AIAA AVIATION Forum
June 25-29, 2018, Atlanta, Georgia
2018 Applied Aerodynamics Conference

10.2514/6.2018-3168

Reduced-Order Modeling of Steady Aerodynamics for 2D Store Separation Analysis

Aniruddha Sinha[1]* and Sparsh Garg[2]†

[1] *Dept. of Aerospace Engineering, Indian Institute of Technology Bombay, Mumbai 400076, INDIA*

[2] *Dept. of Aerospace Engineering, PEC University of Technology, Chandigarh 160012, INDIA*

We report on a novel empirical reduced-order model (ROM) approach to efficiently simulate the aerodynamics of two nearby bodies that can have a wide range of relative position and orientation. The method is demonstrated for a pair of two-dimensional slender bodies, but is potentially applicable to external store separation analysis where the aircraft may experience a range of freestream Mach numbers and angles of attack. The entire flow domain is decomposed into two sub-domains. The smaller sub-domain typically extends for some distance below the wing to cover the possible range of positions of the store beneath it where it is still influenced by the wing. The remaining flow domain is typically much larger, and the aerodynamics here are approximately modelled using proper orthogonal decomposition modes and a ROM that minimizes conserved flux residues. The smaller sub-domain has to be solved with usual CFD methods like RANS. The two solutions are iteratively matched at the interface of the two sub-domains to arrive at the overall solution – a method that we term domain-decomposed ROM. Results presented here demonstrate encouraging agreement with the CFD simulation of the full flow, at about one-fourth of the computational cost.

I. Introduction

Whenever a new fighter aircraft is introduced into service, or an existing aircraft undergoes substantial modifications or must be certified to employ new stores, a thorough investigation of the safe-separation flight envelope is necessitated. There are three main approaches – flight testing, wind tunnel model experiments and computational fluid dynamics (CFD) model testing; usually a combination of two or all three of these methods is employed for improved effectiveness. All three approaches pose challenges in terms of resources and time; the first two also pose physical dangers.

The objective of the present research is to develop a reduced-order model (ROM) approach to the problem. ROM refers to an approximate representation of reality that captures the essential physics of a problem while remaining amenable to rapid evaluation. ROMs usually involve a judicious mix of empirical data and physics-based governing equations. As such, they are much sought after in situations requiring rapid design decisions, like certification tests of designs and multi-disciplinary optimization (MDO).

The ultimate aim of the research is the efficient characterization of the safe flight envelope for store separation. To this end, we develop a ROM for predicting the two-dimensional aerodynamics of two bodies in close proximity; an illustrative configuration in this scenario is shown in fig. 1a. In particular, the ROM tool is able to efficiently predict the approximate steady aerodynamic forces acting on a store (Body 2) when it is near a wing (Body 1), for a large range of freestream conditions (w.r.t. the wing) – viz. Mach numbers and angles of attack. Moreover, this prediction is possible for a range of positions and orientations of the store relative to the wing. Although the ROM tool presented here is for 2D subsonic aerodynamics, the approach thus developed will be extended to 3D subsequently. Such a tool may replace or augment expensive full-order CFD calculations in characterization of safe-separation flight envelopes.

*Assistant Professor; AIAA Member; Corresponding author: as@aero.iitb.ac.in
†Undergraduate student

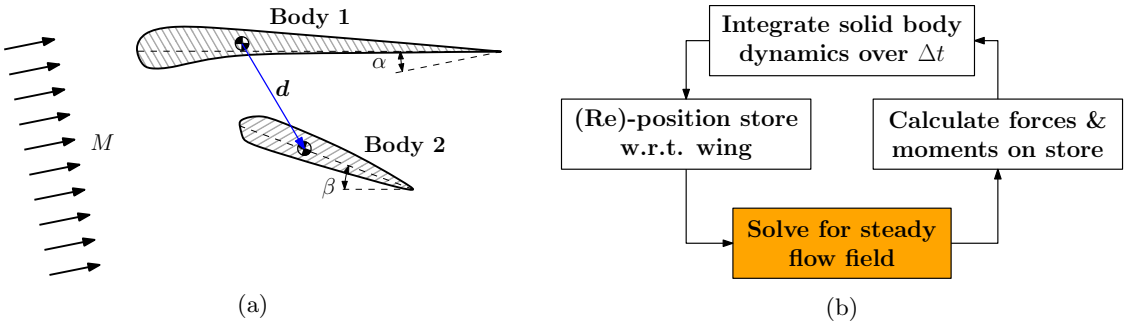


Figure 1: (a) Setup for reduced-order model of 2D aerodynamics of two bodies in an oncoming stream. (b) Flowchart of typical steps involved in quasi-steady simulation of store trajectory near an aircraft.

II. Background

We start with a brief background on the current state-of-the-art in trajectory prediction of separated stores in Section A. In Section B, we describe the previous applications of reduced-order modelling to similar problems.

A. Existing methods for trajectory prediction of separated stores

The earliest attempts at store separation analysis were flight tests conducted in a trial-error approach – the store would be dropped from an aircraft flying at increasing speeds as the store came closer and closer to the aircraft, or in some cases actually hit the aircraft.

Considerable improvements in the art resulted from the development of the captive trajectory system method for wind tunnel testing of store separation in the 1960's[1,2]. However, since the models that could be used in wind tunnels were fairly small, large discrepancies were sometimes noticed in comparisons made against flight test data.

One of the first uses of computer simulation in store separation studies was the Influence Function Method[3] in the 1980's. Wind tunnel test data on a general store positioned near a particular aircraft was used to determine the 'influence' coefficients at points in the vicinity. This database was used to determine the forces and moments on another test store positioned relative to the aircraft. The final trajectory was calculated using a 6-degree-of-freedom (6DOF) rigid body dynamics solver in an iterative loop with the force/moment determination step.

Improvements in computational fluid dynamics – in terms of methods, algorithms and computing resources – have yielded drastic savings in time and effort involved in the store-separation certification exercise. Nowadays, the major part of testing is done using CFD tools, with some test programs eschewing wind tunnel tests and flight tests altogether. A flowchart for the procedure is presented in fig. 1b.

