Reduced Order Modelling for Aeroelastic Simulations

Billton Joseph VITUS
Institut Supérieur de l'Aéronautique et de l'espace

Joseph MORLIER
Institut Supérieur de l'Aéronautique et de l'espace, Institut Clément Ader

Oriol CHANDRE-VILA
Institut Supérieur de l'Aéronautique et de l'espace, AIRBUS

The project aims to build a non intrusive reduced order solver, based in Python, for performing Aeroelasticity simulations. A snapshot matrix is created comprising of high fidelity solutions at different parametric conditions, which shall be used as the offline training data to create the reduced order solver. A principal component analysis, namely Proper Orthogonal Decomposition is performed over the snapshot matrix to estimate the dominant modes. Radial Basis Function is used as the regression/prediction algorithm in the online phase. A test case of a 2D advection-diffusion problem is demonstrated to understand the construction and working of a reduced order model. Massive improvements in simulation times are observed (200 times faster) with minimal compromises on the accuracy (4% relative error).

I. Nomenclature

ROM	=	Reduced Order Model
PCA	=	Principal Component Analysis
RBF	=	Radial Basis Function
POD	=	Proper Orthogonal Decomposition
SVD	=	Singular Value Decomposition
FEM	=	Finite Element Method
CFD	=	Computational Fluid Dynamics
$\Phi(\|\| \cdot \|\|)$	=	Radial Basis Functions
ϕ_k	=	POD Vectors
VLM	=	Vortex Lattice Method
v	=	Convection speed
κ	=	Coefficient of Diffusion
$\bar{y}$	=	Parameterised Left Boundary Condition

II. Introduction

The need for fast and accurate computations in the scientific field has been at an all time high. In the aerospace industry wind tunnel tests are very expensive and involves a lot of complexity associated to recording data at every point of interest. In such a situation, CFD simulations are preferred. The simulation times for estimating the aeroelastic responses is way too high in preliminary parametric-space exploration. Thus the need of the hour is to obtain fast solutions that are accurate enough for the preliminary design phase. The aim of this study is to create a Reduced Order Model (ROM) based solver for the aeroelasticity problem, to get solutions as fast as possible, without a major compromise on the accuracy. The main principle is building the solver that is data driven, in contrast to deriving it using conservation laws. This requires a large dataset of simulation/ experimental results on which the ROM shall be built upon.

In a previous project Chandre-Vila *et al.* explore the usage of reduced order modelling for an aeroelastic system [1]. The study consists of two phases, 1) offline phase and 2) online phase. Another study by the same authors explored the usage of different techniques of ROM creation on the convection-diffusion equation on a 2-D plate of set dimensions and boundary conditions [2]. Multiple parameters are set to be varied between a lower and upper bound of values, and the ROM was tested for its validity within those bounds. This case study is based on an online course on Model Reduction, offered in Stanford University by Prof. C. Farhat and Prof. D. Amsallem [3]. The test case is based on this work of ROM creation. Another work in the same domain by Yaduo Li compared multiple methodologies used for creating a ROM [4]. This study concluded that a non intrusive process like the Similarity Estimation Criterion with Tensor Flow (TF) POD showed the lowest offline training time, with the RMS errors ranging between 10-180K depending on the input parameters used for a 2D heat equation. The other methods investigated were the TF POD with Greedy Algorithm and TF POD with Gaussian Process. The RMS errors for these two processes were similar, in the range of 4-5K for the 2D heat equation, whereas the offline time was different since GP does not have to account the extra time for the Greedy Algorithm to run.

Several studies have been performed illustrating the usage of ROMs in the fields of structural mechanics, fluid mechanics and control [5] [6] [7]. All these ROMs utilise the modification of the physics problem or the source code in some way or the other. There are several situations where the source codes are not available or are too complex to undergo modification. This is where non-intrusive ROMs help the most. This methodology projects the solution in a reduced basis estimated using PCA, and thus the solution is represented as a vector of reduced coefficients. Finally, a regression model is used to map the reduced coefficients with the parameter values [8]. The regression model used in this study is the Radial Basis Function [9]. There are many papers that take advantage of the non-intrusive ROM methodology, especially in the field of fluid flows [8] [10] [11].

As for aeroelastic appliations, ROMs have a well established rooting in this domain. A study by Thomas et al. explores the usage of POD for ROM generation [12]. The study had novel results and concluded that while creating the snapshot matrix, the solution snapshots must include actual structural mode snapshots at the endpoints of the frequency range of interest. The other entries in the snapshot matrix need not resemble the actual mode shapes. Rather, the first stage of creating the snapshot matrix can be done without using structural mode shapes under consideration. A study by Lieu et al. investigates the ROM generation methods at varying flight conditions, like mach number and angle of attack [13]. The authors do so by establishing a methodology to interpolate between solution snapshots matrices captured in different flight conditions. A snapshot matrix is not created from two or more different flight conditions as it reduces the convergence properties of the individual flight conditions used in the matrix. This method is applied to an F-16 aircraft in different flow streams and the results are verified using full scale non linear simulations and flight data. A method frequently used by control engineers, the realization method shows good results for the functioning of the reduced model for an unstable aeroelastic system. A study by Hesse and colleagues shows the implementation of the ROM using the linearization method on a flexible HALE aircraft [14]. The model is first tested on a Goland wing configuration.
A few studies have looked into the combination of different ROM formulation techniques for accurate modelling of a reduced order system. A study by Lucia et al. uses POD combined with Volterra theory, to create a ROM. The main idea behind this coupling is to replace generalized aerodynamic forces with physics based ROM, that is capable of producing non linear aeroelastic forces [15]. Studies by Xie and colleagues and Nikbay et al. show the comparison of POD with Galerkin method and non-intrusive Polynomial Chaos expansion respectively. The studies showed similar findings with POD being the superior method, with lower errors and lesser computation time for simulations compared to other [16] [17].

