Sādhanā (2024)49:17
https://doi.org/10.1007/s12046-023-02365-z

Multi domain decomposed reduced order model for store trajectory prediction

NAVDEEP PANDEY and ANIRUDDHA SINHA*

Department of Aerospace Engineering, IIT Bombay, Mumbai 400076, India
e-mail: 204010004@iitb.ac.in; as@aero.iitb.ac.in

MS received 20 April 2023; revised 12 September 2023; accepted 18 October 2023

Abstract. Two-body aerodynamic analysis is crucial whenever there is a store that separates from its parent body, i.e., the aircraft. Any time that a new aircraft is developed or an existing one undergoes some modification with its associated store, it has to undergo a meticulous analysis to predict the path of the separated store across a range of operating parameters (freestream conditions). The present work demonstrates an efficient albeit approximate semi-empirical technique that employs a reduced-order model based on proper orthogonal decomposition in conjunction with a multi-domain-decomposition approach for predicting the flow field around a store-aircraft dyad. Encouraging preliminary results are obtained in the verification that is pursued on a two-dimensional problem for simplicity. The approach can be readily extended to three-dimensional problems as well.

Keywords. MDDROM; POD; store-separation trajectory.

1. Introduction

The aircraft-store separation analysis is crucial, whenever a new aircraft-store duo is developed or an existing one undergoes some design changes. In this analysis, trajectories followed by the store after its release from the parent body (i.e., the aircraft) are computed under different operating conditions (i.e., freestream parameters) to predict the safe separation flight envelope. Although initial attempts employed expensive and risky flight tests, and subsequent efforts relied on wind tunnel model testing using captive trajectory system, almost all current store trajectory predictions are pursued using computational fluid dynamics (CFD).

The CFD-based approach to store-trajectory simulation is summarized in figure 1. One starts with the initial configuration where the store is on the verge of separating from the aircraft. Invariably, a quasi-steady approach is employed, wherein CFD simulations of the flow field for any time instant assume that the flow is steady. Once the instantaneous forces and moments are obtained on the store from such a calculation, they are supplied to a 6 degrees of freedom (DOF) rigid body dynamics solver to determine the position of the store at the next time step. Steady CFD calculations are again conducted for this new aircraft-store configuration, and the simulation proceeds iteratively in this manner. The calculations are ended once the store is out of the influence zone of the aircraft.

Even with the exponential development of computational power and memory, the quasi-steady CFD calculations are very resource intensive and take time. Hence, there is an opportunity for developing a method that can predict the approximate trajectory of the store within limits of practical applicability with minimal computational cost and time. The reduced-order model (ROM) approach discussed herein is one such empirical technique. ROMs are well known for their ability to predict flow fields efficiently with a small turnaround time [1–6], which makes them suitable for applications where rapid design decisions have to be made – e.g., multi-disciplinary analysis and optimization. In the context of ROMs, the well-resolved CFD approach is called full-order model (FOM); the latter is taken as the 'truth' solution against which the performance of the ROM is evaluated.

The overall objective of this work is to establish and verify a ROM to efficiently predict a store's trajectory under different operating conditions of the parent aircraft – viz. its Mach number M_∞, angle-of-attack α, etc. For simplicity, we have considered a two-dimensional (2D) problem to demonstrate the proposed method. The problem setup shown in figure 2 consists of two bodies – viz. a wing (i.e., an airfoil in 2D) and another body that represents the store whose aerodynamics are influenced by the wing. We attempt to predict the forces and moments acting on the store when it is positioned relative to the aircraft flying at a given (M_∞, α) condition. The actual computation of trajectory can be performed by integrating a 3 DOF rigid body solver with the present setup.

*For correspondence
Published online: 08 January 2024

In the proposed ROM approach, we first parameterize the flow field using a minimal basis derived from proper orthogonal decomposition (POD) [7, 8]. This leverages the observed fact that apparently high-dimensional flow fields can often be approximated very well using a much lower dimensional embedding. The reduced-order POD basis is identified empirically, which means that it is based on a 'learning' database generated by a FOM. Subsequently, the coefficients of the basis modes towards the solution for a new case are determined in an optimization step that attempts to minimize the residual of the governing equations while satisfying the boundary conditions. The latter approach is the same as in any CFD, with the sole difference being the severely curtailed DOF to optimize due to the reduced POD basis being employed.

POD requires that the learning database, as well as the new cases to be predicted, be for the same geometry and mesh. This is impossible in our store trajectory prediction problem since the store continuously changes its position and orientation relative to the aircraft. Hence, the usual POD-ROM approaches for single-body aerodynamics [1–6] cannot be employed for the entire flow domain. To circumvent this limitation, Ref. [9] proposed a POD-based domain-decomposition reduced-order model (DDROM) approach, wherein the overall flow domain was decomposed into one sub-domain that enveloped the aircraft and extended to the far-field boundary but had a 'hole' or 'dropbox' in it located below the aircraft, and the other sub-domain that accounted for the dropbox containing the moving store. The former 'staticzone' sub-domain was the same across the entire learning database so that POD-ROM could be applied. The resulting solution was iteratively matched with FOM calculations on the dropbox. Since the majority of the mesh cells were in the staticzone, the use of POD-ROM promised efficiency gains.