The induction of CFD tools for the problem of predicting safe separation of stores was greatly facilitated by the consistent investment of the US Air Force and Navy in efforts to develop and validate CFD methods for this specific purpose in the 1990's and 2000's. A database was created in the early 1990's from detailed experiments documenting the flow field around a generic wing-pylon-store model under both captive and separated configurations[4]. This served as a test-bench for validating a hierarchy of CFD approaches – a transonic full-potential solver[5,6], two different inviscid Euler codes[7,8] and a viscous thin-layer unsteady Reynolds-averaged Navier-Stokes code[9]. All CFD approaches demonstrated encouraging agreement with the experimental database, with no clear distinction in the quality of the results. This provided an early indication that inviscid quasi-steady calculations may be sufficient for the purposes of predicting store trajectories in most cases. Other researchers reported on parallel efforts along similar lines – using steady inviscid/RANS codes with increasing sophistication in grid generation[10–12].

Although quasi-steady computations (of the kind charted in fig. 1b) were found to be sufficiently accurate for wing-mounted stores released in steady flight, doubts had remained regarding the validity of this approach for more complex cases like release of multiple stores, release during sudden manoeuvres, and release from weapon bays. One of the earliest successes in performing unsteady time-resolved flow simulations for the store release problem was reported in Ref. 13. However, the experimental data used for validation by those

authors were not from a case that may have warranted an unsteady CFD approach, and no results were reported for the same case solved using the quasi-steady approach. More recently, the store separation problem with unsteady Euler calculations has been revisited in Refs. 14,15, with further improvements in solution methodologies. The authors re-confirm that for external stores released from aircraft in steady flight, the inviscid quasi-steady approach is sufficiently accurate, and the inviscid unsteady CFD calculations do not yield any distinct advantage while costing much in resources. Even in simulations of cases where the aircraft makes sudden manoeuvres at the time of store release, the quasi-steady and unsteady calculations did not show any significant difference in results pertaining to the store trajectory; however, the store attitude predictions of the two approaches were found to disagree.

The problem of predicting the trajectory of a store released from a weapon bay is significantly different, due to the unsteadiness of the flow field over a cavity. As such, this problem cannot be solved with the quasi-steady formulation (see for example Ref. 16). Our ROM approach is also unsuitable for this problem.

B. Previous applications of ROMs for similar problems

In discussing the previous reduced-order modelling approaches relevant to the present work, 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 – examples may be found in Refs. 17–21. 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 indicated in Section A, it has been repeatedly determined that steady CFD techniques are sufficient for the prediction of store separation trajectories. Thus, steady ROM approaches, as reported in Refs. 22–26, are most relevant for the problem at hand.

The steady ROM development starts from 'snapshots' of the steady flow field for different values of a set of parameters of the problem. For example, for predicting the steady aerodynamic loads on a wing at a particular freestream condition of Mach number, angle of attack and side-slip angle, the snapshots would consist of the steady flow field at a sparse set of other freestream conditions, obtained from prior full-order numerical computations (or experiments). This step requires heavy computational effort – it represents the upfront cost of the ROM approach.

The next step is the eduction of a set of basis functions (or modes) that represents the dominant flow variations observed across all snapshots. In the literature, this is consistently pursued with proper orthogonal decomposition (POD). 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 set of linearly uncorrelated variables (basis functions/modes)[27,28]. The POD modes afford a low-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 final step is the prediction of the steady flow solution for a particular parameter set that was not computed in the snapshot-creation step. This is pursued in two main ways – interpolation (or extrapolation) of the available POD modes[22], or derivation of a flow solution as a linear combination of the POD modes that optimally satisfies the steady governing equations of the flow along with appropriate boundary conditions[23–26]. Strictly speaking, only the latter method can be termed 'steady ROM'. The steady ROM method is more robust and versatile, although it is more computationally expensive than the interpolation approach.

The steady ROM approach has been shown to be useful for predicting the steady aerodynamics of an aircraft wing in transonic conditions. In this paper, we extend it to the more complicated two-body (wing-store) problem.

III. ROM approach

The preceding literature survey has established that inviscid, quasi-steady CFD is suitable for the computational approach to the store separation problem. Thus, we develop a corresponding inviscid, steady reduced-order model for the problem. Section A describes the existing ROM approach for predicting single-body steady aerodynamic loads. In Section B, we outline a method to use the above ROM in a two-body setting. Implementation details are provided in Section C.

A. ROM of single body aerodynamics

The basic methodology for creating a ROM for steady aerodynamics of a single body has been adopted from LeGresley and Alonso[23]. This has been subsequently refined by Alonso et al.[24],[25] and Zimmermann and Görtz[26].

Let $\boldsymbol{q}$ represent the vector field of flow variables over the domain of interest. For the 2D Euler equations, one choice of $\boldsymbol{q}$ is $[\rho, u, v, p]^{\mathrm{T}}$, where ρ is the density, u and v are respectively the x and y Cartesian components of velocity, and p is the pressure. Let us represent the vector of governing conservation equations as

$$\frac{\partial}{\partial t}(\mathscr{C}(\boldsymbol{q})) = \mathscr{R}(\boldsymbol{q};\boldsymbol{\mu}), \tag{1}$$

where $\mathscr{C}$ is the operator that maps the primitive flow variables $\boldsymbol{q}$ to the conserved flow variables, and $\mathscr{R}$ is a shorthand notation for the remaining terms (other than the explicit partial time derivative). Note that for steady governing equations, $\mathscr{R}$ represents the 'residual'. The set of operating conditions (viz. Mach number and angle of attack) that parametrize the function $\mathscr{R}$ are denoted by $\boldsymbol{\mu}$. We are interested in the steady solution of the governing equations for a particular choice of $\boldsymbol{\mu}$, say $\boldsymbol{\mu}_0$. In other words, we need to find the vector field $\boldsymbol{q}$ such that $\mathscr{R}(\boldsymbol{q};\boldsymbol{\mu}_0)$ vanishes.