The report is structured such that the complete methodology is explained in Section III, presenting an approach and framework that is different from the works done previously by the research group. Section IV discussed the results obtained with the new methodology and have been compared with previously known methods. Section V concludes the findings of the study and discusses the future scope of this project.

III. Methodology

This section deals with the methodology used in the offline and the online phases of the ROM. It is important to understand the steps involved in creation of a non-intrusive POD-RBF ROM.

A. Creation of Snapshot Matrix

The first step of creating a ROM is to create the training data. This is done in the form of a snapshot matrix, **A**. The snapshot matrix consists of n solution vectors a that have been generated by n sets of parameters μ. Parameter vector μ consists of design parameters that are set to vary as an input condition. These design parameters can be anything- like the physical dimension, equation coefficients etc. These parameters have a defined upper and lower bound limit within which these parameters are randomly selected from. The choice of random selection is Latin Hypercube Sampling. These generated points are unique in nature and do not have a repeating value in the selection of a particular μ [18]. This ensures that the parameters selected from within the bounds have a Gaussian distribution, which ensures all points in the design space have been equally represented in the snapshot matrix. A visual representation of the procedure has been shown in Figure 1.

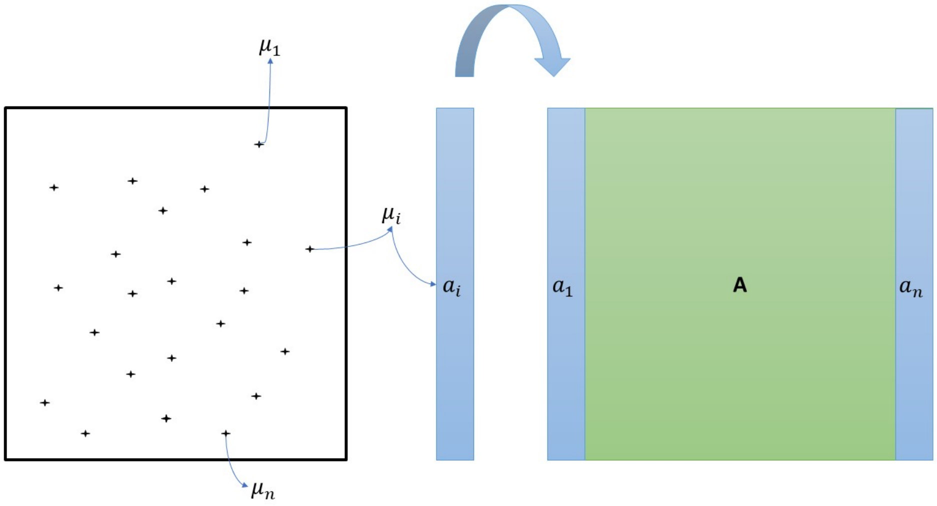


Fig. 1 A general representation of the snapshot matrix formation. parameter points from the design space in the left are chosen to perform high fidelity simulations and the solution vectors are stacked to form a matrix.

B. Proper Orthogonal Decomposition

The next step is to estimate the reduced basis functions for model order reduction. The choice of methodology followed for this is the proper orthogonal decomposition [19]. The operation is applied to the snapshot matrix **A**, which gives

$$POD(A)_{M\times N} = \mathbf{U}_{M\times M} \times \Sigma_{M\times N} \times \mathbf{V}^T_{N\times N} \tag{1}$$

where

- **U** is the left singular matrix.
- Σ is the diagonal matrix consisting of $min(M,N)$ singular values.
- **V** is the right singular matrix.

The assumption made for the following parts is that M is greater than N, i.e. the DOF of the problem is more than the number of snapshots collected. The **V** matrix in Equation (1) is unitary in nature, thus the equation can be re written in the form

$$AV = U\Sigma \tag{2}$$

This equation can we written for the i^{th} column of **U** and **V** matrix individually as

$$Av_i = \sigma_i u_i \; ; \; i \in 1,2,3,\dots N \tag{3}$$

The POD vector associated to the eigen values σ_k of **A** is given by

$$\phi_k = \frac{Au_k}{\sqrt{\sigma_k}} \; ; \; k \in 1,2,\dots N \tag{4}$$

The singular values σ signify the energy contained in each singular vector, i.e. the columns of the **U** matrix. The reduced basis is formed by the truncation of the **U** matrix up to the columns that capture about a threshold of energy, denoted by ϕ_r, where r is the number of columns after truncation. The energy is not a physical quantity, but a representation of the total dynamics of the problem captured in it. The equation to represent the total captured energy after truncation is given by

$$E_{tot} = \frac{\sum_{i=1}^{r} \sigma_i}{\sum_{i=1}^{N} \sigma_i} \;;\;\; N > r \tag{5}$$

The accuracy of the reduction is dependent on the total energy value, the closer it is to 1, the more accurate it is. Using these POD vectors, the output variable can be considered as a summation of the product of the POD vectors and a coefficient α, given by

$$X = \sum_{i=1}^{r} \alpha_i \phi_i \tag{6}$$

The coefficient in Equation (6), α is the coefficient of POD expansion. The estimation of this coefficient is done through Radial Basis Function (RBF) and that shall be explained in the next section.

