has a half-wedge angle of 20°, and length equal to d. The far-field boundary is chosen to be a circle centered on the leading edge of the airfoil, with radius equal to 50 times the airfoil's chord.

The mesh is generated with Gmsh v4.8.4 [23]. Structured grids are best suited for controlling the mesh resolution near the wall, but for the two-body problem we can not have such a mesh throughout the flow domain. To circumvent this problem, a body-fitted structured mesh has been generated around the airfoil as well as the store body, and an unstructured mesh spans the remaining region till the far-field boundary. To allow the proposed domain-decomposed approach, a structured mesh has also been generated for the overlap regions near both boundaries of the dropbox sub-domain. Figure 5 depicts the mesh topology for the two-body problem where quadrilateral cells are used in the structured mesh and triangular cells are used in the unstructured mesh. The Gmsh meshing is driven by a Python code to automatically ensure that identical meshes (with the same numbering of nodes and cells) exist in Ω_c, Ω_s, Ω_{cd} and Ω_{sd} regions, irrespective of the position of the store relative to the airfoil.

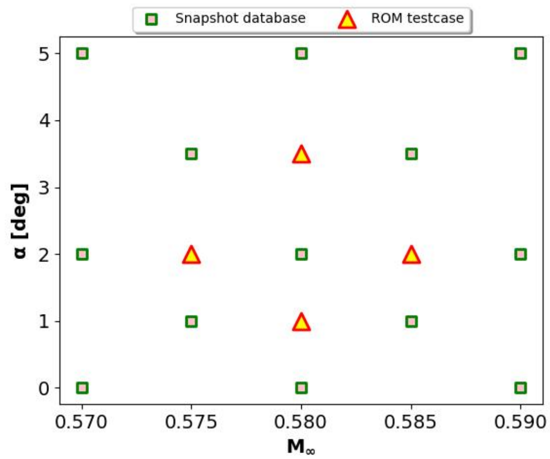


Fig. 6 Operating parameters (i.e., M_∞, and α) combinations employed for snapshot database and ROM test cases.

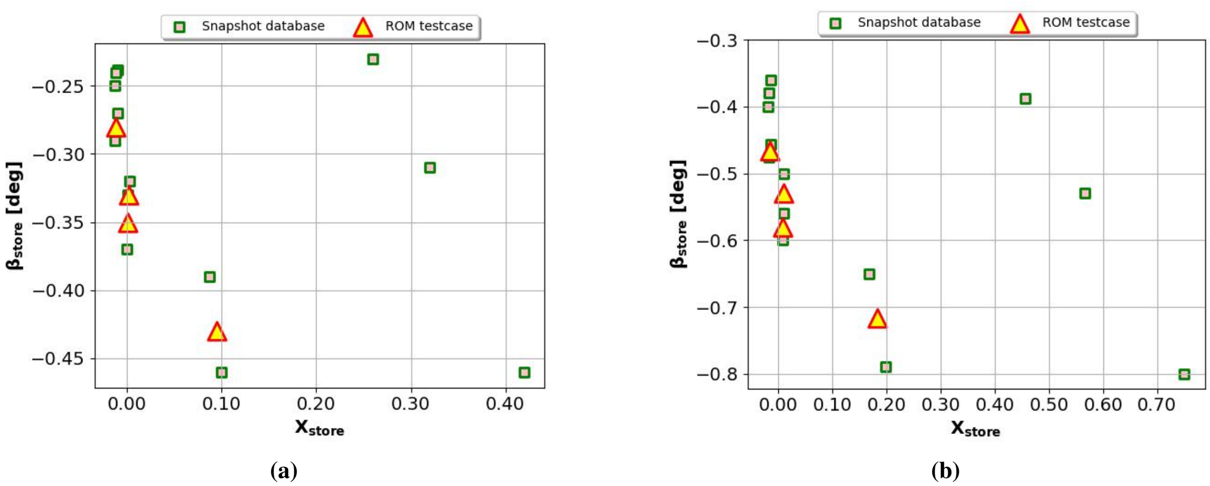


Fig. 7 X_{store} and β_{store} coordinates of the trajectories corresponding to the $M_\infty - \alpha$ combinations shown in fig. 6 for the Y_{store} coordinate equalling (a) -1.4 & (b) -2.0.