In the approach proposed in Ref. 23, the vector field $\boldsymbol{q}$ is approximately expanded in terms of a few proper orthogonal decomposition (POD) modes (say N in number):

$$\boldsymbol{q}(\boldsymbol{x};\boldsymbol{\mu}) \approx \boldsymbol{Q}(\eta_1(\boldsymbol{\mu}), \boldsymbol{\Phi}_1(\boldsymbol{x}), \eta_2(\boldsymbol{\mu}), \boldsymbol{\Phi}_2(\boldsymbol{x}), \ldots, \eta_N(\boldsymbol{\mu}), \boldsymbol{\Phi}_N(\boldsymbol{x})) \tag{2}$$

Here, $\boldsymbol{\Phi}_1$, $\boldsymbol{\Phi}_2$, etc. are the leading spatial POD modes of the data $\boldsymbol{q}$ available in a (learning) set of the parameters $\boldsymbol{\mu}$; this empirical information typically comes from full-order simulations. The corresponding POD coefficients are η_1, η_2, etc, and these vary with $\boldsymbol{\mu}$. Using this expansion, the residual $\mathscr{R}$ can be approximated as a function of η_1, η_2, etc. since the $\boldsymbol{\Phi}_1$, $\boldsymbol{\Phi}_2$, etc. are known at this point; i.e.,

$$\mathscr{R}(\boldsymbol{q};\boldsymbol{\mu}) \approx \mathscr{R}(\eta_1, \eta_2, \ldots;\boldsymbol{\mu}). \tag{3}$$

Since the truncated POD expansion is an approximation, in general we will not be able to find a set of $(\eta_1, \eta_2, \ldots, \eta_N)$ such that $\mathscr{R}$ vanishes exactly. Instead, we define a positive semi-definite functional J whose minimum will correspond to the closest approximation to the desired solution. One possible choice for J is

$$J(\eta_1, \eta_2, \ldots, \eta_N;\boldsymbol{\mu}) = \varepsilon||\mathscr{R}_\Omega(\eta_1, \eta_2, \ldots, \eta_N;\boldsymbol{\mu})||_2^2 + (1-\varepsilon)||\mathscr{E}_\partial(\eta_1, \eta_2, \ldots, \eta_N;\boldsymbol{\mu})||_2^2. \tag{4}$$

Here $\mathscr{R}_\Omega$ represents the discretized residual over the solution domain, and $\mathscr{E}_\partial$ denotes the discretized boundary condition errors. The first term determines the error in the solution over the domain, in the sense of the squared 2-norm of the residual. The second term encodes the solution mismatch at the boundary, where the parameters $\boldsymbol{\mu}$ typically appear explicitly. The relative weights of the two terms are determined by the factor ε. The task of finding the steady state solution $\boldsymbol{q}$ for the parameter set $\boldsymbol{\mu}_0$ is thus transformed into one of minimizing J over the set $(\eta_1, \eta_2, \ldots, \eta_N)$.

The description elides many of the implementation details. Over the years, several attempts have been made to arrive at efficient yet accurate models of this type[24–26]. Some of the key conclusions that are applicable to the present problem are:

- The POD modes $\boldsymbol{\Phi}$ 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. In particular, the different angles of attack should be represented by different angles of inflow, and not by different orientations of the body itself.

- Although the 'training' solutions of $\boldsymbol{q}$ generally come from high-fidelity RANS, the residual may be computed using a low fidelity Euler solver. This is because the lowest-order POD modes cannot resolve the localized viscous/turbulent effects anyway (they being representative of global behaviour), although they indirectly account for these effects since they arise from the RANS results that incorporate the no-slip condition.

- The evaluation of the solution residual need not be performed over the entire domain. Instead, critical regions may be identified from prior experience, and the residual may be evaluated in those regions alone. This reduces the overhead involved in the optimization.

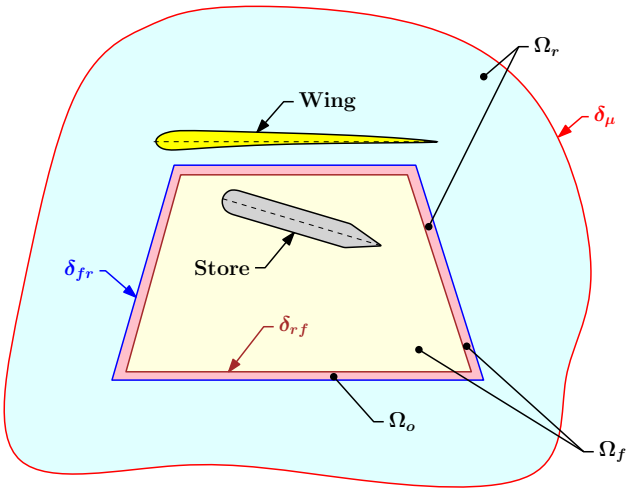


Figure 2: Setup for domain-decomposed reduced-order solution of two-body aerodynamics.

- In the POD literature, a distinction is made between 'scalar' and 'vector' POD modes [17]. In the scalar approach, each independent variable (e.g., ρ, u, etc.) is separately subjected to the POD procedure, and is subsequently expanded in terms of the corresponding POD modes with their respective POD coefficients. In the vector approach, all the relevant variables are together subjected to the POD procedure, and are subsequently expanded in terms of the corresponding POD modes with the same POD coefficients. Both these approaches are formally represented by the expansion in eqn. (2). In the present work, we use the scalar approach.
- The ROM calculation is initialized with the interpolated POD mode coefficients from the learning database [22].

B. Domain-decomposed ROM of two-body aerodynamics

One of the constraints of the POD-ROM method described in Section A is that the geometry of the problem must remain unchanged across all possible parameters $\boldsymbol{\mu}$. This becomes impossible in a two-body setting when we wish to allow variations in relative position and orientation (see fig. 1a). To circumvent this problem, we use the method of 'domain decomposition' in conjunction with the POD-ROM approach described above. This procedure will be termed domain-decomposed ROM (DD-ROM), and may readily replace the 'Solve for steady flow field' sub-step highlighted in the original store trajectory calculation methodology presented in fig. 1b.

Consider the situation in fig. 2, where the overall flow domain is decomposed into two sub-domains, Ω_r and Ω_f, having an overlap denoted by Ω_o. The smaller domain Ω_f is designed to always contain the store. This is practical since we are interested in a store separating and falling under its own weight in the vicinity of the wing. The aerodynamics simplify greatly when the store leaves the proximate zone of influence of the wing, and we can use single-body ROM tool in this regime. In this set-up, δ_{fr} denotes the boundary of Ω_f that is internal to Ω_r, and δ_{rf} is the boundary of Ω_r that is internal to Ω_f. Notice that whereas Ω_f has a variable geometry, Ω_r has a fixed geometry for all choices of $\boldsymbol{\mu}$ and all possible trajectories of the store as long as the aircraft wing geometry is fixed. The far-field boundary, which is directly influenced by the specified parameters $\boldsymbol{\mu}$, is denoted δ_μ.

