Spatially Clustered Local Bases for Reduced Order Models

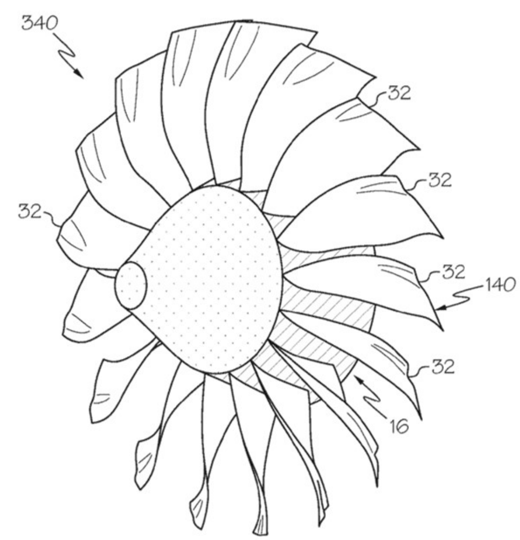
STANFORD UNIVERSITY – FARHAT RESEARCH GROUP – 7/19/2017

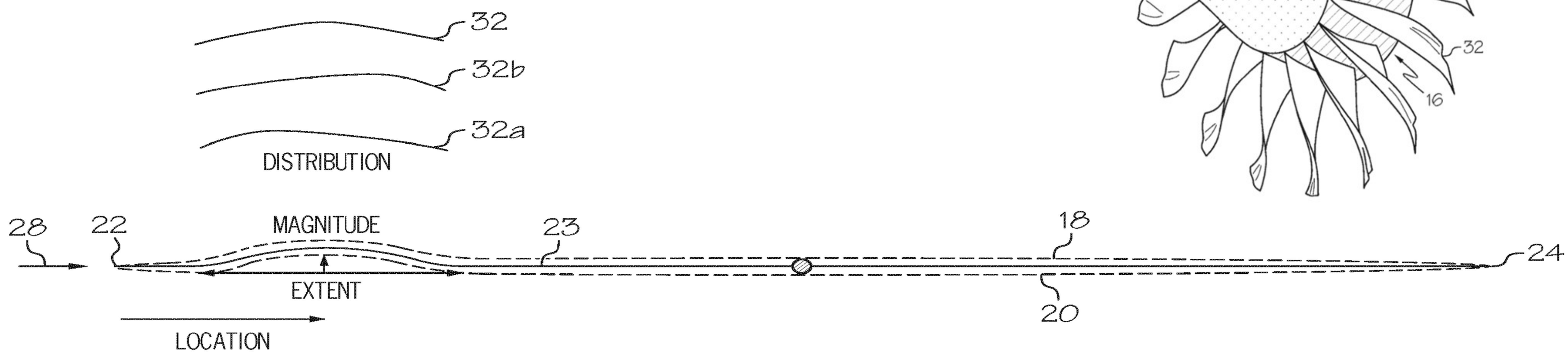
TINA WHITE

Motivation

Numerical experiments can tell you how a parameter will influence your design's performance. For example, the local changes to fan blade curvature shown may have a large influence on parameters of interest.

But full order simulations can be very slow!

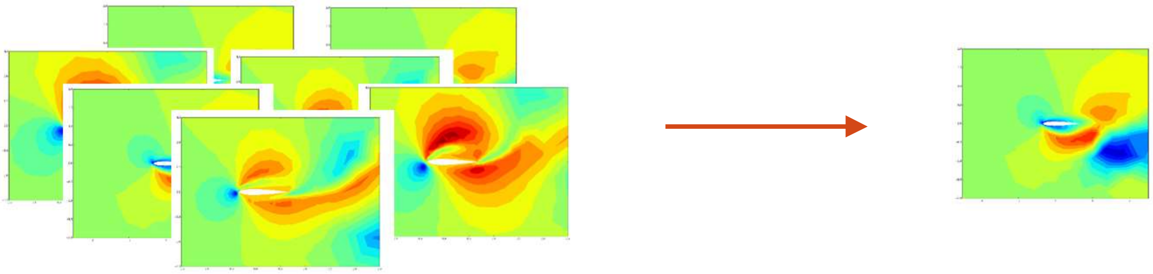




What’s the broader problem?

How can you use available solutions to predict future results?