B. Learning database generation and POD

If the store is released from the aircraft in the same configuration always, then each store trajectory is completely specified by the particular freestream condition experienced by the aircraft. For 2D problems, these are the freestream Mach number M_∞ and the angle of attack of the airfoil α. The learning database comprises of the store trajectories (and their corresponding sets of quasi-steady flow solutions) for a number of $M_\infty - \alpha$ combinations. Indeed, in fig. 6, the $M_\infty - \alpha$ combinations chosen for the store trajectories of the learning database are shown with green squares. For now, they cover a small range of Mach numbers (0.57 to 0.59), but a larger range of angles of attack (0 deg to 5 deg). For later reference, the test (or validation) cases are also indicated in the same figure, this time with red triangles.

A point on the trajectory of the store is completely specified by its three degrees of freedom (DOF). These are the two position coordinates of the nose of the store relative to the leading edge of the airfoil (X_{store}, Y_{store}), and the pitch angle β_{store} of the store relative to the airfoil's chord line (see fig. 1). Just as all lengths in the problem, the position coordinates are also implicitly normalized by the chord of the airfoil.

The computation of trajectories based on a RANS solver is very expensive. Since the primary objective of the present work is the validation of the proposed DDROM method, trajectories are computed with the lower-fidelity Euler-based solver in SU2 v7.1.1 [24, 25], in communication with a 3-DOF rigid body dynamics solver written in Python. Figure 7 presents the X_{store} and β_{store} of the store as it is crossing the $Y_{store} = -1.4$ and -2.0 levels in all the

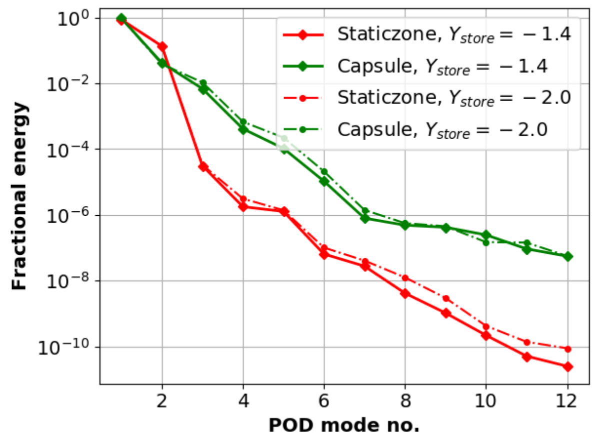


Fig. 8 Vector POD eigenspectra for the staticzone and capsule sub-domains in terms of the fractional energy content – viz. $\lambda_i/\sum_j \lambda_j$, where λ_i is i^{th} POD eigenvalue.

trajectories – both for the learning and the test databases.