The present work is an extension of the DDROM approach, wherein further efficiencies are realized by futher dividing the dropbox sub-domain itself into two sub-domains, and applying POD-ROM to one of them again. It is termed multi-domain-decomposed ROM (MDDROM). We demonstrate here that the MDDROM delivers reasonable accuracy vis-à-vis the FOM 'truth'.

2. Background

We start with a brief background of existing methods for trajectory prediction in Section 2.1; subsequently, we discuss the previous applications of ROMs to similar problems in Section 2.2.

2.1 *Existing methods for trajectory prediction*

The very first attempts to predict the trajectory of the stores were performed using flight testing. The store was

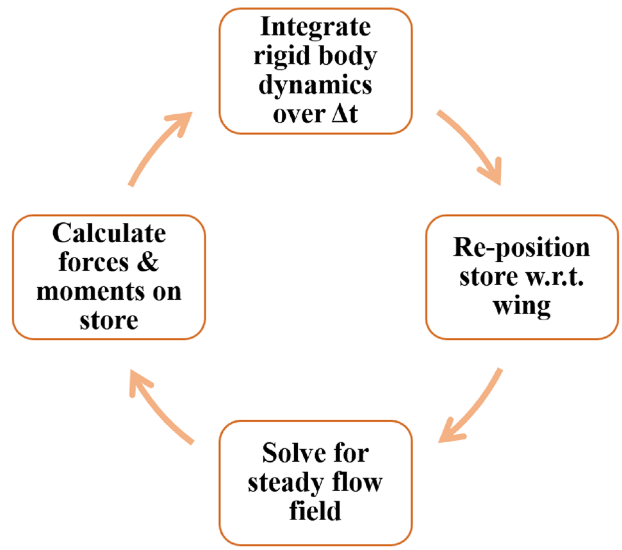


Figure 1. Flowchart showing the iterative steps involved in store trajectory prediction using quasi-steady CFD.

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 [10, 11]. Since wind tunnel tests typically use small-scale 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 [12]. In the 1980's, the Influence Function Method became popular [13], 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

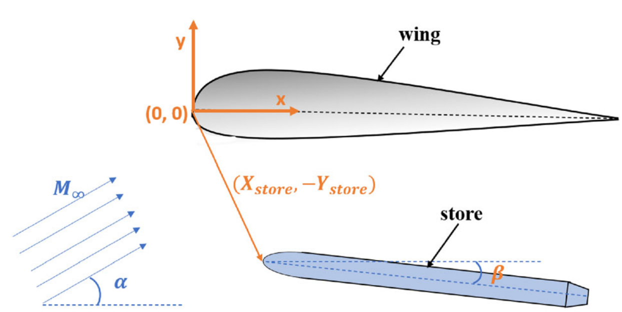


Figure 2. Setup of the two-dimensional 2-body store-separation problem.

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 [14], which further helped researchers to validate CFD approaches developed for trajectory prediction [15–19].

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 figure 1); 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 [20, 21] 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.

2.2 *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 2.1.

The application of parameter-varying steady ROM to single-body aerodynamics can be found in several works [1–6]. 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) [7, 8]. 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 [22], 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 [4, 5], 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 Ref. [9], and the development of the aerodynamic database of a missile (i.e., a single body) using (single domain) POD-ROM Ref. [6]. The former effort established the basic philosophy of the approach. However, it failed to achieve any substantial savings 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.

3. Methodology

3.1 *POD*

Let us denote the flow vector field by $\boldsymbol{q}(\boldsymbol{x};\boldsymbol{\mu})$, where $\boldsymbol{x} := (x, y)$ is the 2D Cartesian coordinate and $\boldsymbol{\mu}$ is the parameter vector. For example, in a 2D problem governed by Euler equations, $\boldsymbol{q} = [\rho, \rho u, \rho v, p]^{\mathrm{T}}$, where ρ is density, u and v are x- and y-components of velocity, and p is pressure. Density and velocity are normalized by the freestream density ρ_∞ and sound speed a_∞, respectively; pressure is normalized by $\rho_\infty a_\infty^2$. In POD, we assume that the flow vector field can be approximated as

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

where, $\overline{\boldsymbol{q}}(\boldsymbol{x})$ is the mean flow vector field (typically averaged over the flow solutions in the learning database) and the remaining 'fluctuations' are approximated as linear combinations of spatial basis functions $\{\widetilde{\boldsymbol{q}}^n(\boldsymbol{x})\}_{n=1}^{N_p}$ called POD modes, weighted by POD coefficients $\{\eta^n(\boldsymbol{\mu})\}_{n=1}^{N_p}$. For later reference, $\boldsymbol{\eta} := (\eta^1, \ldots, \eta^{N_p})^{\mathrm{T}}$. The actual determination of the POD modes follows the established 'snapshot' POD approach [6, 8, 23], and is not repeated here.

3.2 *Reduced-order model*

The ROM predicts the flow field for a new parameter vector $\boldsymbol{\mu}_0$ by invoking the governing equations. This is a more robust and accurate approach than the more straightforward interpolation in the parameter space. The POD-based ROM technique employed here was originally developed for single-body steady aerodynamics [1–5]; particular details relevant to this work may be found in Ref. [6]. Here we give only a brief overview.