How do you combine modeling and experience to make good predictions?



ROMs solve this problem

ROMs solve the original governing equations after projecting them into a smaller space, called a basis, which is formed using knowledge gained from previous simulations. The smaller basis limits the degrees of freedom in order to run faster simulations.

Start with a PDE $$\dot{\mathrm{w}} + F(\mathrm{w}) = 0$$

Discretize $$\frac{\mathrm{w}^{n+1} - \mathrm{w}^{n}}{\Delta t} + F(\mathrm{w}^{n}) = 0$$

$$w^{n+1} = w^{n} + \Delta t\, F(w^{n})$$

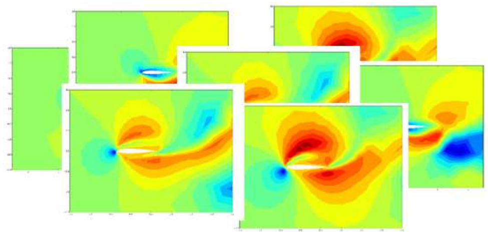
ROMs need a basis V

Reduce dimensionality $$w \sim \boldsymbol{V} y$$

$w \in \mathbb{R}^{M},\ V \in \mathbb{R}^{M\times n},\ y \in \mathbb{R}^{n}$ $$y^{n+1} = y^{n} + \Delta t\, \boldsymbol{V}^{T} F(\boldsymbol{V} y^{n})$$

How do you choose a basis? What is V?

Given a snapshot matrix A of with full order simulations along the columns, a basis V can be constructed.

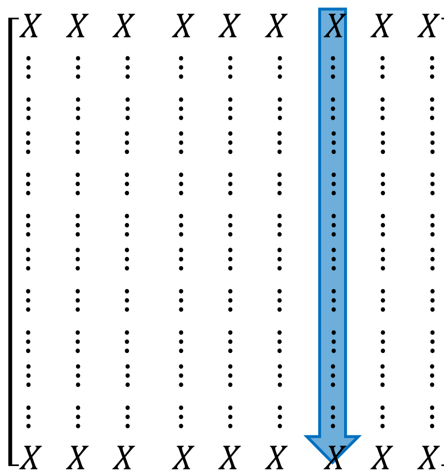


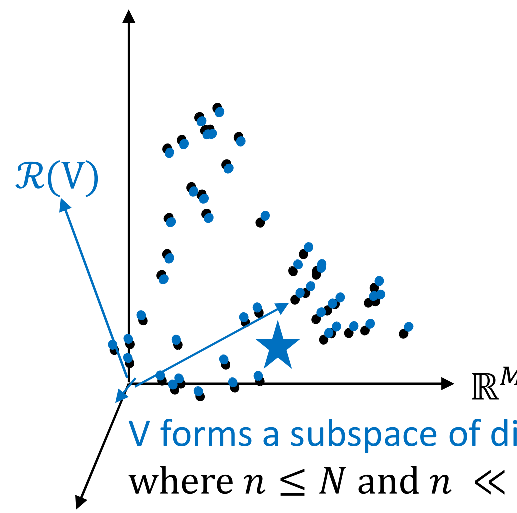
Snapshot matrix $A \in \mathbb{R}^{M \times N}$ A

Perform SVD $A = U\Sigma V^T$

Truncate and save first *n* left singular vectors $V \in \mathbb{R}^{M \times n}$ $\boldsymbol{V} = U_{trunc}$