We solve for the flow field in Ω_r using the POD-ROM procedure elucidated in Section A. The flow in Ω_f has to be solved with the full-order solver. The two flow fields must be (approximately) matched in the overlap domain Ω_o, which will typically require a few iterations. However, the efficiency of the POD-ROM approach and the compactness of Ω_f should still result in significant computational savings compared to the original approach of applying the full-order solver to the overall domain. The situation is not very different from the multi-block solution process adopted by parallelized CFD solvers. The flowchart for the approach is presented in fig. 3. Note that whereas Euler equations will be (approximately) solved in the POD-ROM domain, RANS equations may be solved in Ω_f, depending on preliminary assessments.

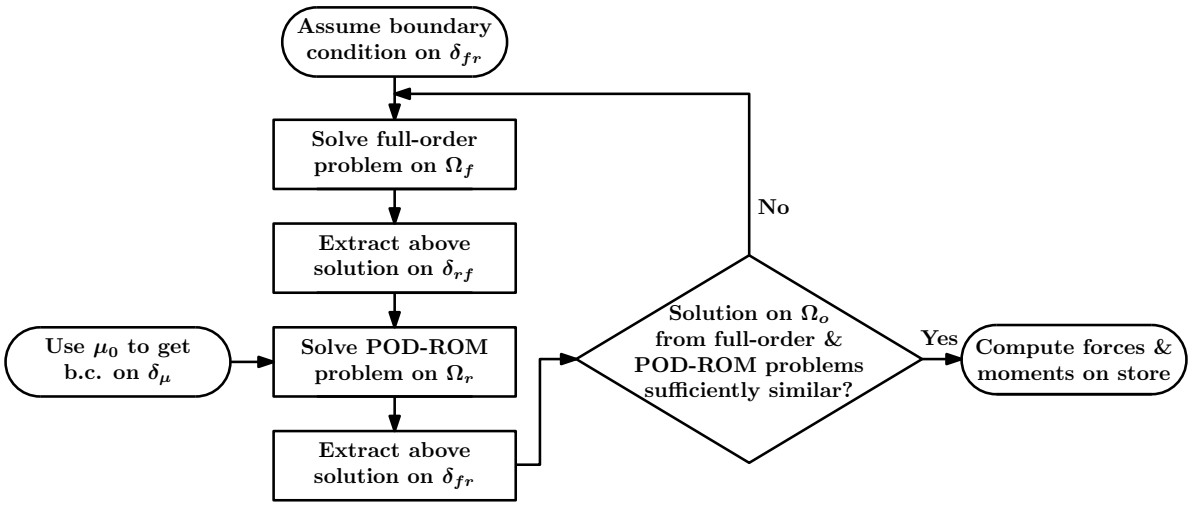


Figure 3: Flowchart of steps in domain-decomposed reduced-order solution of two-body aerodynamics.

The steps for deriving the POD modes in the Ω_r sub-domain for use in the ROM are charted in fig. 4. Full-order solutions (learning snapshots) need to be obtained for a set of representative parameters $\boldsymbol{\mu}$ and store positions (relative to the wing); in this process, the mesh in Ω_r must be kept unchanged. These solutions, extracted in Ω_r, can be submitted to the POD computation as before.

The DD-ROM technique delineated above still requires the full-order simulation in a region surrounding the store that includes the refined mesh in the vicinity of the store. To improve the situation, it may be feasible to develop a ROM for the store in a domain enclosing it and moving with it. This will require the matching of solutions at another interface, this time between the store ROM domain and Ω_f. The latter sub-domain requiring full-order computations may then have a much coarser mesh, thereby improving the efficiency of the method. This refinement of the ROM method may be pursued in the future.

C. Implementation details

The ROM solution consists of several interfacing programs – all of them use open-source software. The mesh is generated using Gmsh[29], automated by a Python script. The full-order 'learning' datasets are obtained using SU2[30,31], the process being automated with Python. The derivation of the POD modes from the learning solutions is executed in Python.

The ROM solver is implemented in Python. In particular, the optimization is performed using the sequential least-squares quadratic programming (SLSQP) method available in the `scipy` library. The full-order solution on the partial domain is obtained from SU2. Both solvers are driven by a Python code that performs the iterative solution detailed in fig. 3. A custom boundary condition, as required at the interface of the two sub-domains, was implemented in SU2.

Although not discussed here, a 3-DOF rigid-body dynamics solver for the store motion has also been implemented in Python. The changes in position and orientation of the store relative to the wing require re-meshing – this is commanded to the automated mesh generating Python code mentioned above. An overarching Python code interfaces with the 3-DOF dynamic solver and the domain-decomposed ROM solver to calculate the trajectory of the store.

IV. Results

A. Geometry and database generation

We present some results of the application of the DD-ROM methodology for two-body aerodynamics in two dimensions. A NACA0012 airfoil section models the wing. The store is a slender body with a rounded tip and pointed tail (see fig. 5c). Its length is 80% of the wing chord, and its width is 10% of its length. The tip consists of a rounded nose of radius equal to 50% of the width, followed by a tangent ogive to give an

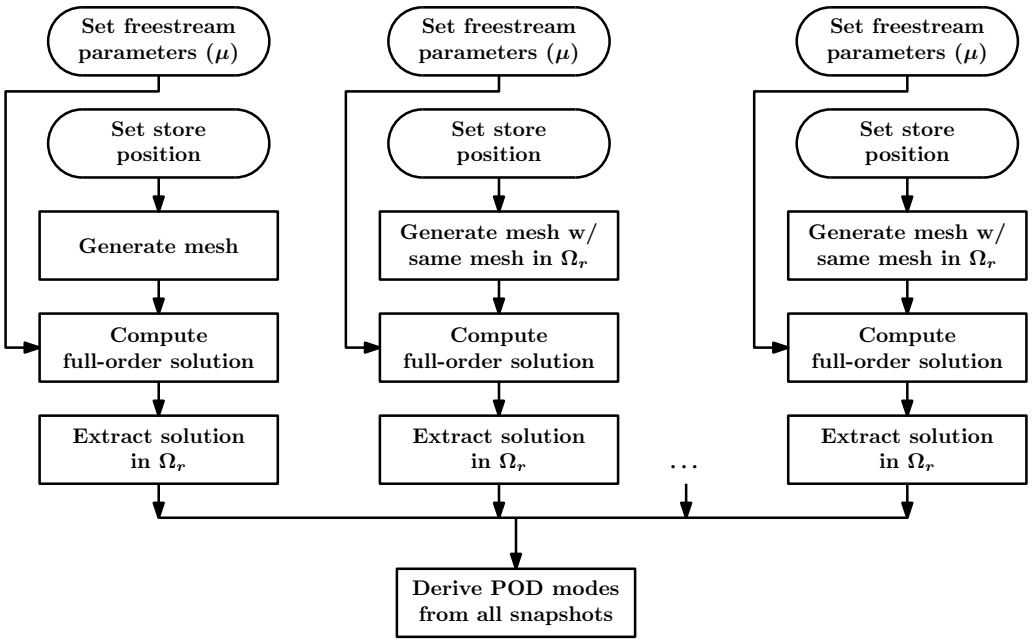


Figure 4: Flowchart of steps for deriving POD modes in domain-decomposed problem.