Let the vector of governing (unsteady) conservation equations and boundary conditions be represented as

$$\frac{\partial(\boldsymbol{C}(\boldsymbol{q}))}{\partial t} = \boldsymbol{R}(\boldsymbol{q}),\ \boldsymbol{x} \in \Omega, \quad \text{s.t. } \boldsymbol{B}(\boldsymbol{q}) = 0,\ \boldsymbol{x} \in \partial\Omega. \tag{2}$$

Here, $\boldsymbol{C}$ is the operator that maps $\boldsymbol{q}$ to the vector of conserved flow variables, $\boldsymbol{R}(\boldsymbol{q})$ is a shorthand notation for the terms other than the local time derivative in the vector governing equations, Ω represents the flow domain, and $\boldsymbol{B}(\boldsymbol{q}) = 0$ codifies the conditions imposed on the boundary $\partial\Omega$. Since the solution $\boldsymbol{q}(\boldsymbol{x}; \boldsymbol{\mu}_0)$ must be steady, we should ideally have $\boldsymbol{R}(\boldsymbol{q}(\boldsymbol{x}; \boldsymbol{\mu}_0)) = 0$ along with $\boldsymbol{B}(\boldsymbol{q}(\boldsymbol{x}; \boldsymbol{\mu}_0)) = 0$. Just as in the FOM, we cannot hope for the ROM to find such a solution that exactly satisfies these conditions at all interior and boundary points in the flow; of course, the match is expected to be worse for the ROM.

Once the POD modes are determined from the learning database, only the coefficient vector $\boldsymbol{\eta}$ is unknown in the approximate expansion of eqn. 1. Thus, we can write – $\boldsymbol{R}(\boldsymbol{q}(\boldsymbol{x}; \boldsymbol{\mu}_0)) \approx: \widetilde{\boldsymbol{R}}(\boldsymbol{x}; \boldsymbol{\eta}(\boldsymbol{\mu}_0))$ and $\boldsymbol{B}(\boldsymbol{q}(\boldsymbol{x}; \boldsymbol{\mu}_0)) \approx: \widetilde{\boldsymbol{B}}(\boldsymbol{x}; \boldsymbol{\eta}(\boldsymbol{\mu}_0))$. This notation reinforces the fact that the residual at any point only requires knowledge of the POD coefficients, as does the boundary condition function.

For a given $\boldsymbol{\mu}_0$, the optimization problem is posed as:

$$\min_{\boldsymbol{\eta}} \|\widetilde{\boldsymbol{R}}(\cdot; \boldsymbol{\eta}(\boldsymbol{\mu}_0))\|^p_{\Omega,\mathcal{L}_p} \quad \text{s.t. } \|\widetilde{\boldsymbol{B}}(\cdot; \boldsymbol{\eta}(\boldsymbol{\mu}_0))\|^r_{\partial\Omega,\mathcal{L}_r} < \epsilon. \tag{3}$$

Here, $\mathcal{L}_p$ is the p-norm, and the domain of evaluation is indicated in the subscript of the norm too; ϵ is a suitably chosen threshold. In the present work, we have used the $\mathscr{L}_1$ norm of the residual. Moreover, there are so-called hyper-reduction techniques that drastically reduce the number of control volumes in which the residual has to be evaluated [6].

3.3 *Multi-domain-decomposed ROM*

Consider the two-body problem in figure 3, where the store is under the influence of the wing and can take any arbitrary position relative to the wing. Due to this continuous change in the placement of the store, we can not implement the ROM based on POD for the full domain as discussed in section 1. Following Ref. [9], we propose a methodology termed multi domain decomposition ROM (MDDROM) to circumvent this issue. As shown in figure 3, we decompose the overall domain into multiple sub-domains as follows:

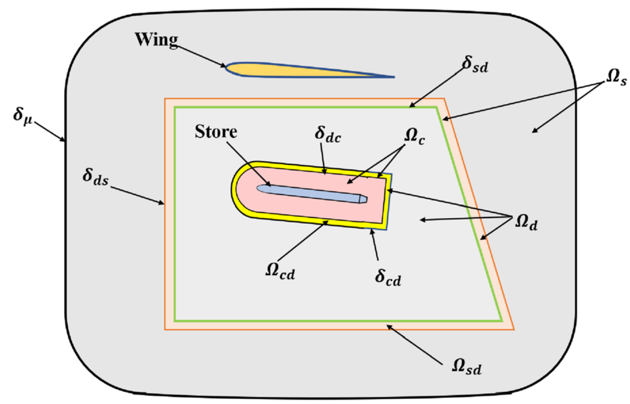


Figure 3. Three domain-decomposed ROM approach for the two-body problem of figure 2.

- *Capsule*, Ω_c – region immediately surrounding the store, which comprises a grid that does not change as the store moves.
- *Dropbox*, Ω_d – maximal region enveloping the capsule where it may be expected to reach in its separation trajectory and still remain under the influence of the wing.
- *Staticzone*, Ω_s – remaining flow domain surrounding the wing and dropbox and extending to the far-field boundary.

The staticzone and dropbox sub-domains are designed to have a small overlap – the staticzone-dropbox overlap Ω_{sd} in figure 3. Similarly, the overlap of the capsule with the dropbox yields Ω_{cd}. With careful attention, the overall mesh may be designed so as to remain unchanged in the capsule and staticzone sub-domains. The dropbox is the only sub-domain where the mesh needs to change in the course of the store separation. Evidently, POD-based ROM may be applied to Ω_c and Ω_s, and it is only in Ω_d that one has to look for a FOM solution. But this region, being away from solid boundaries, should also have the least number of mesh cells, thereby potentiating significant computational savings. In the work of Ref. [9], the dropbox and capsule were a single sub-domain. By identifying the capsule sub-domain containing a finely-resolved mesh around the store as a POD-ROM domain, we seek even greater efficiencies now.