$w \sim \boldsymbol{V} y$

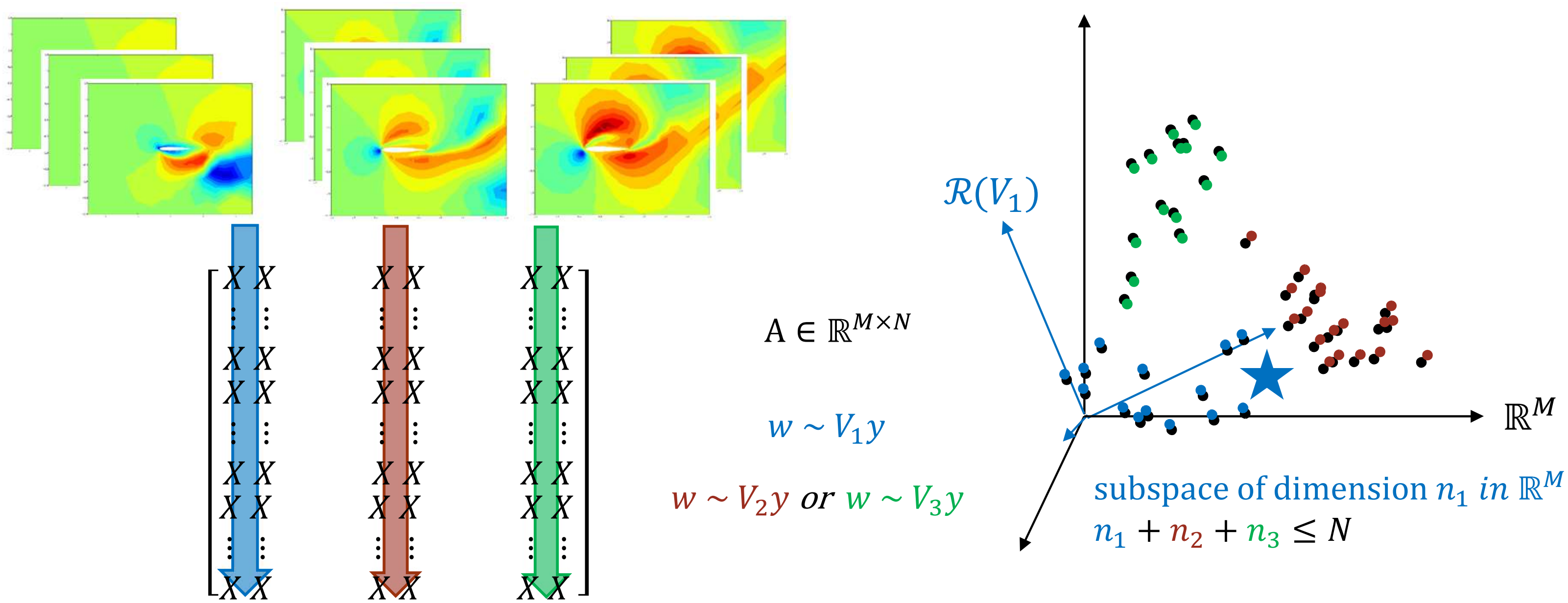




V forms a subspace of dim n
where $n \leq N$ and $n \ll M$

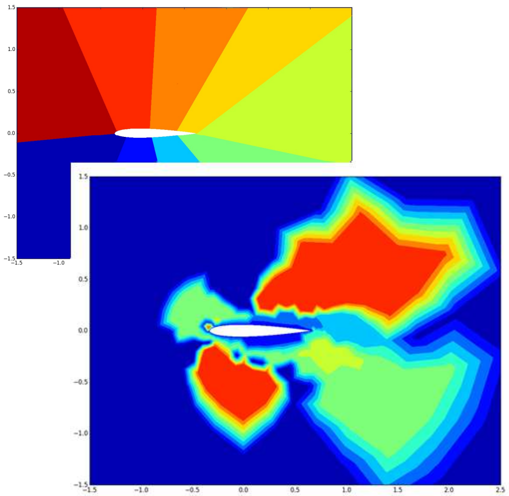
What is a local basis?

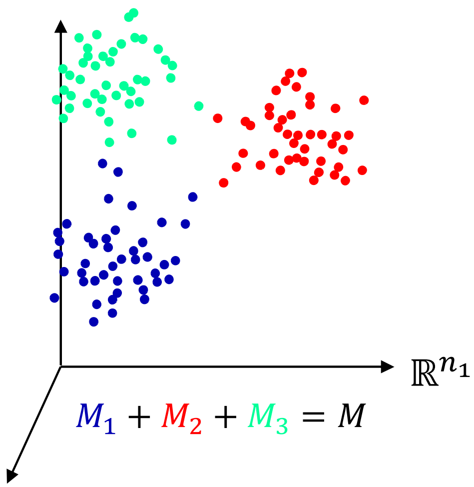
Similar snapshots are grouped together using the k-means algorithm. Within each column cluster, the ROM ignores other clusters of simulations that are not relevant, and the ROM runs faster. Local ROMs are solved in the basis corresponding to the closest cluster center.