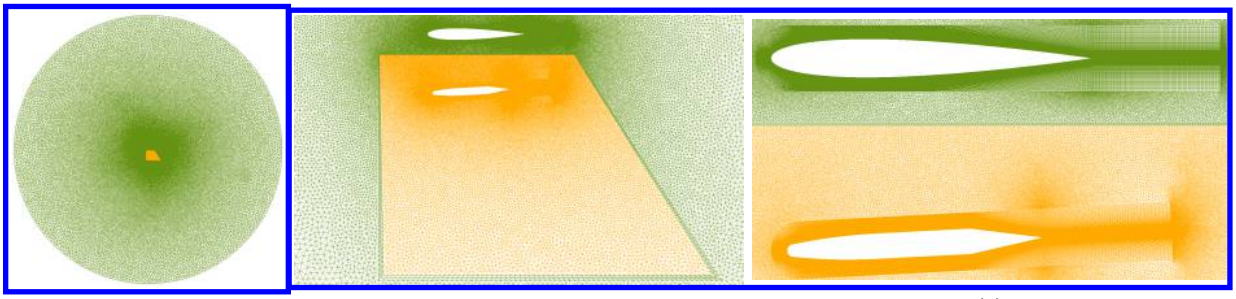


Figure 5: The domain-decomposed mesh for the DD-ROM of wing-store aerodynamics in 2D. (a) Overall mesh. (b) Zoomed view of mesh around wing and store, with structured overlap region. (c) Further zoom of body-fitted structured mesh.

overall fineness ratio of 3.5 (nose length to width). The tail half-angle is 10°.

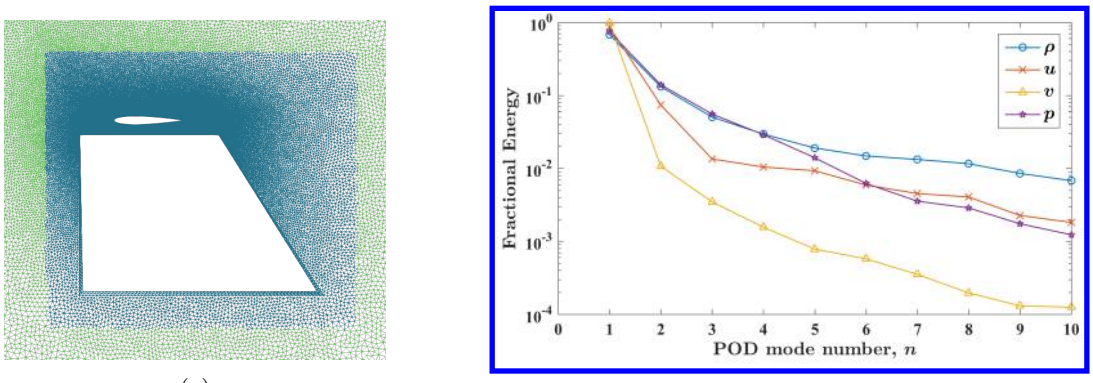
The domain-decomposed mesh developed for this problem is depicted in fig. 5. The far-field boundary is a circle centred on the wing leading edge and its radius is thirty times the wing chord. A trapezoidal sub-domain where the store can potentially go (and still be influenced by the wing) is meshed separately, as shown in fig. 5b. The overlap region is a structured grid consisting of two layers of triangular cells. A structured body-fitted mesh is created around both the wing and the store to adequately resolve the boundary layer (see fig. 5c). Note that the entire mesh, and the mesh in the sub-domain in particular, is generated by a Python program wrapping Gmsh[29]. This ensures that the mesh for the remaining domain (and the overlap region) is identical in all respects (including node and cell numbering) for various positions and orientations of the store. The number of grid nodes in the overall domain is about 220,000; that in the sub-domain surrounding the store is 77,000.

The compressible Reynolds-averaged Navier-Stokes equations are solved with the Spalart-Almaras turbulent model. The details of the numerical method are identical to those used for the RAE 2822 transonic airfoil simulation reported in Ref. 31. The freestream conditions are kept subsonic in the cases considered in this paper.

The learning database parameters are given in table 1. In particular, we use two freestream Mach

Parameter	Values
Mach number	0.45, 0.55
Angle of attack of wing, α	0°, $\pm2^\circ$, $\pm4^\circ$, $\pm6^\circ$
Orientation of store relative to wing, γ	$\pm5^\circ$
$x-$coordinate of store tip relative to wing leading edge	0, ±0.1
$y-$coordinate of store tip relative to wing leading edge	-0.5, -0.75, -1.0

Table 1: Parameter value combinations used for developing 'learning' database of 252 snapshots. The last two parameters are normalized by the wing chord.



(a) (b)

Figure 6: (a) Sub-region of interest where the POD inner product is defined. (b) Scalar POD eigenspectra for the four variables in terms of the fractional 'energy', i.e., $\lambda_n / \sum_m \lambda_m$ with λ_n being the nth POD eigenvalue.