The MDDROM approach is an iterative procedure, as shown in figure 4. The iterative process starts with

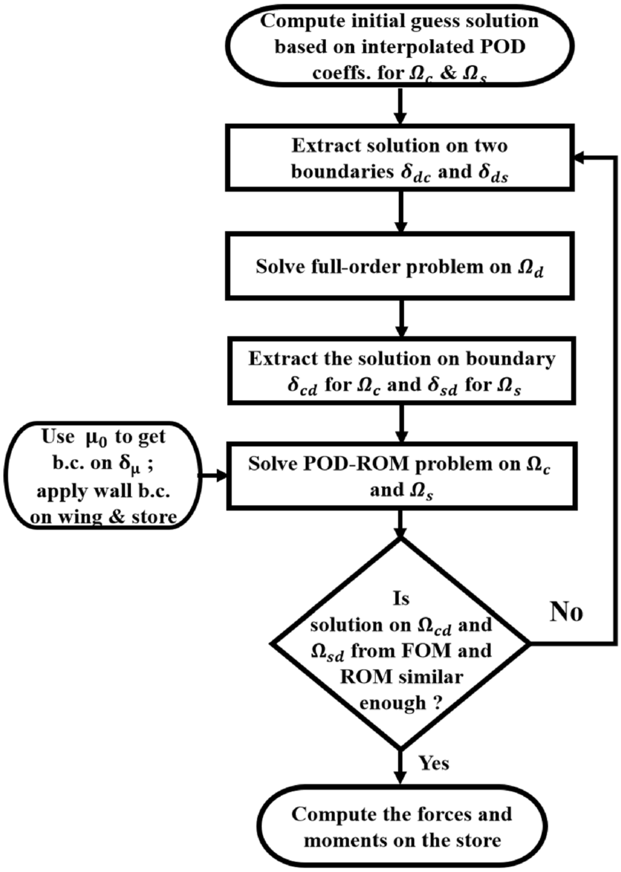


Figure 4. Flow chart of steps involved in three domain-decomposed ROM for 2-body flow analysis.

computing initial solutions on Ω_s and Ω_c by interpolating POD coefficients in parameter space from the learning database. From these solutions, we extract the flow variables on the interior interfaces δ_{ds} and δ_{dc}, respectively (see figure 3). These together serve as the (non-uniform) Dirichlet boundary conditions for the FOM solved in Ω_d. From this FOM solution, we extract the flow variables on the interior interfaces δ_{sd} and δ_{cd} (see figure 3 again). The flow information on δ_{sd}, along with the far-field boundary condition, and wall boundary condition on the wing, serve as constraints for the POD-based ROM solution pursued in the staticzone sub-domain Ω_s. Similarly, the flow information on δ_{cd} and the wall boundary condition on the store, fully specify the POD-based ROM to be solved for the capsule sub-domain Ω_c. Now that we have the ROM solution in Ω_s and FOM solution in Ω_d, the solutions are compared on the overlap regions Ω_{sd}; a similar check is done for the other overlap region Ω_{cd}. If the both comparisons are satisfactory, then we have a self-consistent solution over the entire flow domain. Else, we go for another iteration.

The learning database for the present work is generated by solving Euler equations; so the ROM, as well as FOM involved in the MDDROM, will also use the Euler equations. It is shown in many of the earlier works on ROM for single-body aerodynamics referred earlier that, even if the learning database uses RANS and the ROM minimizes Euler residuals, the predicted flow fields automatically satisfy the no-slip wall condition. Furthermore, the dropbox sub-domain being away from boundaries, one may solve the Euler equations therein too without significant penalty.

4. Results and discussion

We have evaluated results based on the proposed MDDROM method for its verification. A two-dimensional domain having two bodies, a wing-like body (airfoil in 2D) and a store-like body under the influence of the wing is considered for the analysis figure 2.

4.1 *Geometry and mesh generation*

A circular domain having two bodies, i.e. airfoil, and store is constructed. The airfoil is a standard RAE-2822 profile, with chord c. The store is a rectangular slender body, with a nose and a boat-shaped tail. The length of the store is $0.5c$ and its slenderness ratio is 10, thereby making its thickness $d = 0.05c$. The nose is a circular-blunted tangent ogive, with nose radius and nose length as $0.05d$ and $1.5d$. The boat-tail has a half wedge angle of 20° and its length equals d. A circle centered at the leading edge of the airfoil and having a radius $50c$ constitutes the far-field boundary.

Gmsh v4.8.4, an open-source software, is employed for the mesh generation [24]. A structured mesh is constructed around the airfoil and the store as well as for the overlap regions for ease of handling different domains, and an unstructured mesh is generated for the remaining domain. Figure 5, depicts the mesh topology for the domain, having quadrilateral cells for structured mesh and triangular cells for unstructured mesh. The overall mesh generation process is automated using Python scripts in integration with Gmsh. This allows us to ensure the same mesh (with the same node and cell numbering) exists in the capsule and static-zone subdomains for all positions of the store.