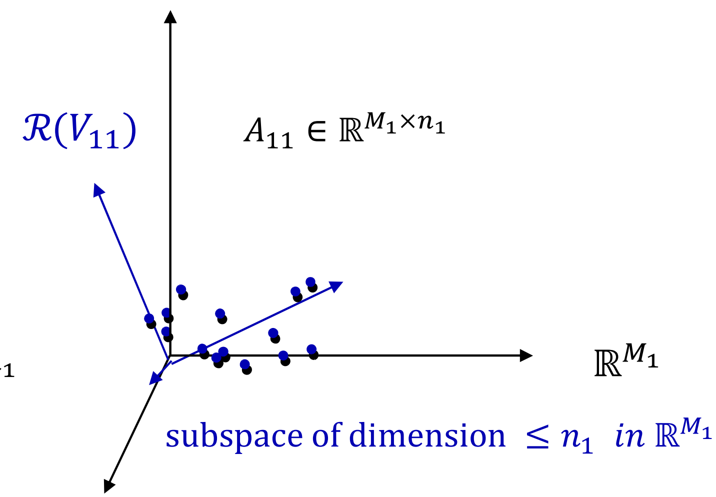


The Idea – Spatially Clustered Local Bases

Within a local column cluster, points in space, corresponding to a rows of the matrix, can be clustered, giving regions that show similar behavior over time – similar to flow features.



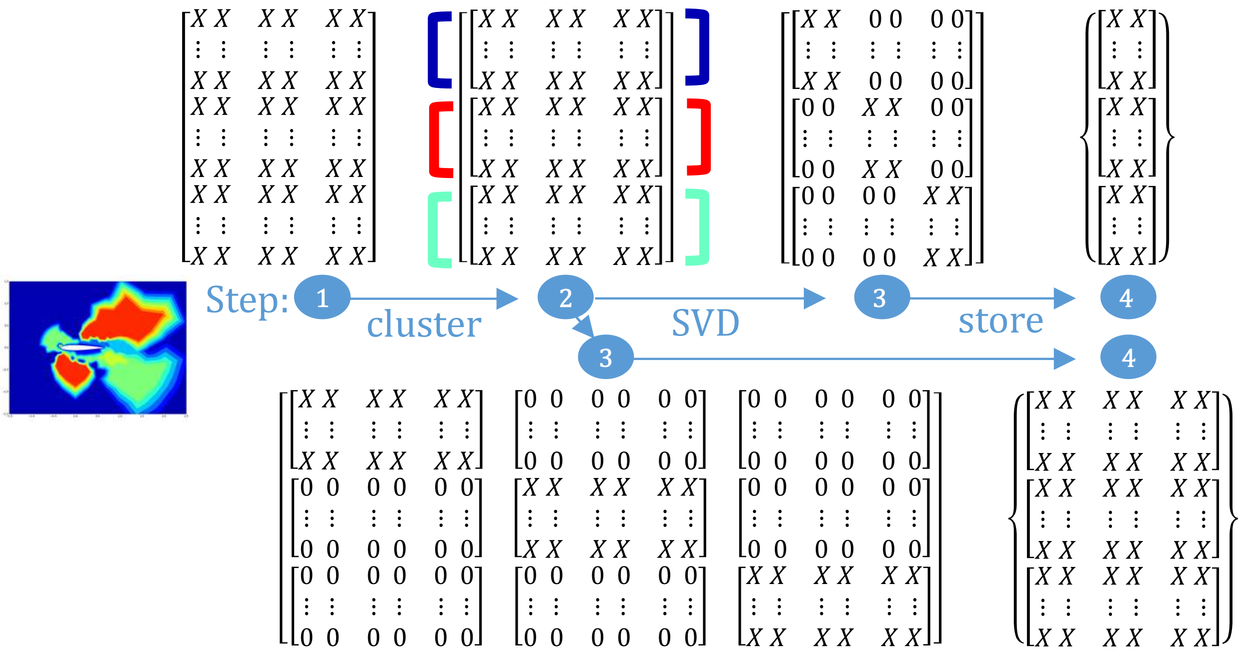




Thus, different models are run for different spatial flow features. The feature-based subspaces are not chosen by hand, but by k-means clustering along the rows of the snapshot matrix.

The Idea – Spatially Clustered Local Bases

More basis vectors can be extracted from the matrix by increasing the number of row clusters, giving more accurate results even with very few full order solutions available.