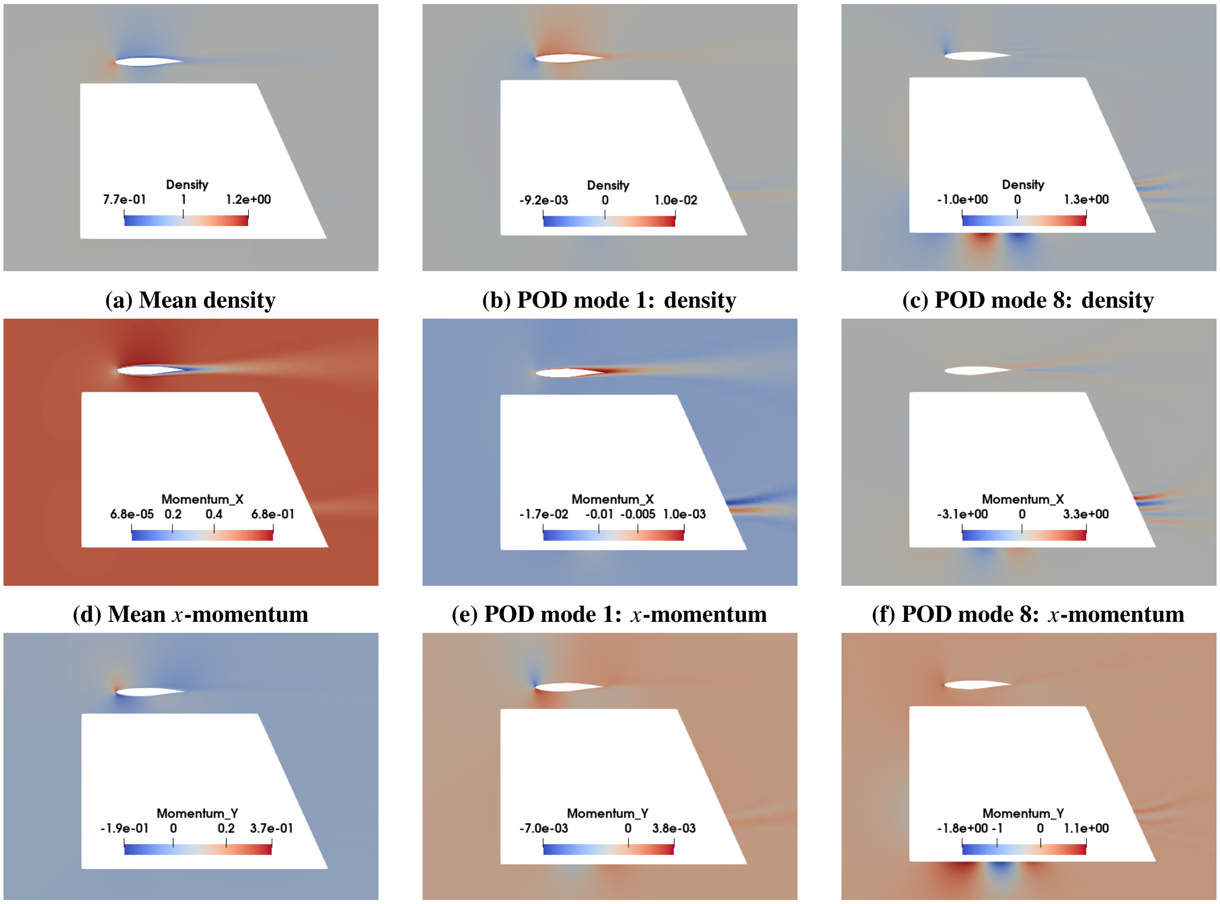
For the purpose of validating the DDROM approach, we will do as follows. For the case of store trajectories passing through the $Y_{store} = -1.4$ level, we will determine the extended parameter vector $\boldsymbol{\mu} = (M_\infty, \alpha, X_{store}, \beta_{store})$ for all the 13 learning database trajectories (calculated using the Euler solver). Next, these 13 snapshots will be simulated again, but this time using a RANS solver, again in SU2 v7.1.1 [24, 25]. These flow fields will become the actual snapshot database for POD in the staticzone and capsule sub-domains. Then the corresponding POD-based DDROM will be exercised to predict the forces and moments on the store for all the four $\boldsymbol{\mu}$'s corresponding to the intersection of the test-case trajectories with the $Y_{store} = -1.4$ level. These results will be compared versus the 'truth' solution obtained from RANS simulations of these cases. The entire exercise will be repeated for the $Y_{store} = -2.0$ level. Such an exercise will reveal the viability of the proposed DDROM technique for predicting an entire store trajectory for a $M_\infty - \alpha$ combination that is outside the learning database of trajectories, even with RANS. The two particular Y_{store} levels are chosen to give a large enough range of $\boldsymbol{\mu}$'s, while the store remains well within the influence zone of the airfoil.

The actual RANS solutions in SU2 are automated using Python scripts. The Spalart-Almaras turbulence model is chosen for its proven reliability in external aerodynamics problems. The far-field boundary has the relevant freestream condition; the walls of the airfoil and store have the no-slip constraint applied. A grid convergence study is pursued to ensure the reliability of the data, although its results are not presented here.

The conserved energy variable in the solution is converted to pressure; the remaining three conserved flow variables (viz. density, and the two Cartesian components of momentum) are retained as they are. This flow solution vector field – viz. $\mathbf{q} = [\rho, \rho u, \rho v, p]^T$ – is extracted on the staticzone and capsule sub-domains in all the snapshots. Density is normalized by the freestream density ρ_∞, velocity by the freestream sound speed c_∞, and pressure by $\rho_\infty c_\infty^2$. Snapshot-based vector POD is performed on these two sub-domains separately. One may remark that the turbulence variables are neglected in extracting the solution vector field; this will be rationalized subsequently.