C. Radial Basis Function

The POD expansion coefficients are calculated based on a linear combination of K radial basis functions, given by

$$\alpha_i = \sum_{1}^{N} w_i \times \Phi(||\alpha_i - \hat{\alpha_j}||) \tag{7}$$

Here $\hat{\alpha_j}$ is the j^{th} centre point in the design space created by the RBF algorithm and w_i is the weight of the radial basis function $\Phi(|| \cdot ||)$. $|| \cdot ||$ is the Euclidean norm of the two points. There are several choices for $\Phi(|| \cdot ||)$, which are shown below in Table 1. The confusion arises in the selection of the w_i parameter. These weights cannot be randomly assigned since they determine the influence of each data point from the assigned centers. The method is to evaluate the RBF interpolation the known data points, i.e. in the training data. Thus the unknown variable in the left hand side becomes a known value, $y(k)$ and thus it forms a linear system of equations of the form

Function	Definition
Gaussian	$\Phi(r) = e^{-(\frac{r}{\sigma})^2}$
Linear Spline	$\Phi(r) = r$
Multi-Quadratic	$\Phi(r) = \sqrt{r^2 + \sigma^2}$
Inverse Multistory	$\Phi(r) = \frac{1}{\sqrt{r^2+\sigma^2}}$
Inverse Cadric	$\Phi(r) = \frac{1}{\sigma^2+r^2}$
Cubic Spline	$\Phi(r) = r^3$
Thin Plate Spline	$\Phi(r) = r^2 \log(r)$

Table 1 Most commonly used RBF Functions. r is the distance and σ is a constant dependent on the shape.

$$Y = W\Phi \tag{8}$$

, where

$$Y = \begin{bmatrix} y(1) & y(2) & \dots & y(n) \end{bmatrix}^T \tag{9}$$

$$W = \begin{bmatrix} w_i(1) & w_i(2) & \dots & w_i(n) \end{bmatrix}^T \tag{10}$$

$$and \;\; \Phi = \begin{bmatrix} \phi(||\alpha_1 - \hat{\alpha_1}||_2) & \phi(||\alpha_1 - \hat{\alpha_2}||_2) & \dots & \phi(||\alpha_1 - \hat{\alpha_n}||_2) \\ \phi(||\alpha_2 - \hat{\alpha_1}||_2) & \phi(||\alpha_2 - \hat{\alpha_2}||_2) & \dots & \phi(||\alpha_2 - \hat{\alpha_1}||_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi(||\alpha_n - \hat{\alpha_1}||_2) & \phi(||\alpha_n - \hat{\alpha_1}||_2) & \dots & \phi(||\alpha_n - \hat{\alpha_n}||_2) \end{bmatrix} \tag{11}$$

The choice of RBF Φ is multi-quadratic since it makes sure that Φ matrix is symmetric and non-singular [20]. The choice of basis function is more accurate than some of the other choices available [21][22]. This makes finding the weights used in the expansion coefficient α_i an inversion problem of known quantities, of the form

$$W = \Phi^{-1} Y \tag{12}$$

Thus with all the available data, for new sets of input parameters, new expansion coefficients are calculated and then projected onto the real basis using Equation (6). This entire methodology of the POD-RBF ROM is implemented into the code using the EZyRB Python library [23].

D. ROM Methodology Applied to a 2-D Advection-Diffusion Equation

The methodology described in the previous sub-sections have been applied to a 1-equation 2-D Heat Equation. This test case serves as a verification and validation process before applying the steps to an aeroelastic problem. The domain of choice is a square of dimensions $1m \times 1m$ with a parameterised left wall.
The governing equation of concern for this problem is

$$v \cdot \nabla T - \kappa \cdot \Delta T = 0 \text{ for } x \in \Omega \tag{13}$$

The design parameters in this problem that have been set to vary are the

- Advection speed in the x-axis, v
- Diffusion coefficient, κ
- Parametrised left wall boundary condition, $\bar{y}$

The physical significance of these parameters dictate the spread of the temperature across the domain - v dominates the spread in the $x - axis$, a high value of κ ensures a high diffusion of tempreature from a region of high temperature to a region of low temperature. The parameterised boundary condition $\bar{y}$ governs the position of the heat source on the left wall. The boundary conditions are set up as:

$$T(x, t; \mu) = T_D(x, t; \mu_D) \text{ for } x \in \Gamma_D \tag{14}$$

$$\nabla T(x, t; \mu) \cdot n(x) = 0 \text{ for } x \in \Gamma_N \tag{15}$$

where

$$T_D = \begin{cases} 300 & \text{for } y \in [0, \frac{1}{3}] \\ 300 + 325 \cdot (sin(3\pi \cdot |y - \overline{y}| + 1) & \text{for } y \in [\frac{1}{3}, \frac{2}{3}] \\ 300 & \text{for } y \in [\frac{2}{3}, 1] \end{cases} \tag{16}$$

A visual representation of the boundary conditions and the geometry in shown in the Figure 2. Γ_D represents a Neumann boundary condition and Γ_N stands for a Dirichlet boundary condition. The 3 parameters mentioned before have been varied between the minimum and maximum bounds set for the problem shown in the Table 2.