Option 1

Capture the same amount of energy as a single SVD, with a same-sized sparser matrix, **same** accuracy with **faster** computation time.

Option 2

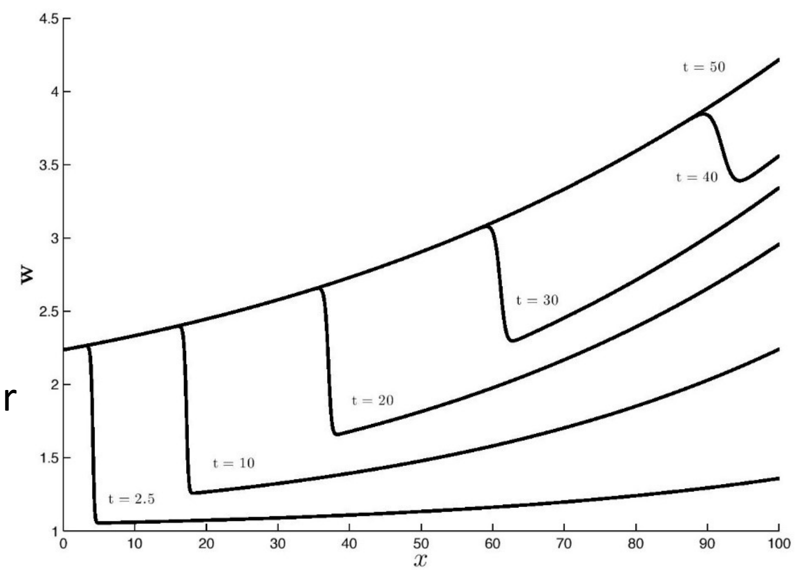
Capture more energy than a single SVD with a larger sparser matrix holding same number of non-zero values, **greater** accuracy with **same** computation time.

Test Case – Burgers' Equation

The Burgers' equation is a good test case for ROMs. Results tend to translate to harder problems. It is a one-dimensional application of an initial-boundary-value problem that models the movement of a shockwave in a fluid.

$$\underset{\text{UNSTEADY TERM}}{\frac{\partial u}{\partial t}} + \underset{\text{CONVECTIVE TERM}}{u\frac{\partial u}{\partial x}} = \underset{\text{VISCOUS TERM}}{\nu\frac{\partial^2 u}{\partial x^2}}$$

Datasets and .py MOR testbed code from Matt Zahr

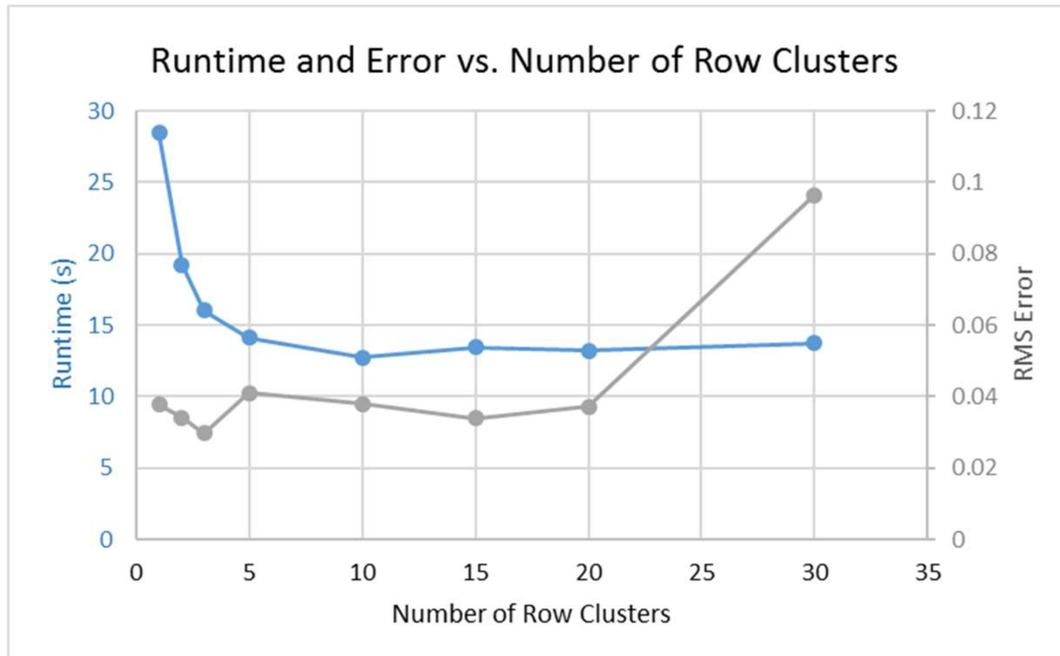


Test Case Results – ROM Solutions

Option 1

Capture the same amount of energy as a single SVD, with a same-sized sparser matrix, **same** accuracy with **faster** computation time.