The fractional eigenspectra of POD for the staticzone and capsule sub-domains are presented in fig. 8 for both the cases of Y_{store} locations. There is a rapid (4 orders of magnitude) decline in the captured energy content (or relative importance) between POD modes 2 and 3 in the staticzone sub-domain. This is because the first few POD modes capture flow features of the overall flow domain, while the subsequent POD modes capture localized flow characteristics near the airfoil and around the dropbox. On the contrary, the capsule sub-domain is itself in the form of a thin annulus around the store, and thus its eigenspectrum displays a more gradual decay. For the staticzone, POD mode 8 accounts for about 10^{-7} less 'energy' vis-'a-vis POD mode 1; for the capsule, it is POD mode 10 that has about this relation with its POD mode 1. This justifies the retention of the first 8 and 10 POD modes for the staticzone and capsule sub-domains, respectively, for the subsequent ROMs. There are only minor differences between the fractional eigenspectra at two Y_{store} locations. Apparently, the $Y_{store} = -2.0$ eigenspectra have a slightly gentler decay compared to the $Y_{store} = -1.4$ case. The reason for this can be the greater range of X_{store} and β_{store} values in case of $Y_{store} = -2.0$ compared to $Y_{store} = -1.4$ (see fig. 7).

The contour plots of mean flow and POD modes 1 and 8 for the staticzone subdomain is presented in fig. 9.



(a) Mean density **(b) POD mode 1: density** **(c) POD mode 8: density**

(d) Mean x-momentum **(e) POD mode 1: x-momentum** **(f) POD mode 8: x-momentum**

(g) Mean y-momentum **(h) POD mode 1: y-momentum** **(i) POD mode 8: y-momentum**

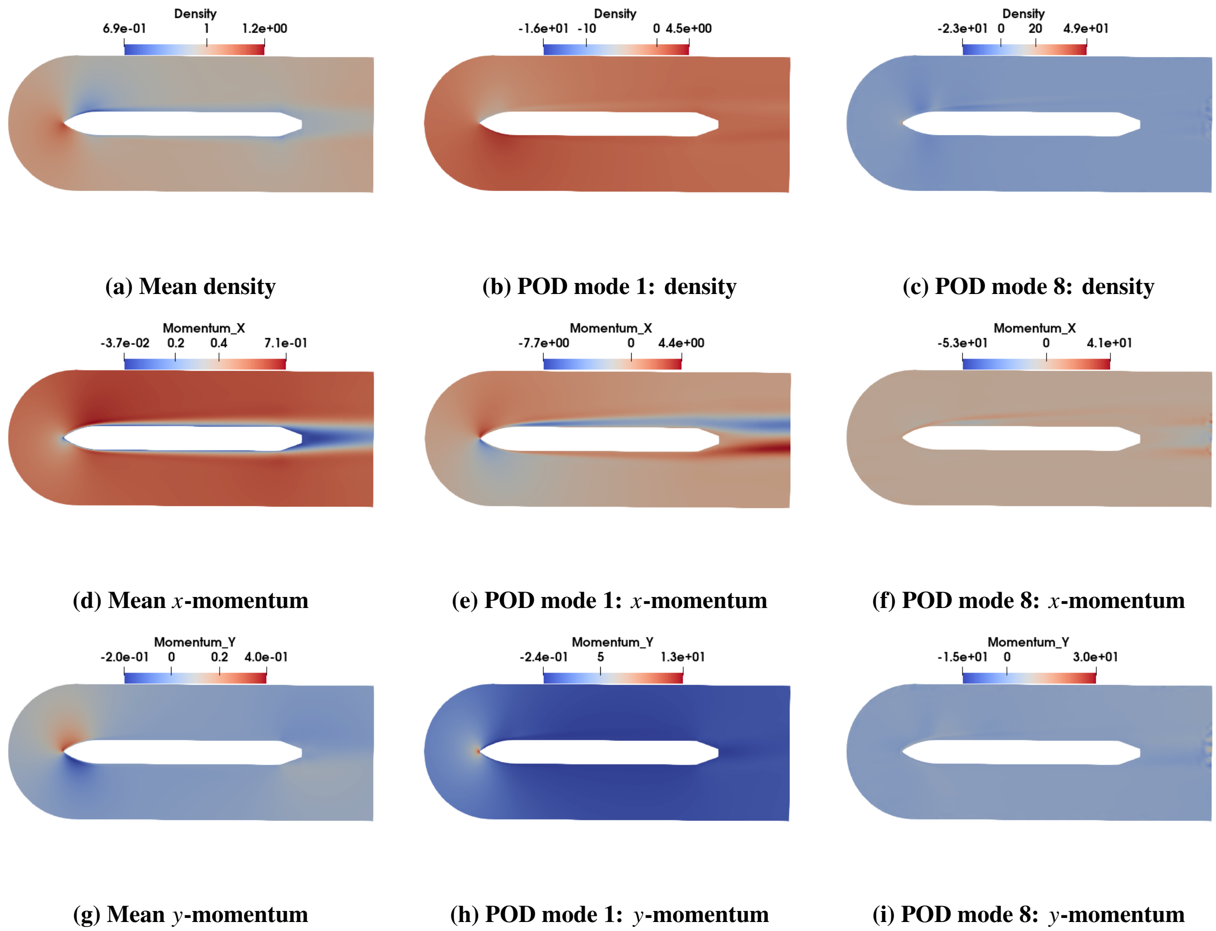
Fig. 9 Density (top row), x-momentum (middle row), and y-momentum (bottom row) components of the mean field (left column), first POD mode (middle column), and eighth POD mode (right column) for the staticzone sub-domain.