4.2 *Database generation and POD*

The final learning database should comprise of snapshots of the entire store-separation trajectory for various choices of wing $M_\infty - \alpha$ pairs. However, for the present verification exercise, we treat these wing operating conditions and the three position DOFs of the store as independent parameters of a much smaller learning database. Thus, our parameter vector is $\boldsymbol{\mu} = (M_\infty, \alpha, X_{store}, Y_{store}, \beta)$ (see figure 2). The sets of values used for these five parameters are given in table 1; all possible combinations of them yield 162

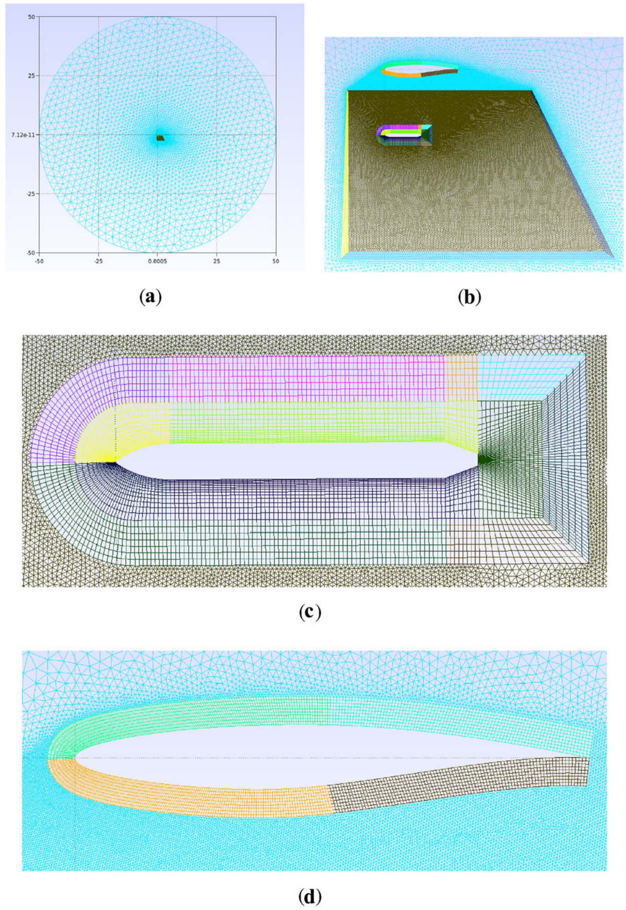


Figure 5. Mesh topology for wing-store in 2D. (a) Overall domain. Zoomed views of (b) dropbox sub-domain, (c) store with body-fitted structured mesh as capsule sub-domain, and (d) wing with body-fitted structured mesh.

Table 1. Operating parameters for learning database.

Parameter, $\boldsymbol{\mu}$	Values
Angle of attack, α	$0°, 1°, 2°$
Free stream Mach no., M_∞	0.45, 0.47, 0.49
x-position of store, X_{store}	0, 0.2, 0.4
y-position of store, Y_{store}	−0.9, −1.0, −1.2
Rotation angle of store, β	$0°$, $-1°$

snapshots. As mentioned earlier, the present work uses Euler calculations, for which SU2 v7.1.1 [25] is used. The far-field boundary is provided with the free-stream condition, and the wing and store surfaces are given a no-through-flow condition. The database generation is automated using Python scripts that integrate Gmsh and SU2.

The conserved variables obtained from the SU2 solution are converted into the chosen set of normalized POD variables mentioned in section 3.1. From the generated database, the flow solutions for staticzone and capsule sub-

Table 2. Top half: parameters of five verification cases. Bottom half: comparison between the truth and MDDROM model predictions of the store's lift coefficient c_l and pitching moment coefficient c_m about its nose, in these cases.

Case no.		I	II	III	IV	V
$\boldsymbol{\mu}$	α	0.5°	0.5°	0.95°	1.2°	1.2°
	M_∞	0.46	0.46	0.47	0.48	0.46
	X_{store}	0.2	0.2	0.2	0.4	0.4
	Y_{store}	−1.0	−0.9	−0.1	−1.2	−1.2
	β	−1°	0°	−1°	0°	0°
Truth $c_l \times 10$		1.10	0.41	1.34	0.73	0.72
Model $c_l \times 10$		1.08	0.41	1.33	0.73	0.71
% error in c_l		1.82	0.00	0.75	0.00	1.39
Truth $c_m \times 100$		1.53	0.65	1.82	0.94	0.93
Model $c_m \times 100$		1.49	0.66	1.8	0.94	0.92
% error in c_m		2.61	1.54	1.10	0.00	1.08

domains are extracted, and vector POD is performed on the snapshots in each sub-domain separately for these two sub-domains. The fractional eigenspectra of POD corresponding to the two sub-domains are shown in figure 6. There is a rapid decrease (4 orders of magnitude) in the 'energy' content (or relative importance) of the POD modes from mode 2 to 3 in the staticzone sub-domain. The reason for this rapid decrease is that the first few modes capture the flow characteristics for the overall domain, whereas subsequent higher modes capture the localized flow features near the wing. On the contrary, the same is not true for the capsule sub-domain, because the capsule sub-domain consists of a very narrow region near the store, and all the parameters' variations affect the flow in this region. For the staticzone, POD mode 10 accounts for about 10^{-6} less 'energy' vis-à-vis POD mode 1; for the capsule, it is POD mode 15 that has about this relation with its POD mode 1.