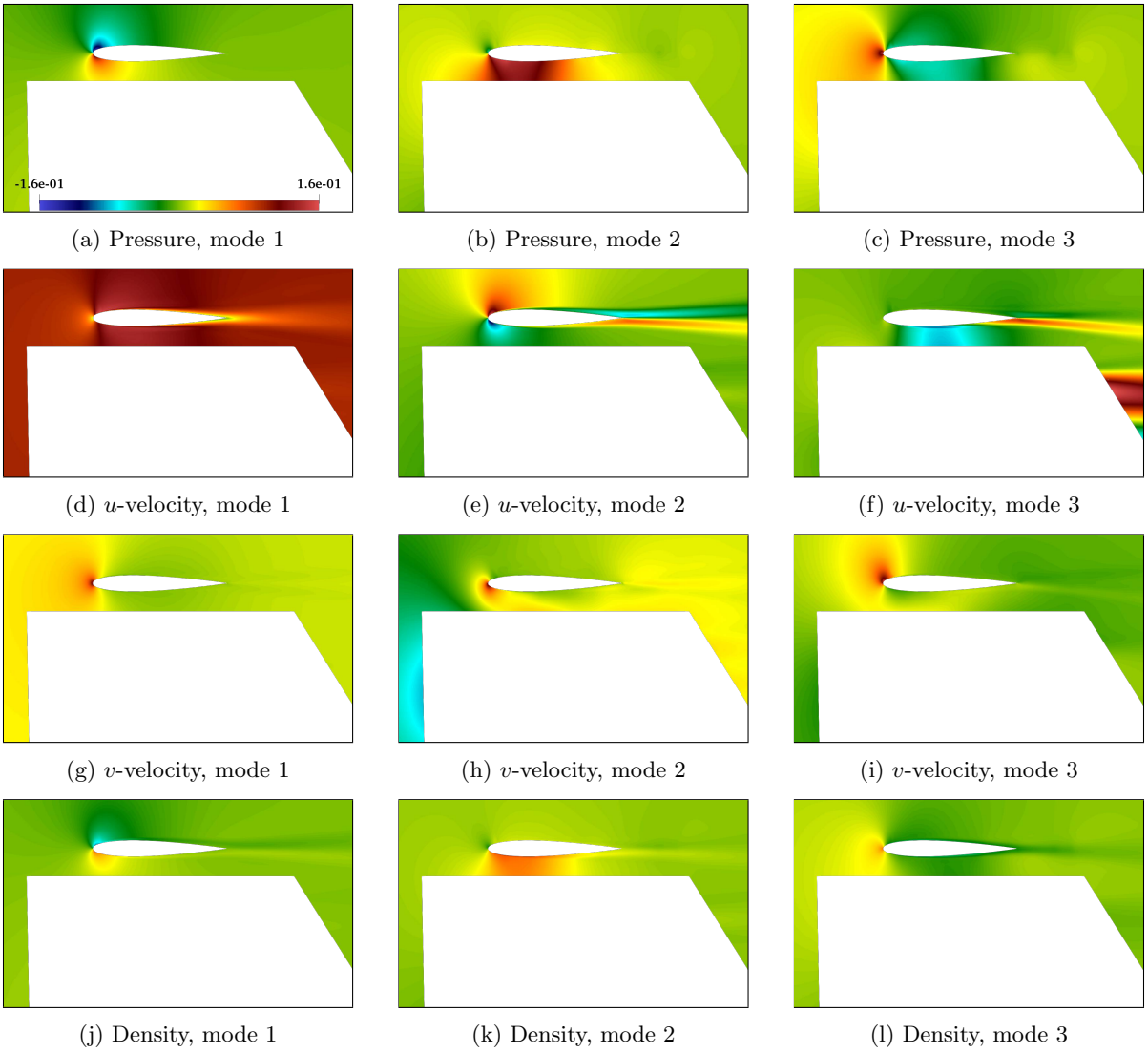
numbers, seven angles of attack of the wing, two orientations of the store relative to the wing, and three each of $x-$ and $y-$coordinates of the tip of the store relative to the leading edge of the wing. This results in a database of 252 snapshots.

B. POD

POD modes are calculated for the majority of the flow domain except for the small trapezoidal sub-domain where the store is expected to be located, as shown in fig. 5b and again in fig. 6a. As discussed in Ref. 24, the inner product considered for POD need not be defined over the entire POD domain. Instead, it is computationally expedient and physically acceptable to restrict attention to the immediate vicinity of the solid bodies in the flow. The region of interest chosen for the inner product definition is shown in fig. 6a. The POD eigenfunctions that result from this are still defined for the entire flow domain (except the trapezoidal hole).

The eigenvalues resulting from the scalar POD approach applied to the mean-subtracted 'learning database' are shown in fig. 6b. The extremely rapid decay of 'energy' or importance of the higher POD modes is apparent. In fact, the first POD modes of ρ, u, v and p respectively account for 68%, 87%, 98% and 74% of their overall energy. Also, for all four variables, the tenth mode is at least two-orders of magnitude lower in energy compared to the first.

An analysis of the POD eigenfunctions depicted in fig. 7 elucidates the observation made regarding the eigenvalues. Once we subtract the mean, the density and pressure fluctuations essentially vanish at the far-field boundary for all snapshots. However, this is not the case for the velocity components. For our database, the mean Mach numbers corresponding to u and v at the boundary are 0.5 and 0.0, respectively. However, the snapshots have varying fluctuation values at the boundaries for these variables, which also apply over almost the entire flow domain except in the immediate vicinity of the solid bodies. The first POD mode captures this important variation mode. The first mode of u is overwhelmingly of one sign (positive in the chosen normalization). The first mode of v is also biased towards positive values, although this is much



(a) Pressure, mode 1 (b) Pressure, mode 2 (c) Pressure, mode 3

(d) u-velocity, mode 1 (e) u-velocity, mode 2 (f) u-velocity, mode 3

(g) v-velocity, mode 1 (h) v-velocity, mode 2 (i) v-velocity, mode 3

(j) Density, mode 1 (k) Density, mode 2 (l) Density, mode 3

Figure 7: First three POD eigenfunctions of each of the four variables.

less apparent from the figure. Being relatively uniform over the majority of the flow domain, they represent very significant fractions of the respective energies.

The other POD modes account for the remainder of the variations observed across the snapshots. As discussed in the literature (e.g., Ref. 17), it is not always possible to directly relate the higher POD eigenfunctions to any single observed flow structure – thus, we would not undertake an analysis of the details of these modes further.

C. DD-ROM

The DD-ROM was set the goal of predicting the flow in several 'validation' cases, as listed in table 2. We note that none of the $M - \alpha$ combinations were a part of the learning database presented in table 1. The ROM was based on the first five density and u POD modes, and the first four v and pressure POD modes, based on a study of the POD eigenspectra in fig. 6b.