Compressibility effects are weak at these Mach numbers so that pressure fields closely resemble density fields; hence, only the latter are shown. The idea behind presenting contour plots of these POD modes is to highlight the difference in the flow features captured by lower and higher POD modes. The first POD mode captures most of the important global characteristics of the flow field, whereas the higher-order modes capture the localized features, e.g., wakes of the store corresponding to its different positions (Y_{store} and β_{store}) in the learning database. The freestream pressure and temperature conditions were kept constant across all the snapshots. Hence, the density component of the POD modes obtained based on the fluctuation density flow field is not showing any variation (magnitude close to zero).

These POD modes also substantiate the specific choice of local (in parameter space) POD bases. If one were to compose POD modes from all snapshots along the learning trajectories, then the footprints of the store's wake in the staticzone would have been diffuse. This wake footprint is what allows the proper transfer of information between the staticzone and dropbox sub-domains in the DDROM. It is the Y_{store} variable that changes most rapidly as the store goes through its trajectory. Keeping Y_{store} fixed ensures that the local reduced basis composed from all learning trajectories has adequate information about the store wake structures that are possible for various choices (within reason) of the other four parameters – viz. M_∞, α, X_{store} and β_{store}.

The corresponding depiction of the POD modes of the capsule sub-domain appears in fig. 10. Again, the lower-order POD modes appear to have greater global support compared to the higher-order ones. In the case of the capsule sub-domain, POD mode 1 already displays some fine-scale structure, which is due to the narrower region of the capsule sub-domain itself. However, the structures in mode 8 are even finer.

To reiterate, based on the preceding discussion, the first 8 POD modes are chosen for the staticzone's ROM; on the



(a) Mean density **(b) POD mode 1: density** **(c) POD mode 8: density**

(d) Mean x-momentum **(e) POD mode 1: x-momentum** **(f) POD mode 8: x-momentum**

(g) Mean y-momentum **(h) POD mode 1: y-momentum** **(i) POD mode 8: y-momentum**

Fig. 10 Mean field and POD modes of the capsule sub-domain, with presentation scheme following from fig. 9.