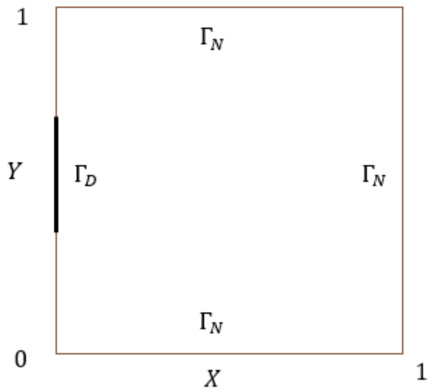


Fig. 2 A visual representation of the problem setup.

It is evident that the LHS sampling was done within these bounds of values and were sampled multiple times for the creation of the snapshot matrix.

Parameter	Minimum	Maximum
μ (ms^{-1})	0	0.5
κ (ms^{-2})	0.005	0.025
$\bar{y}$	0.4	0.6

Table 2 Parametric design variables in the 2-D Heat Equation.

E. ROM Methodology Apllied to a 2 - Equation Aeroelasticity problem

The following section outlines the application of the ROM creation on a static aeroelasticity problem. This section is based on the works of Oriol [1]. The general outline of the problem is based on the coupling of the aerodynamics aspect and the structural aspect to solve for the displacements. The two equations used are -
Aerodynamics:

$$A(\xi) \cdot \tau(\xi) = b(\xi) \tag{17}$$

Structures:

$$K(\xi) \cdot U(\xi) = F(\xi) \tag{18}$$

The aerodynamics equation is solved using a Vortex Lattice Method (VLM) solver, with $\mathbf{A}(\xi)$ matrix being the AIC matrix, $\tau(\xi)$ as the circulation vector and $b(\xi)$ is the RHS vector in the VLM equation, for a set of parameters ξ. The equation governing the structural part is dependent on the stiffness matrix, $\mathbf{K}(\xi)$, forces acting on the structure, $F(\xi)$ vector, and the displacement vector $U(\xi)$. These two equations are coupled in a way that the aerodynamic forces induce a displacement on the wing, which changes the shape of the wing, thus augmenting the force applied on it and so on. The estimation of the structural nodal force (F_S) using interpolation of the aerodynamic forces (F_A) on the structural node points is done using an interpolation matrix, which links the aerodynamic mesh to the structural mesh, denoted by $\mathbf{H}$, using the equation

$$F_S = H^{-1} \cdot F_A \tag{19}$$

The geometry of concern is a swept back wing as shown in Figure 3. Since there are many structural and aerodynamic parameter that can be varied, the choice of the design parameters comes down to 6 which are the -

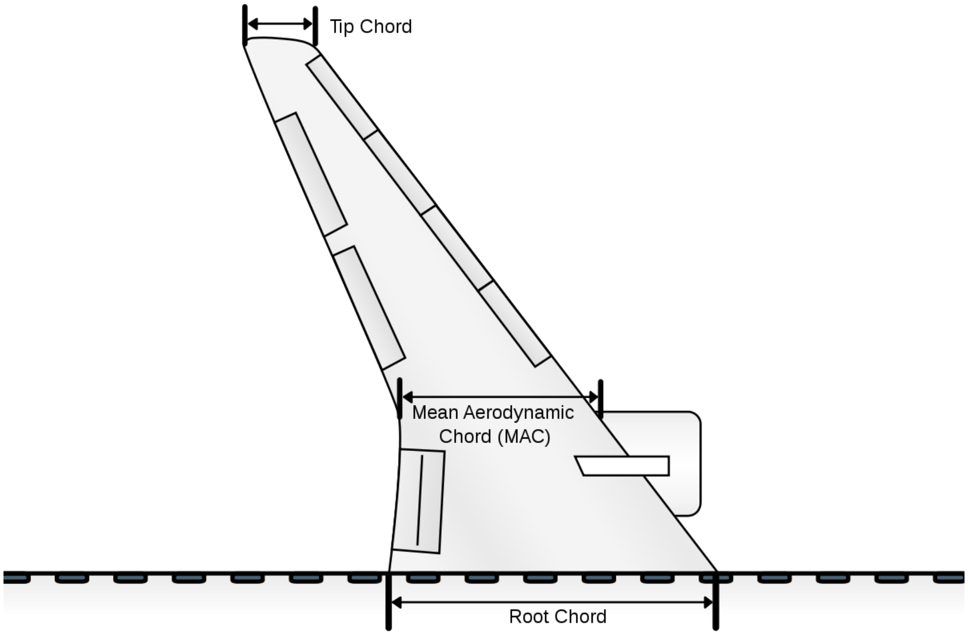


Fig. 3 A swept back wing used for the the aeroelasticity problem. [24]

- Skin thickness in *mm*
- Rib thickness in *mm*
- Leading edge spar thickness in *mm*
- Trailing edge spar thickness in *mm*
- Wing Span in *m*
- Wing Surface Area in m^2

These 6 parameters have been selected within a range of fixed values for sampling and are shown in the Table 3. The lower limits of the sizing criteria matching the dimensions of an A320 wing and the upper limit matching the A380 wing. The other parameters like angle of attack, mach number, Young's modulus, Poisson's ratio wing positioning parameters, dihedral angle, number of ribs and altitude, have all been set to a constant value for this problem.