$$\begin{bmatrix} X\,X & X\,X & X\,X \\ \vdots\ \vdots & \vdots\ \vdots & \vdots\ \vdots \\ X\,X & X\,X & X\,X \\ X\,X & X\,X & X\,X \\ \vdots\ \vdots & \vdots\ \vdots & \vdots\ \vdots \\ X\,X & X\,X & X\,X \\ X\,X & X\,X & X\,X \\ \vdots\ \vdots & \vdots\ \vdots & \vdots\ \vdots \\ X\,X & X\,X & X\,X \end{bmatrix} \rightarrow \begin{bmatrix} \begin{bmatrix} X\,X & 0\,0 & 0\,0 \\ \vdots\ \vdots & \vdots\ \vdots & \vdots\ \vdots \\ X\,X & 0\,0 & 0\,0 \end{bmatrix} \\ \begin{bmatrix} 0\,0 & X\,X & 0\,0 \\ \vdots\ \vdots & \vdots\ \vdots & \vdots\ \vdots \\ 0\,0 & X\,X & 0\,0 \end{bmatrix} \\ \begin{bmatrix} 0\,0 & 0\,0 & X\,X \\ \vdots\ \vdots & \vdots\ \vdots & \vdots\ \vdots \\ 0\,0 & 0\,0 & X\,X \end{bmatrix} \end{bmatrix}$$

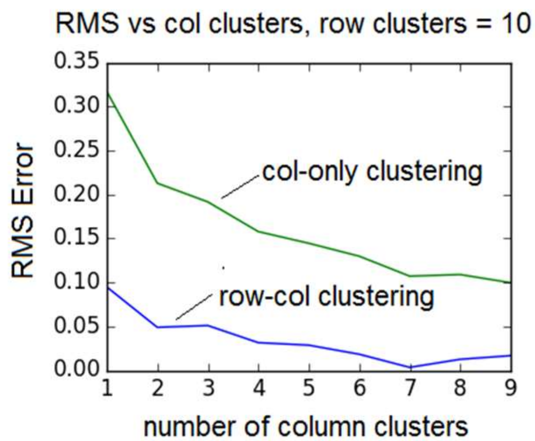


Test Case Results – Projections

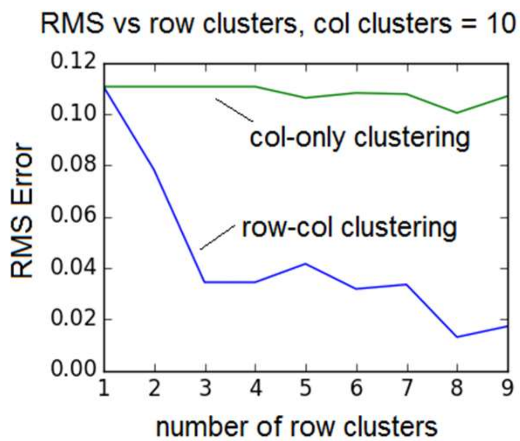
Option 2

Capture more energy than a single SVD with a larger sparser matrix holding same number of non-zero values,
greater accuracy with
same computation time.

$$
\begin{bmatrix}
\begin{bmatrix} X & X & X & X & X & X \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X & X & X & X & X & X \end{bmatrix} &
\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} &
\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} &
\begin{bmatrix} X & X & X & X & X & X \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X & X & X & X & X & X \end{bmatrix} &
\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} &
\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} &
\begin{bmatrix} X & X & X & X & X & X \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X & X & X & X & X & X \end{bmatrix}
\end{bmatrix}
$$