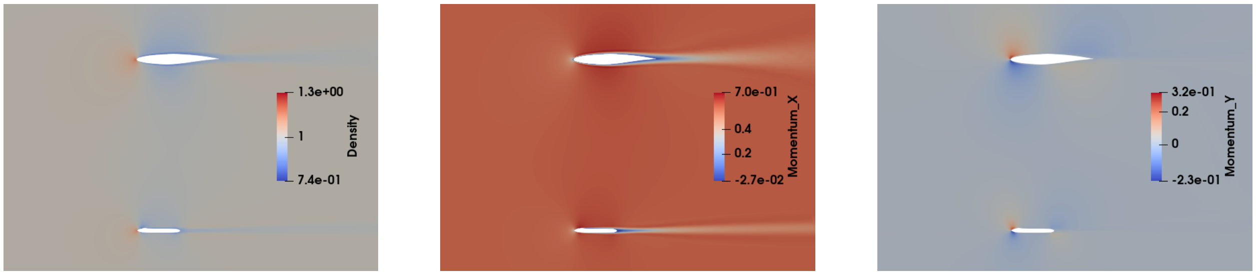
other hand, the first 10 POD modes are retained in the capsule's ROM.

C. Validation results

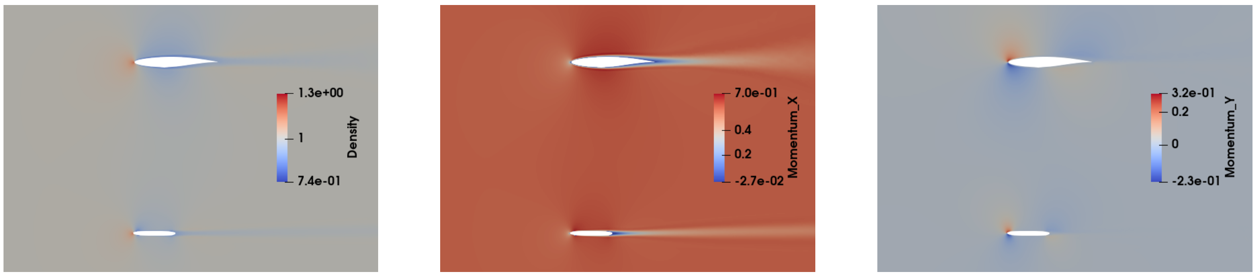
It will be recalled from section IV.B that, although all trajectory computations are performed with inviscid equations, all snapshots – be they for learning or validation – are based on RANS. Still, as discussed by many previous researchers [1, 16–18], the residual that is minimized in the ROM (see eqns. (2–5)) is for the Euler equations. Moreover, since flow gradients are expected to be small in the dropbox region, the FOM that is solved for this sub-domain as part of DDROM, are also the Euler equations; they are solved using SU2. It is evident that the Dirichlet boundary conditions on the inner and outer boundaries of the dropbox are not constant, but are different at each point in general. For this, a custom boundary condition is implemented in the SU2 framework. The actual optimization problem involved in the POD-ROM applied to the staticzone and capsule sub-domains is implemented using the sequential least-squares quadratic programming (SLSQP) method available in the `scipy` library of Python.

The scheme for validating the proposed DDROM approach has been laid out in section IV.B. Two separate models are created from the empirical data of store trajectories sectioned at the two Y_{store} parameter values of -1.4 and -2.0. They are then exercised by keeping this parameter constant but varying all the other four parameters (viz. M_∞, α, X_{store} and β_{store}) as pertinent for four other test trajectories (see figs. 6 and 7). It is noted that all test parameter values remain within the convex hulls of their respective values in the learning database, so that there is no extrapolation involved.

For illustrative purposes, fig. 11 presents the flow fields reconstructed by DDROM for the first test trajectory (i.e., $M_\infty = 0.58$, $\alpha = 1.0\,$deg) at the $Y_{store} = -2.0$ location. They demonstrate very reasonable agreement with the corresponding 'truth' solutions obtained directly from the FOM (i.e., steady RANS with Spalart-Almaras turbulence



(a) Density from DDROM **(b) x-momentum from DDROM** **(c) y-momentum from DDROM**



(d) True density **(e) True x-momentum** **(f) True y-momentum**

Fig. 11 Flow fields predicted by the DDROM (top row) compared with the corresponding 'truth' solutions in case of the $M_\infty = 0.58$, $\alpha = 1.0\,\text{deg}$ **trajectory intersected at** $Y_{store} = -2.0$**.**

model) applied to the overall domain.

For the purposes of store-separation trajectory calculations, the estimation of the store's aerodynamic coefficients is of course much more important than the prediction of the overall flow fields. In view of this, table 1 demonstrates that the coefficients of lift, drag and pitching moment (about the nose) of the store are approximated very well by the DDROM approach developed here. The maximum error in c_l and c_d are within 3%; errors in pitching moment, which is more difficult to predict accurately, are still limited to 5.5%.