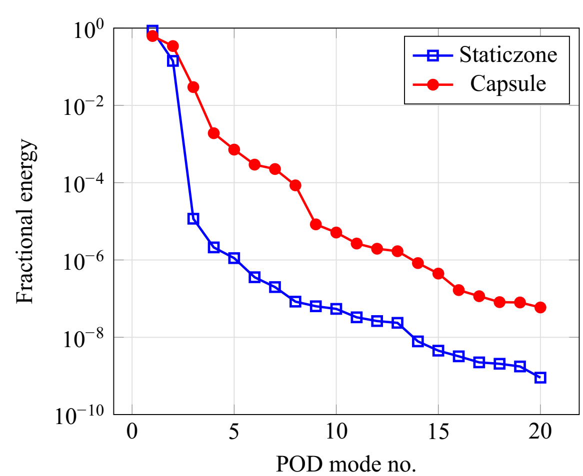
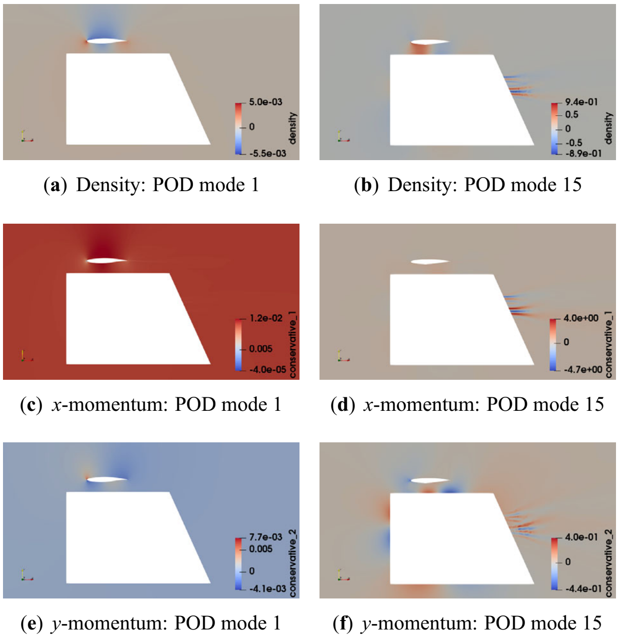


Figure 6. Vector POD eigenspectra for the staticzone and capsule sub-domains in terms of the fractional energy content – viz. $\lambda_i / \sum_j \lambda_j$, where λ_i is i^{th} POD eigenvalue.

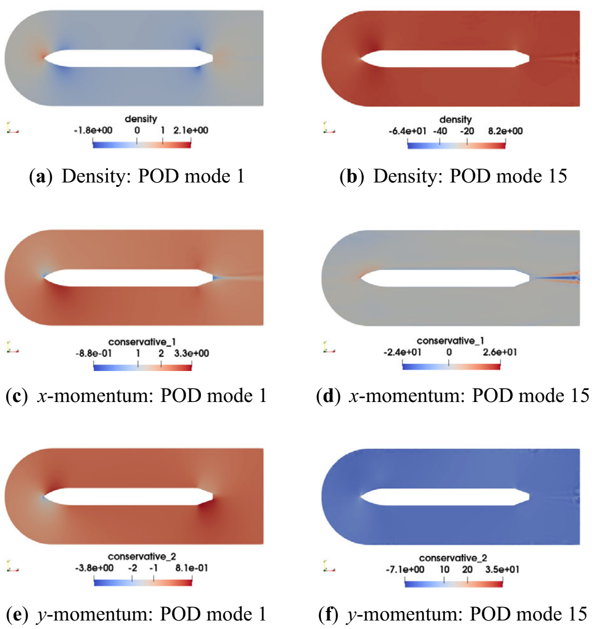


(**a**) Density: POD mode 1 (**b**) Density: POD mode 15

(**c**) x-momentum: POD mode 1 (**d**) x-momentum: POD mode 15

(**e**) y-momentum: POD mode 1 (**f**) y-momentum: POD mode 15

Figure 7. POD mode 1 (left column) and mode 15 (right column) of the staticzone sub-domain.



(**a**) Density: POD mode 1 (**b**) Density: POD mode 15

(**c**) x-momentum: POD mode 1 (**d**) x-momentum: POD mode 15

(**e**) y-momentum: POD mode 1 (**f**) y-momentum: POD mode 15

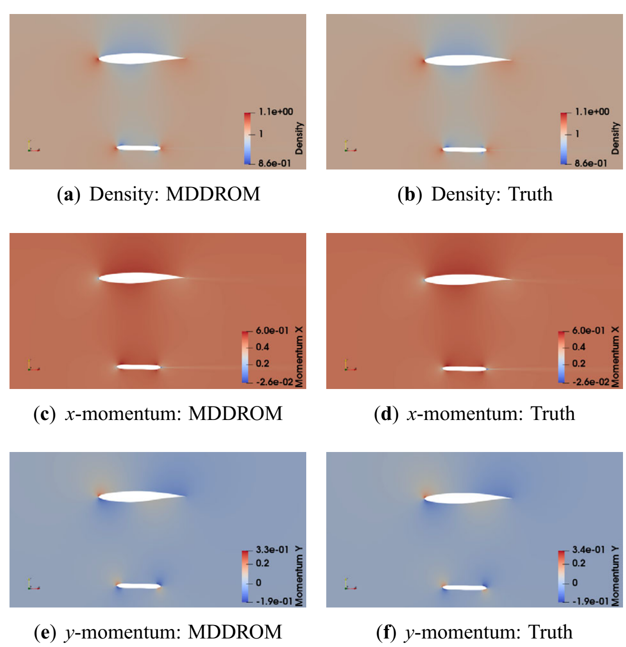
Figure 8. POD mode 1 (left column) and mode 15 (right column) of the capsule sub-domain.