Parameter	Minimum	Maximum
Skin Thickness (mm)	1.45	30
Rib Thickness (mm)	1.45	15
LE Spar Thickness (mm)	4.35	90
TE Spar Thickness (mm)	4.35	90
Wing Span (m)	34	80
Wing Area (m^2)	122	845

Table 3 Parametric design variables in the Aeroelasticity problem.

IV. Results and Discussion

This section deals with the associated results obtained from the ROM and possible understandings. The two cases have been individually discussed in their respective sections.

A. 2-D Heat Equation

The first task was to study the convergence of the solution with its dependence on the dimension of the snapshot matrix. This was done by increasing the dimension by adding full scale solutions vectors in steps of 25 and 100 vectors to the snapshot matrix. The time taken to generate these snapshot matrices were also recorded to see the increase in time on adding more columns to the training data. The time taken to generate the snapshot matrix is a part of the offline training time. The time variation for generating the snapshot matrix has been shown in Figure 4. As expected, the time taken to generate snapshot matrix is linear to the number of columns used. This is because the time taken to generate n snapshots should theoretically be n times the time taken to generate one snapshot. The slight deviation from it being a straight line is due to the variations in the computational resources available during the computation. To test the ROM and study the convergence, 80% of the generated parameters are used to create the snapshot matrix, and the rest of the 20% parameters are used as the input parameters to test the created ROM. The average root mean squared error for those test parameters is recorded for each chosen dimension. Thus this parameter is used to judge the independence of snapshot dimension from the solution. The root mean squared error, E_{RMS}, was calculated using the equation

$$E_{RMS} = \sqrt{\frac{\sum_1^N (T_{ROM} - \hat{T})^2}{N}} \tag{20}$$

Where N is the number of nodal points where the temperatures were recorded, T_{ROM} is the temperature value predicted by the ROM and $\hat{T}$ is the temperature obtained from the full scale simulation. The RMS error is recorded in Kelvin (K) since the temperature outputs are in the same unit. The estimation of error can be verified by predicting the temperature values using a set of parameters already used in creating the snapshot matrix. This yields an RMS error in the order of $10^{-12}K$. The error reaches such low values owing to the way python stores and calls data in matrices, and/or any rounding of decimals happening in the process. Apart from the accuracy of prediction, the time taken for the online phase and offline phase is of great importance to us, since the primary purpose of creating a ROM is to reduce the computation time. In an ideal sense, the offline training time consists of the time taken to the snapshot matrix and to set

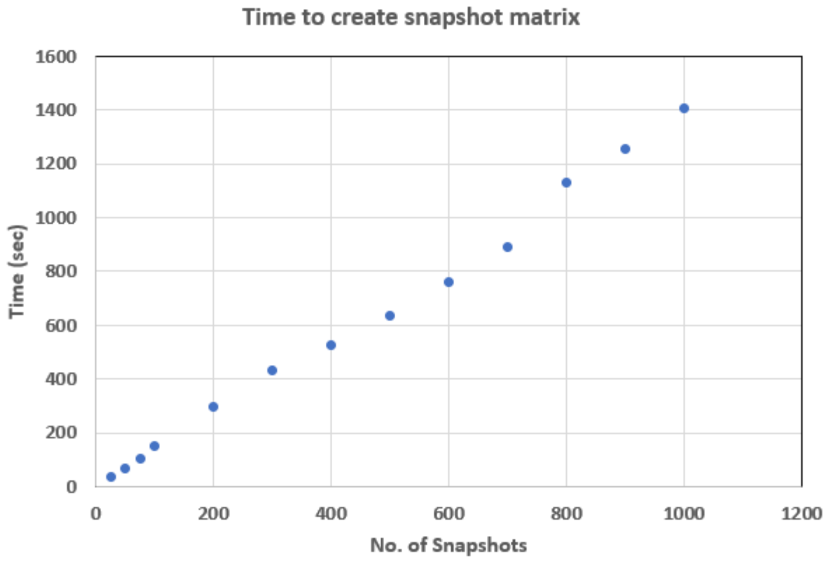


Fig. 4 Variation of the time taken to generate the snapshot data versus the number of columns.

up the interpolation/regression technique used in the online phase, i.e. train the data. But it was observed that the time taken to generate the snapshot matrix was much greater than the time taken to train the data, thus the two times have been shown separately. The speed up in computation time is the ratio of the time taken to compute the real solution and the online computation time, given by