V. Conclusion

This paper presents a reduced order model (ROM) based on proper orthogonal decomposition (POD) in conjunction with a domain-decomposition approach – termed DDROM – for two-body aerodynamics problems. The overall motivation is to reduce the computational expense of store-separation trajectories prediction, as required in the certification of the safe-separation flight envelope. The proposed approach replaces Euler or RANS simulations on the overall flow domain with a much more lightweight optimization problem for the major portion of the domain, resorting to Euler calculations for relatively few grid cells that change when the store moves relative to the aircraft. The approach is validated on a two-dimensional two-body problem, where RANS calculations are pursued in the full domain for the generation of the underlying empirical database. Results demonstrate the appreciable capability of the approach in terms of accuracy in the prediction of the lift, drag and moment coefficients of the store, as well as the overall flow field.

The work can be taken forward along several avenues. In the DDROM, the full order model (FOM) involves Euler calculations in the 'dropbox' sub-domain where the grid must change depending on the position of the store relative the aircraft. Although this approach is found to be quite accurate already, we may achieve even better results by implementing RANS simulations for this FOM. Secondly, the actual store-separation problem may be addressed with the DDROM, whereas here we have only validated the method at certain points of the trajectory in terms of the accuracy of estimation of force and moments on the store. Finally, the methodology needs to be extended to the more complex (in degree but not in kind) three-dimensional aircraft-store separation problem.

References

[1] Sinha, A., Kumar, R., and Umakant, J., "Reduced-order model for efficient generation of a subsonic missile's aerodynamic database," *The Aeronautical Journal*, Vol. 126, 2022, pp. 1546–1567.

Table 1 Store's coefficients of lift, drag and pitching moment (about nose) predicted by DDROM compared with the 'truth' for four trajectories at two Y_{store} levels each.

Trajectory no.		**I**		**II**		**III**		**IV**	
M_∞		0.58		0.575		0.585		0.58	
α		1.0		2.0		2.0		3.5	
Y_{store}		–1.4	–2.0	–1.4	–2.0	–1.4	–2.0	–1.4	–2.0
	Truth ×10	0.49	0.68	1.03	1.23	1.05	1.27	1.84	2.10
c_l	**DDROM ×10**	0.50	0.68	1.05	1.23	1.08	1.27	1.83	2.10
	% error	2.0	0.0	1.9	0.0	2.9	0.0	0.5	0.0
	Truth ×100	1.12	1.08	1.20	1.16	1.22	1.18	1.44	1.43
c_d	**DDROM ×100**	1.15	1.10	1.23	1.16	1.24	1.21	1.44	1.43
	% error	2.7	1.9	2.5	0.0	1.6	2.5	0.0	0.0
	Truth ×100	0.55	0.73	1.11	1.31	1.14	1.36	1.95	2.20
$c_{m,nose}$	**DDROM ×100**	0.52	0.72	1.07	1.27	1.09	1.29	1.90	2.08
	% error	5.5	1.4	3.6	3.1	4.4	5.1	2.6	5.5

[2] Bamber, M., "Two methods of obtaining aircraft trajectories from wind tunnel investigations," *AERO Report*, Vol. 970, 1960.

[3] Christopher, J., and Carleton, W., "Captive-trajectory store-separation system of the AEDC-PWT 4-foot transonic tunnel," Tech. rep., Arnold Engineering Development Center, Arnold AFB, TN, USA, 1968.

[4] Rogers, R., "A comparison between the Nielson and Woodward programs in predicting flow fields and store loads," *Naval Weapons Center TM*, Vol. 2854, 1976, pp. 14–4.

[5] Meyer, R., Cenko, A., and Yaros, S., "An influence function method for predicting store aerodynamic characteristics during weapon separation," Tech. rep., Grumman Aerospace Corp, Bethpage, NY, 1981.

[6] Heim, E., "CFD wing/pylon/finned store mutual interference wind tunnel experiment," Tech. rep., Arnold Engineering Development Center, Arnold AFS, TN, USA, 1991.

[7] Meakin, R., "Computations of the unsteady flow about a generic wing/pylon/finned-store configuration," *38th AIAA Atmospheric Flight Mechanics Conference, paper 4568*, 1992.