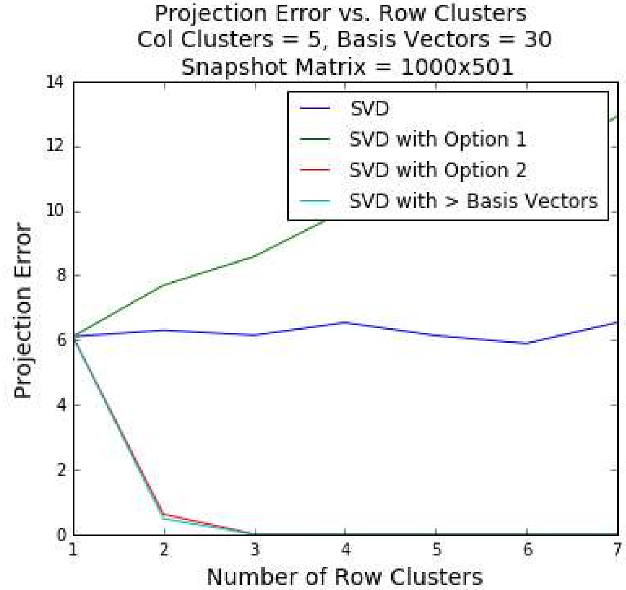
RMS error reduction with increasing column clusters given fixed number of row clusters



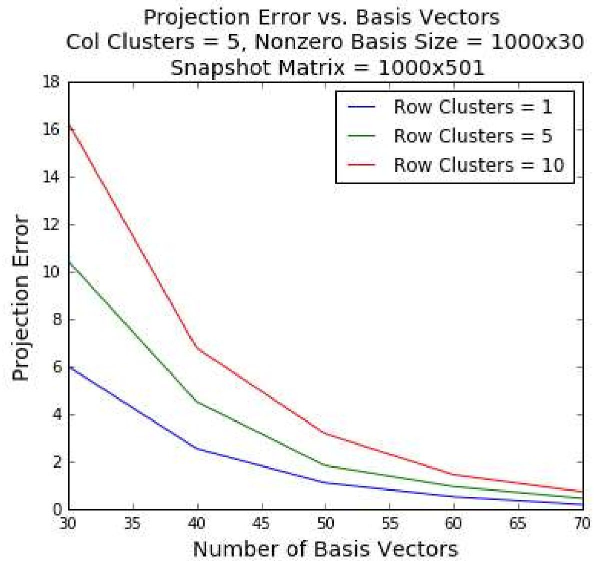
RMS error reduction with increasing row clusters given fixed number of column clusters

Investigation into Projection Error

<u>Option 1 and 2 Projection Error Comparisons</u>
What happens if you project the original snapshot matrix into each type of basis?



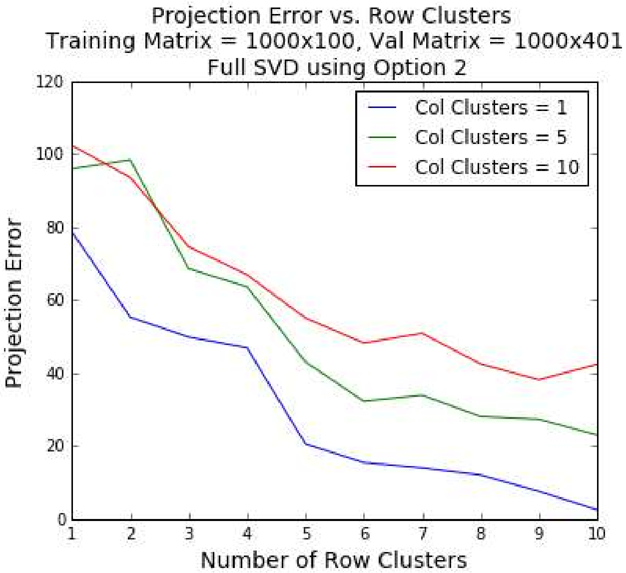
<u>Option 1 Projection Error Investigation</u>
What happens if you project the original snapshot matrix into Option 1 matrices (nonzero basis size held constant) with different numbers of row clusters and basis vectors?