Before discussing the results for the aerodynamic coefficients presented in the table, we study the distribution of pressure coefficient on the surface of the store in fig. 8. The DD-ROM solutions are compared with the corresponding full-order solutions for all the cases listed in table 1. In general, we note the correct reproduction of the trends in distribution of pressure coefficient c_p with changing orientation of the store

Parameter/ Result	Cases								
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
Mach no.	0.45	0.45	0.45	0.5	0.5	0.5	0.55	0.55	0.55
Wing AoA	3°	3°	3°	5°	5°	5°	1°	1°	1°
Store angle	-1°	0°	1°	1°	2°	3°	-3°	-4°	-5°
Store tip x	-0.05	-0.05	-0.05	0.05	0.05	0.05	-0.1	-0.1	-0.1
Store tip y	-0.8	-0.8	-0.8	-0.6	-0.6	-0.6	-0.7	-0.7	-0.7
c_l (truth)	0.25	0.18	0.11	0.22	0.12	0.01	0.27	0.34	0.41
c_l (DD-ROM)	0.30	0.24	0.18	0.19	0.12	0.06	0.44	0.47	0.50
c_d (truth)	0.01	0.01	0.01	0.02	0.02	0.01	0.05	0.06	0.03
c_d (DD-ROM)	0.02	0.01	0.01	0.01	0.01	0.01	0.02	0.02	0.07
c_m (truth)	-0.03	-0.02	-0.01	0.00	0.00	0.01	-0.06	-0.06	-0.06
c_m (DD-ROM)	-0.04	-0.03	-0.02	0.00	0.00	0.00	-0.03	-0.05	-0.07

Table 2: Parameter values used for 'validating' database (for details see table 1), and corresponding results for the store's aerodynamic coefficients from the full-order calculations (truth) and the DD-ROM.

(relative to the wing) across the columns. There is also encouraging quantitative agreement for all the cases. The friction coefficients are typically three orders of magnitude lower, and hence are not portrayed – the DD-ROM was unable to match these.

We now turn to the comparison of the aerodynamic coefficients in table 2. We note that the predictions of lift coefficient c_l are generally close to the truth, although appreciable deviations are observed in a few cases. The coefficient of drag c_d and moment CM are much smaller, and are reproduced by the DD-ROM quite accurately.

The average time taken for the full-order simulations was about 7 hours on a single processor (Intel® Core™ i7-4770 CPU with 3.40 GHz clock). On the other hand, the domain-decomposed ROM typically required four to five iterations between the two sub-domains to converge, taking about 1.6 hours on the same single processor.

V. Conclusion

This paper outlines a novel reduced-order model (ROM) approach to lessen the computational effort involved in safe separation certification of aircraft stores. The trajectory of stores in the vicinity of the aircraft has been previously shown to be predicted well using quasi-steady Euler or RANS simulations. The ROM replaces the Euler simulations in the entire computational domain with a significantly faster multi-variable optimization problem in the majority of the problem domain, coupled with Euler or RANS simulations in a limited region surrounding the moving store.

Results presented here demonstrate the applicability of the procedure to a 2D subsonic problem. Encouraging agreement is shown with full-order calculations of the pressure coefficient on the store in a range of relative positions and orientations with respect to the wing and freestream conditions. The computational saving is approximately a factor of four.

In the future, the proposed procedure may be extended to 3D, and to transonic and supersonic aerodynamic problems involving complicated geometries of both wing and store.

Acknowledgements: AS acknowledges funding from the Defence Research and Development Laboratory (DRDL) of Hyderabad, India, under contract number DRDL/24/42P/17/04531/42932/CMH-II. SG was supported by an IITB Research Internship Award, funded by the Industrial Research and Consultancy Center of Indian Institute of Technology Bombay.