[8] Newman, J., III, and Baysal, O., "Transonic solutions of a wing/pylon/finned store using hybrid domain decomposition," *38th AIAA Atmospheric Flight Mechanics Conference, paper 4571*, 1992.

[9] Parikh, P., Pirzadeh, S., and Frink, N., "Unstructured grid solutions to a wing/pylon/store configuration using VGRID3D/USM3D," *38th AIAA Atmospheric Flight Mechanics Conference, paper 4572*, 1992.

[10] Madson, M., Moyer, S., and Cenko, A., "TranAir computations of the flow about a generic wing/pylon/finned-store configuration," *32nd AIAA Aerospace Sciences Meeting and Exhibit, paper 155*, 1994.

[11] Madson, M., and Talbot, M., "F-16/generic store carriage load predictions at transonic Mach numbers using TranAir," *14th AIAA Applied Aerodynamics Conference, paper 2454*, 1996.

[12] Berglind, T., and Tysell, L., "Numerical investigation of the impact of maneuver on store separation trajectories," *28th AIAA Applied Aerodynamics Conference, paper 4241*, 2010.

[13] Berglind, T., and Tysell, L., "Time-accurate CFD approach to numerical simulation of store separation trajectory prediction," *29th AIAA Applied Aerodynamics Conference, paper 3958*, 2011.

[14] LeGresley, P., and Alonso, J. J., "Investigation of non-linear projection for pod based reduced order models for aerodynamics," *39th AIAA Aerospace Sciences Meeting and Exhibit, paper 926*, 2001.

[15] Alonso, D., Velazquez, A., and Vega, J., "A method to generate computationally efficient reduced order models," *Computer Methods in Applied Mechanics and Engineering*, Vol. 198, No. 33-36, 2009, pp. 2683–2691.

[16] Alonso, D., Vega, J., and Velazquez, A., "Reduced-order model for viscous aerodynamic flow past an airfoil," *AIAA journal*, Vol. 48, No. 9, 2010, pp. 1946–1958.

[17] Zimmermann, R., and Görtz, S., "Non-linear reduced order models for steady aerodynamics," *Procedia Computer Science*, Vol. 1, 2010, pp. 165–174.

[18] Alonso, D., Vega, J., Velazquez, A., and Pablo, V., "Reduced-order modeling of three-dimensional external aerodynamic flows," *Journal of Aerospace Engineering*, Vol. 25, 2012, pp. 588–599.

[19] Lumley, J. L., "The structure of inhomogeneous turbulent flows," *Atm. Turb. and Radio Wave Prop.*, edited by A. M. Yaglom and V. I. Tatarsky, Nauka, Moscow, 1967, pp. 166–178.

[20] Holmes, P., Lumley, J. L., Berkooz, G., and Rowley, C. W., *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*, Cambridge University Press, 2012.

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

[22] Sinha, A., and Garg, S., "Reduced-order modeling of steady aerodynamics for 2D store separation analysis," *2018 AIAA Applied Aerodynamics Conference, paper 3168*, 2018.

[23] Geuzaine, C., and Remacle, J.-F., "Gmsh: A 3-D finite element mesh generator with built-in pre-and post-processing facilities," *International Journal for Numerical Methods in Engineering*, Vol. 79, No. 11, 2009, pp. 1309–1331.

[24] Palacios, F., Colonno, M. R., Aranake, A. C., Campos, A., Copeland, S. R., Economon, T. D., Lonkar, A. K., Lukaczyk, T. W., Taylor, T. W. R., and Alonso, J. J., "Stanford University Unstructured (SU2): An open-source integrated computational environment for multi-physics simulation and design," *51st AIAA Aerospace Sciences Meeting, paper 0287*, 2013.

[25] Palacios, F., Economon, T. D., Aranake, A. C., Copeland, S. R., Lonkar, A. K., Lukaczyk, T. W., Manosalvas, D. E., Naik, K. R., Padrón, A. S., Tracey, B., Variyar, A., and Alonso, J. J., "Stanford University Unstructured (SU2): Open-source analysis and design technology for turbulent flows," *52nd AIAA Aerospace Sciences Meeting, paper 0243*, 2014.