Therefore, for reconstructing the flow field using ROM, we use 10 and 15 POD modes for the staticzone and capsule sub-domains, respectively.

The shapes of POD modes 1 and 15 are presented in Figures 7 and 8 for the staticzone and capsule sub-domains, respectively. Since the flows considered are weakly compressible, pressure follows density, and is hence omitted from the plots. The main motivation in presenting these extreme POD modes is to highlight the difference in the flow features captured by lower and higher POD modes. As stated previously, POD mode 1 of staticzone captures the flow characteristics for the overall domain, whereas its mode 15 counterpart displays localized flow features – viz. wakes of the store corresponding to its three $y-$positions included in the learning database. In the case of the capsule, POD mode 1 already displays some fine-scale structure; however, the structures in mode 15 are much finer.

4.3 *Verification of multi-domain-decomposed ROM*

The MDDROM approach necessitates an iterative solution of the POD-ROM in the staticzone and capsule sub-domains and the FOM in the intermediate dropbox sub-domain to sustain the continuity of the flow field. As mentioned earlier, Euler residuals are minimized in the POD-ROM. The sequential least-square quadratic programming (SLSQP) method available in the `scipy` library of Python is employed for this optimization. The FOM solves the Euler equation using SU2, which has been augmented to implement the non-uniform boundary conditions arising at the interface between the sub-domains. The conserved variables solutions at the interface boundaries between dropbox-staticzone (δ_{ds}) and dropbox-capsule (δ_{dc}) are provided as custom boundary conditions in SU2. The POD-ROM and SU2-FOM are driven iteratively from a Python script.

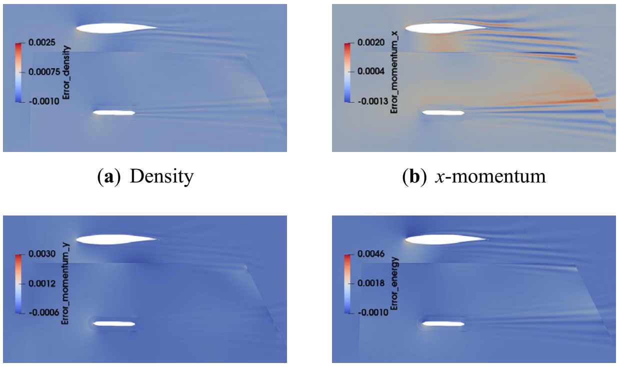


(**a**) Density: MDDROM (**b**) Density: Truth

(**c**) x-momentum: MDDROM (**d**) x-momentum: Truth

(**e**) y-momentum: MDDROM (**f**) y-momentum: Truth

Figure 9. MDDROM solution (left column) and the 'truth' result (right column) for case I of table 2.



(**a**) Density (**b**) x-momentum

(**c**) y-momentum (**d**) Energy

Figure 10. Error in conserved variables for case I of table 2.

The MDDROM approach is verified using five new cases as delineated in table 2. Full-order solutions for these cases are computed using SU2's Euler solver and are used as truth solutions in the verification exercise. For illustration, representative flow variables from the case I computed using the MDDROM approach are compared with their 'truth counterparts in figure 9. The visual agreement is very reasonable. The error contour plots (see figure 10) corresponding to the case I (table 2) for all conserved variables (i.e., Density, $x-$ momentum, $y-$momentum, and Energy) are also computed for the demonstration of magnitude as well as the location of maximum error. Error is defined as the difference in the magnitude of the flow variables between MDDROM and truth. It can be observed that the maximum non-dimensional error in the flow variables is of the order of 10^{-3}, which can be admitted considering the practical applicability of the results. The error is even lower on the store surface, where forces and moment coefficients are to be predicted.

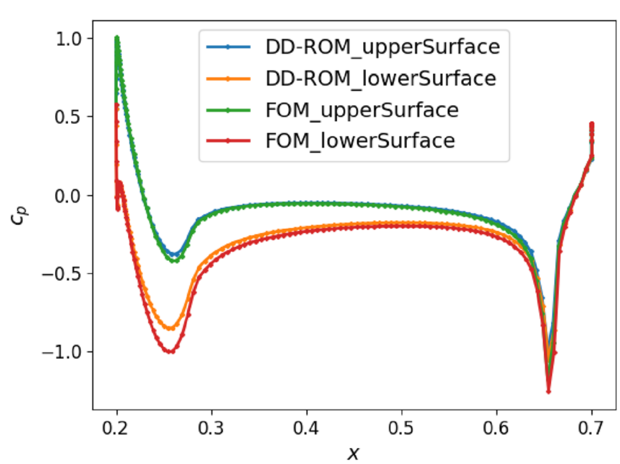


Figure 11. Store pressure coefficient for case I of table 2.