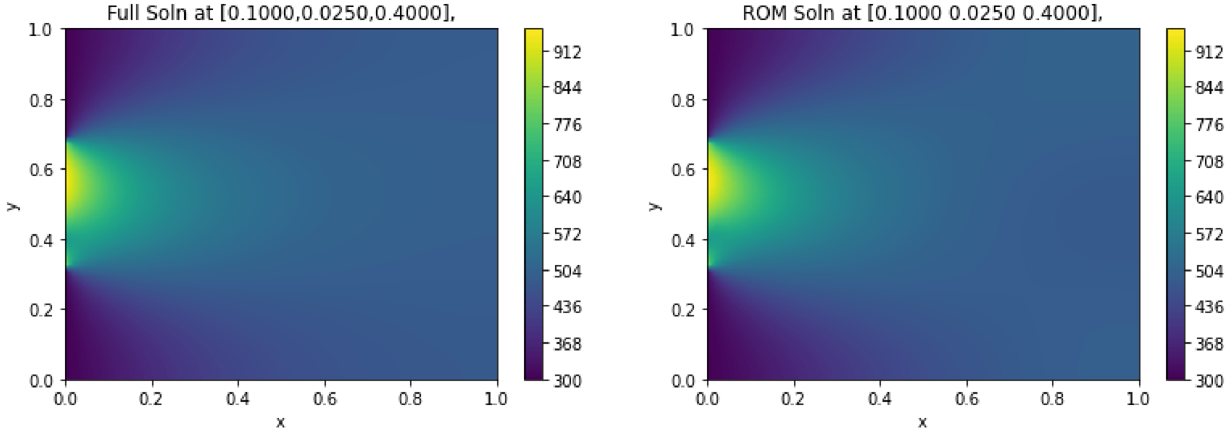
$$Speedup = \frac{Time_{Real}}{Time_{ROM}} \tag{21}$$

The results from this sensitivity study have been presented in Table 4. All the computations were performed for a fixed set of parameters that were not used in the training data. The full scale simulation for these set of parameters took an average of 1.32 sec, which is the baseline used for speedup in time.

No. of Columns	Snapshot Time (sec)	Training Time (sec)	Online Time (sec)	Average RMSE (K)	Speedup
20	36	0.014	0.002	15.229	713.361
40	68	0.031	0.004	9.1922	353.631
60	105	0.130	0.005	9.3693	265.799
80	151	0.308	0.005	7.7766	246.947
160	299	0.515	0.014	5.7632	95.698
240	444	0.917	0.024	6.0724	59.277
320	527	1.851	0.028	4.6645	62.454
400	637	7.549	0.027	4.6304	51.189
480	760	12.443	0.040	4.3107	34.276
560	889	20.097	0.051	3.5206	26.365
640	1133	31.669	0.058	3.0670	24.607
720	1256	43.503	0.071	3.0574	18.695
800	1409	62.700	0.072	2.8922	19.664

Table 4 Sensitivity study performed to isolate the dependence of snapshot matrix dimension to the solution accuracy.

There is a clear decrease in the average RMSE on increase in the snapshot matrix, and then it stabilises near a value of $3K$. Prioritising minimal loss of accuracy, with a maximum gain in computation time, the dimension of snapshot matrix was fixed at 80 snapshots for the rest of the study.
A random parameter set was selected from within the bounds to test the ROM. The selected values for [ν, κ, $\bar{y}$] was [0.11, 0.025 0.4]. It was made sure that this parameter set was not used in the training data. The plots show a large degree of similarity in the temperature outputs as shown in Figure 5.


Full Soln at [0.1000,0.0250,0.4000],
ROM Soln at [0.1000 0.0250 0.4000],

(a) Full scale solution.

(b) ROM generated solution .

Fig. 5 Comparison of a) Full scale solution with b) ROM generated solution

We also look at the quantitative results of the ROM solution, given in the Table 5. The results from the plots have also been compared to the results obtained by works of Chandre-Vila and Li in the advection diffusion problem. These values show that the the RMSE is in the acceptable range since, relatively speaking, it is less than 5% of the lowest observable temperature and thus an improvement from the previous methodologies used. This shows a good working of the ROM in a single equation PDE problem.

Parameter	POD-Galerkin [1]	POD-GP [4]	POD-RBF
Online Time	0.6713 sec	0.06 sec	0.005 sec
Max Error	4.0192%	-	6.345%
Average Error	1.4385%	-	0.927%
RMSE	-	4.7461 K	8.5934 K
Reduction in Time	85%	90%	99.6%

Table 5 Properties of Interest of the ROM generated solution for the 2-D advection-diffusion equation.

B. Static Aeroelasticity

This section illustrates the efficiency of a ROM for a 2 equation static aeroelasticity problem. To show the effectiveness of ROM on aeroelastic simulations, some results have been shown from work previously done by the research group [1]. The ROM application was done on a swept back wing geometry, by varying the parameters given in Table 3. The offline training includes the time taken by the high fidelity simulations to create the snapshot matrix as well as the time taken to train the data. The ROM simulation takes 0.3 $secs$ to give an output, whereas the original code runs for around 60 $secs$. This makes the ROM 200× faster than the high fidelity model. These results have been represented in the table below.

Offline Time (sec)	Online Time (sec)	Full Scale Time (sec)	Speedup
27000	0.3	60	200

Table 6 Quantitative properties of the ROM solution for static aeroelasticity.