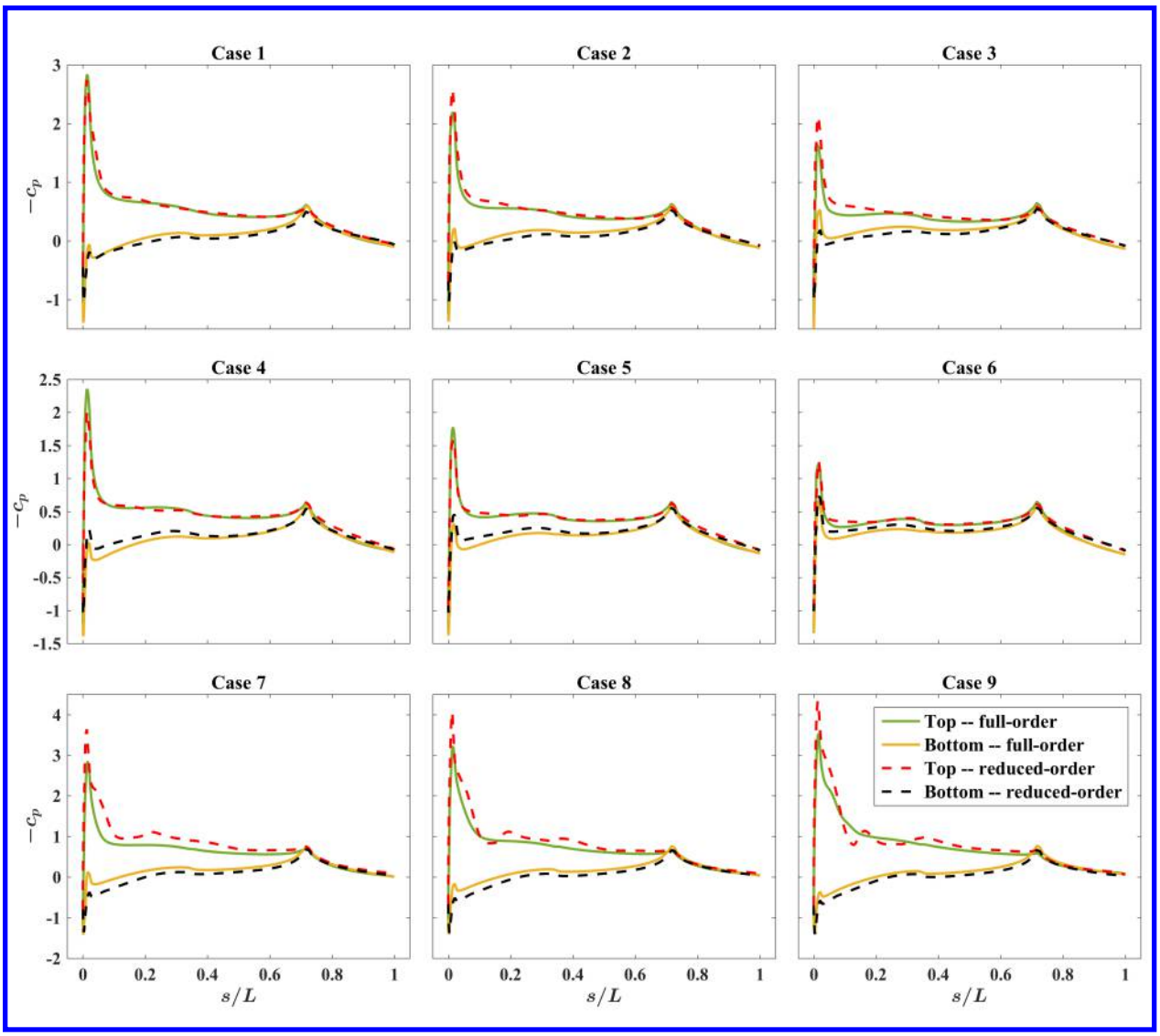


Figure 8: Pressure coefficient on top and bottom surfaces of store in the nine cases of table 2, s/L being the fractional distance along the length of the store from its tip. Each row has uniform ordinate axis.

References

[1] M. J. Bamber. Two methods of obtaining aircraft trajectories from wind tunnel investigations. Technical report, AERO Report 970 (AD 233198), 1960.

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

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

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

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

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

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

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

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

[10] A. Arabshahi and D. L. Whitfield. A multiblock approach to solving the three-dimensional unsteady Euler equations about a wing-pylon-store configuration. In *AIAA Atmospheric Flight Mechanics Conference, AIAA Paper 3401*, 1989.
[11] T. L. Donegan and J. H. Fox. Analysis of store trajectories from tactical fighter aircraft. In *29th AIAA Aerospace Sciences Meeting, AIAA Paper 183*, 1991.
[12] R. H. Nichols, J. L. Jacocks, and M. J. Rist. Calculation of the carriage loads of tandem stores on a fighter aircraft. In *30th AIAA Aerospace Sciences Meeting, AIAA Paper 283*, 1992.
[13] L. E. Lijewski and N. E. Suhs. Time-accurate computational fluid dynamics approach to transonic store separation trajectory prediction. *Journal of Aircraft*, 31(4):886–891, 1994.
[14] T. Berglind and L. Tysell. Numerical investigation of the impact of maneuver on store separation trajectories. In *28th AIAA Applied Aerodynamics Conference, AIAA Paper 4241*, 2010.
[15] T. Berglind and L. Tysell. Time-accurate CFD approach to numerical simulation of store separation trajectory prediction. In *29th AIAA Applied Aerodynamics Conference, AIAA Paper 3958*, 2011.
[16] W. S. Westmoreland. Trajectory variation due to an unsteady flow-field. In *47th AIAA Aerospace Sciences Meeting and Exhibit, AIAA Paper 550*, 2009.
[17] P. Holmes, J. L. Lumley, G. Berkooz, and C. W. Rowley. *Turbulence, coherent structures, dynamical systems and symmetry*. Cambridge University Press, 2012.
[18] N. Aubry, P. Holmes, J. L. Lumley, and E. Stone. The dynamics of coherent structures in the wall region of a turbulent boundary layer. *Journal of Fluid Mechanics*, 192:115–173, 1988.
[19] B. R. Noack, K. Afanasiev, M. Morzynski, G. Tadmor, and F. Thiele. A hierarchy of low-dimensional models for the transient and post-transient cylinder wake. *Journal of Fluid Mechanics*, 497:335–363, 2003.
[20] E. Caraballo, J. Little, M. Debiasi, and M. Samimy. Development and implementation of an experimental-based reduced-order model for feedback control of subsonic cavity flows. *ASME Journal of Fluids Enginnering*, 129(7):813–824, 2007.
[21] A. Sinha, A. Serrani, and M. Samimy. Initial development of reduced-order models for feedback control of axisymmetric jets. *International Journal of Flow Control*, 2(1):39–60, 2010.
[22] T. Bui-Thanh, M. Damodaran, and K. E. Willcox. Proper orthogonal decomposition extensions for parametric applications in compressible aerodynamics. In *21st Applied Aerodynamics Conference, AIAA Paper 4213*, 2003.
[23] P. A. LeGresley and J. J. Alonso. Investigation of non-linear projection for pod based reduced order models for aerodynamics. In *39th Aerospace Sciences Meeting, AIAA Paper 926*, 2001.
[24] D. Alonso, J. M. Vega, and A. Velazquez. Reduced-order model for viscous aerodynamic flow past an airfoil. *AIAA Journal*, 48(9):1946–1958, 2010.
[25] D. Alonso, J. M. Vega, A. Velázquez, and V. de Pablo. Reduced-order modeling of three-dimensional external aerodynamic flows. *Journal of Aerospace Engineering*, 25(4):588–599, 2012.
[26] R. Zimmermann and S. Görtz. Non-linear reduced order models for steady aerodynamics. *Procedia Computer Science*, 1 (1):165–174, 2010.
[27] J. L. Lumley. The structure of inhomogeneous turbulent flows. In A. M. Yaglom and V. I. Tatarsky, editors, *Atm. Turb. and Radio Wave Prop.*, pages 166–178. Nauka, Moscow, 1967.
[28] L. Sirovich. Turbulence and the dynamics of coherent structures, Parts I-III. *Quarterly of Applied Mathematics*, XLV(3): 561–590, 1987.
[29] C. Geuzaine and J.-F. Remacle. Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. *Int. J. Numer. Meth. Enrng.*, 79(11):1309–1331, 2009.
[30] F. Palacios, M. R. Colonno, A. C. Aranake, A. Campos, S. R. Copeland, T. D. Economon, A. K. Lonkar, T. W. Lukaczyk, T. W. R. Taylor, and J. J. Alonso. Stanford university unstructured (su2): An open-source integrated computational environment for multi-physics simulation and design. In *51st AIAA Aerospace Sciences Meeting, AIAA Paper 0287*, 2013.
[31] F. Palacios, T. D. Economon, A. C. Aranake, S. R. Copeland, A. K. Lonkar, T. W. Lukaczyk, D. E. Manosalvas, K. R. Naik, A. S. Padrón, B. Tracey, A. Variyar, and J. J. Alonso. Stanford university unstructured (su2): Open-source analysis and design technology for turbulent flows. In *52nd AIAA Aerospace Sciences Meeting, AIAA Paper 0243*, 2014.