The primary objective of the proposed methodology is to predict the store-separation trajectory by computing the aerodynamic forces and moment acting on the store in an efficient manner. Hence, the store's lift coefficient and moment coefficient are computed from the true solution and the MDDROM, and they are compared in the bottom half of table 2. The percentage absolute error is below 2% for the (c_l) and below 3% for the (c_m), which may be well within the range of practical applicability of the results. The store-surface pressure coefficient c_p, being the ingredient for calculating its lift and pitching moment coefficients, is also presented in figure 11 for case I of table 2. Again, the MDDROM prediction closely follows the truth c_p plot.

5. Conclusions

This paper presents a POD-based ROM approach in conjunction with multi-domain-decomposition methodology (MDDROM) 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 of aircraft-store dyads. The proposed approach replaces Euler simulations on the flow domain with a relatively faster multi-variable optimization problem for the major portion of the domain. It resorts to Euler calculations for a very small number of mesh cells that necessarily change when the store moves relative to the aircraft. The approach is verified on a two-dimensional two-body problem, where Euler calculations are pursued in the full domain for the generation of the underlying empirical database. Comparison of MDDROM results with full-order Euler simulations reveals very encouraging agreement, both in terms of flow structures as well as the lift coefficient of the store.

For preliminary demonstration and verification of the proposed method, the low-fidelity Euler solver is implemented for FOM calculations at present. We propose to use RANS calculations in the future, and investigate if the MDDROM method continues to deliver reasonably accurate results. Moreover, the actual store-separation problem will also be addressed with the MDDROM.

References

[1] LeGresley P and Alonso J. 2001. Investigation of non-linear projection for POD based reduced order models for aerodynamics. In *39th AIAA Aerospace Sciences Meeting and Exhibit, paper 0926*

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

[3] Alonso D, Velázquez A and Vega J M 2009 A method to generate computationally efficient reduced order models.

Computer Methods in Applied Mechanics and Engineering 198(33–36): 2683–2691

[4] Alonso D, Vega J M, Velázquez A and Pablo V 2012 Reduced-order modeling of three-dimensional external aerodynamic flows. *Journal of Aerospace Engineering* 25: 588–599

[5] Zimmermann R and Görtz S 2010 Non-linear reduced order models for steady aerodynamics. *Procedia Computer Science* 1: 165–174

[6] Sinha A, Kumar R and Umakant J 2020 Reduced-order model for efficient generation of a subsonic missile's aerodynamic database. *The Aeronautical Journal* 126(303): 1546–1567

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

[8] Holmes P, Lumley J L, Berkooz G and Rowley C W 2012 Turbulence, coherent structures, dynamical systems and symmetry. Cambridge University Press

[9] Sinha A and Garg S 2018 Reduced-order modeling of steady aerodynamics for 2D store separation analysis. In *36th AIAA Applied Aerodynamics Conference, paper 3168*

[10] Bamber M J 1960 Two methods of obtaining aircraft trajectories from wind tunnel investigations. Technical Report 970 (AD 233198), David Taylor Model Basin, Washington DC

[11] Christopher J P and Carleton W E 1968 Captive-trajectory store-separation system of the AEDC-PWT 4-foot transonic tunnel. Technical Report AEDC-TR-68-200 (AD 839743), Arnold Engineering Development Center

[12] Rogers R M 1976 A comparison between the Nielson and Woodward programs in predicting flow fields and store loads. Technical Report 2854, Naval WeaponsCenter

[13] Meyer R, Cenko A, and Yaros S 1981 An influence function method for predicting aerodynamic characteristics during weapon separation. In *12th Navy Symposium on Aeroballistics*

[14] Heim E R 1991 CFD wing/pylon/finned store mutual interference wind tunnel experiment. Technical Report AEDC-TSR-91-P4, Defense Technical Information Center

[15] Meakin R 1992 Computations of the unsteady flow about a generic wing/pylon/finned-store configuration. In *1992 AIAA/AAS Astrodynamics Conference, paper 4568*

[16] Newman J C, III and Baysal O 1992 Transonic solutions of a wing/pylon/finned store using hybrid domaindecomposition. In *1992 AIAA/AAS Astrodynamics Conference, paper 4571*

[17] Parikh P, Pirzadeh S and Frink N 1992 Unstructured grid solutions to a wing/pylon/store configuration using VGRID3D/USM3D. In *1992 AIAA/AAS Astrodynamics Conference, paper 4572*

[18] Madson M, Moyer S and Cenko A 1994 TranAir computations of the flow about a generic wing/pylon/finned-store configuration. In *32nd AIAA Aerospace Sciences Meeting and Exhibit, paper 0155*

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

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

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

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

[23] Sirovich L 1987 Turbulence and the dynamics of coherent structures. I - Coherent structures. II - Symmetries and transformations. III - Dynamics and scaling. *Quarterly of Applied Mathematics* 45(3): 561–590

[24] Geuzaine C and Remacle J-F 2009 Gmsh: A 3-D finite element mesh generator with built-in pre-and post-processing facilities. *International Journal for Numerical Methods in Engineering* 79(11): 1309–1331

[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 2014 Stanford university unstructured (SU2): Open-source analysis and design technology for turbulent flows. In *52nd AIAA Aerospace Sciences Meeting, paper 0243*