The plots for displacement (Figure 6) and circulation (Figure 7) were plotted for the wing model. Along with the mean values, the upper and lower limits of the variances were also plotted to show the range of accuracy the ROM is able to estimate. The aim of the ROM tuning is based on the fact that the ROM outputs must lie within the upper and lower limit of the variance from the mean value of the high definition model outputs. The "orig" plots correspond to the results obtained from the high dimensional simulations, "mean" results correspond to the mean solutions obtained from the ROM simulations and "variance" plots correspond to the upper and lower limit of tolerance. The maximum Von Mises stress and the location matched in the structural simulations to a good degree of accuracy. Errors were reported to be under max of 15%, with high errors at the lower end of the Von Mises stress output spectrum.

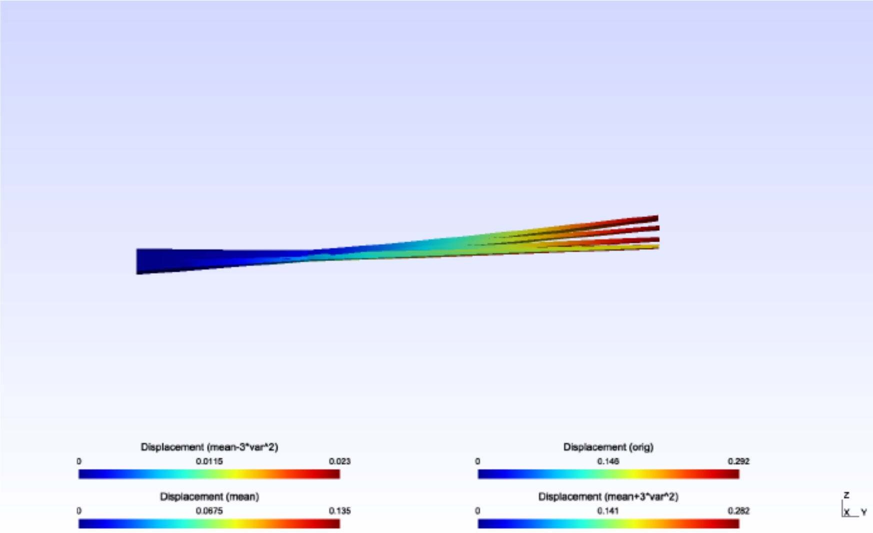


Fig. 6 Displacement profile of the wing.

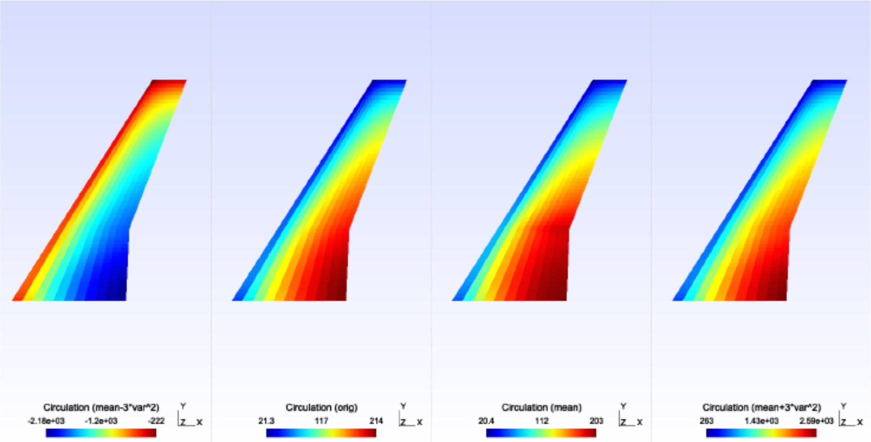


Fig. 7 Circulation contours over the wing surface.

V. Conclusion

The scope of the study was to build a non intrusive ROM for different engineering applications, especially pertaining to Aeroelastic applications. The 2-D heat equation application showed that the solution can be sped up by almost 250×, with minimal loss in accuracy. This established methodology allows a user to create a ROM, purely "Blackbox" in nature, from the training to the testing of a ROM. The future scope of this methodology would be to investigate the effects of the ROM on non-linear problems and verify the changes required in the snapshot matrix or interpolation techniques.

VI. Declaration of Authenticity

This assignment is entirely my own work. Quotations from literature are properly indicated with appropriated references in the text. All literature used in this piece of work is indicated in the bibliography placed at the end. I confirm that no sources have been used other than those stated. I understand that plagiarism (copy without mentioning the reference) is a serious examinations offence that may result in disciplinary action being taken.

References

[1] Chandre-Vila, O., Morlier, J., and Sylvain, D., "Model Reduction in Aeroelasticity for Preliminary Design," , 2019. URL https://github.com/mid2SUPAERO/POD-K_v2-CHANDRE.

[2] Chandre-Vila, O., Morlier, J., and Sylvain, D., "Structural Wing Model Reduction in Fluid-Structure Interactions," , 2018. URL https://github.com/mid2SUPAERO/PGD-ROM_CHANDRE.

[3] Farhat, C., and Amsallem, D., "CME345: Model Reduction," , 2017. URL https://web.stanford.edu/group/frg/course_work/CME345.html.

[4] Li, Y., and Morlier, J., "Parameterised PDE Prediction," , 2019. URL https://github.com/mid2SUPAERO/SFE_Yaduo_LI.

[5] Caicedo, M., Oliver, J., Huespe, A., and Lloberas-Valls, O., "Model Order Reduction in Computational Multiscale Fracture Mechanics," *Key Engineering Materials*, Vol. 713, 2016, p. 248–253. https://doi.org/10.4028/www.scientific.net/kem.713.248.

[6] Pathak, K., and Yamaleev, N., "POD-based Reduced-order Model for Arbitrary Mach Number Flows," *6th AIAA Theoretical Fluid Mechanics Conference*, 2011. https://doi.org/10.2514/6.2011-3111.

[7] Wang, W., "Robust adaptive control for systems with reduced-order model," *29th IEEE Conference on Decision and Control*, 1990. https://doi.org/10.1109/cdc.1990.203746.

[8] Bērziņš, A., Helmig, J., Key, F., and Elgeti, S., "Standardized Non-Intrusive Reduced Order Modeling Using Different Regression Models With Application to Complex Flow Problems," *arXiv*, 2020.

[9] Sharan, M., Kansa, E., and Gupta, S., "Application of the multiquadric method for numerical solution of elliptic partial differential equations," *Applied Mathematics and Computation*, Vol. 84, No. 2-3, 1997, p. 275–302. https://doi.org/10.1016/s0096-3003(96)00109-9.

[10] Xiao, D., Fang, F., Pain, C., and Hu, G., "Non-intrusive reduced-order modelling of the Navier-Stokes equations based on RBF interpolation," *International Journal for Numerical Methods in Fluids*, Vol. 79, No. 11, 2015, p. 580–595. https://doi.org/10.1002/fld.4066.

[11] Dutta, S., Farthing, M. W., Perracchione, E., Savant, G., and Putti, M., "A greedy non-intrusive reduced order model for shallow water equations," , 2020.

[12] Thomas, J. P., Dowell, E. H., and Hall, K. C., "Three-Dimensional Transonic Aeroelasticity Using Proper Orthogonal Decomposition-Based Reduced-Order Models," *Journal of Aircraft*, Vol. 40, No. 3, 2003, p. 544–551. https://doi.org/10.2514/2.3128.

[13] Lieu, T., and Farhat, C., "Adaptation of POD-based Aeroelastic ROMs for Varying Mach Number and Angle of Attack: Application to a Complete F-16 Configuration," *2005 U.S. Air Force T&E Days*, 2005. https://doi.org/10.2514/6.2005-7666.

[14] Hesse, H., and Palacios, R., "Reduced-Order Aeroelastic Models for Dynamics of Maneuvering Flexible Aircraft," *AIAA Journal*, Vol. 52, No. 8, 2014, p. 1717–1732. https://doi.org/10.2514/1.j052684.

[15] Lucia, D. J., Beran, P. S., and Silva, W. A., "Aeroelastic System Development Using Proper Orthogonal Decomposition and Volterra Theory," *Journal of Aircraft*, Vol. 42, No. 2, 2005, p. 509–518. https://doi.org/10.2514/1.2176.

[16] Xie, D., Xu, M., and Dowell, E. H., "Proper Orthogonal Decomposition Reduced-Order Model for Nonlinear Aeroelastic Oscillations," *AIAA Journal*, Vol. 52, No. 2, 2014, p. 229–241. https://doi.org/10.2514/1.j051989.

[17] Nikbay, M., and Acar, P., "Reduced order modelling for static and dynamic aeroelastic predictions with multidisciplinary approach," *CEAS Aeronautical Journal*, Vol. 6, No. 3, 2015, p. 455–469. https://doi.org/10.1007/s13272-015-0154-3.

[18] McKay, M. D., Beckman, R. J., and Conover, W. J., "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code," *Technometrics*, Vol. 21, No. 2, 1979, p. 239. https://doi.org/10.2307/1268522.

[19] LIANG, Y., LEE, H., LIM, S., LIN, W., LEE, K., and WU, C., "PROPER ORTHOGONAL DECOMPOSITION AND ITS APPLICATIONS—PART I: THEORY," *Journal of Sound and Vibration*, Vol. 252, No. 3, 2002, p. 527–544. https://doi.org/10.1006/jsvi.2001.4041.

[20] D., P. M. J., *The theory of radial basis function approximation in 1990*, University of Cambridge. Department of Applied Mathematics and Theoretical Physics, 1990.

[21] Khattak, A. J., Tirmizi, S., and ul Islam, S., "Application of meshfree collocation method to a class of nonlinear partial differential equations," *Engineering Analysis with Boundary Elements*, Vol. 33, No. 5, 2009, p. 661–667. https://doi.org/10.1016/j.enganabound.2008.10.001.

[22] Dehghan, M., and Shokri, A., "A meshless method for numerical solution of the one-dimensional wave equation with an integral condition using radial basis functions," *Numerical Algorithms*, Vol. 52, No. 3, 2009, p. 461–477. https://doi.org/10.1007/s11075-009-9293-0.

[23] Demo, N., Tezzele, M., and Rozza, G., "EZyRB: Easy Reduced Basis method," *The Journal of Open Source Software*, Vol. 3, No. 24, 2018, p. 661. https://doi.org/https://doi.org/10.21105/joss.00661.

[24] "Chord (aeronautics)," , Jan 2021. URL https://en.wikipedia.org/wiki/Chord_(aeronautics).