Investigation into Projection Error

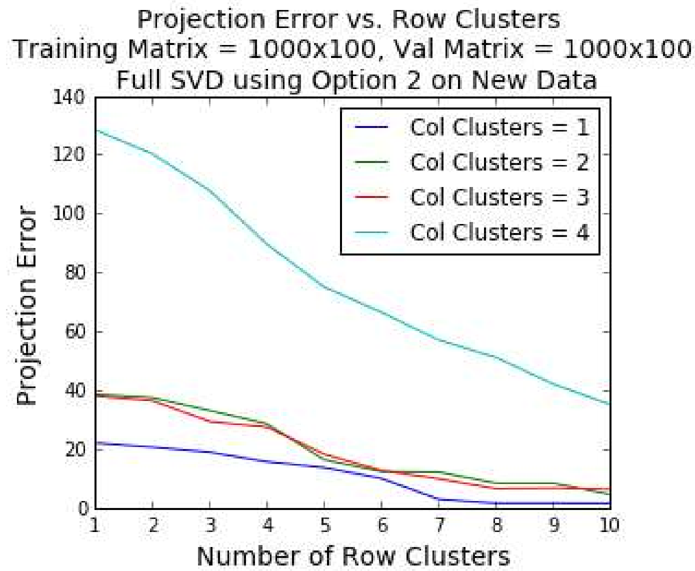
<u>Option 2 Projection Error Investigation</u>

What happens if you separate the snapshot matrix into a randomly chosen small training set and a validation set, and project the validation set on the basis learned from training?



<u>Option 2 Projection Error Investigation</u>

What happens if your training set only includes two hyperparameters, and your validation set only includes a third hyperparameter that must now be interpolated (new data).



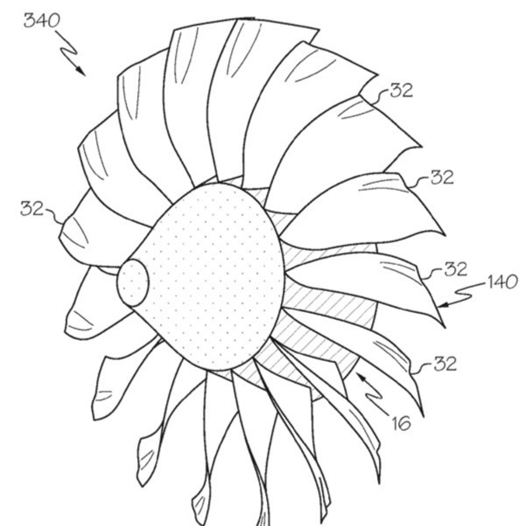
Final Results

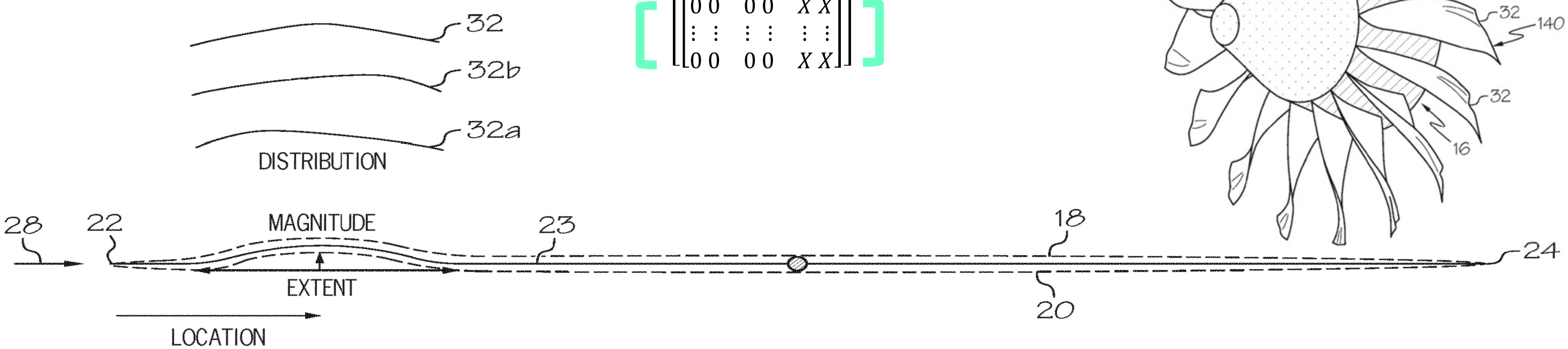
Conclusion

A new method for solving these problems faster and more accurately using ROMs:

Spatially Clustered Local Bases

$$\begin{bmatrix} \begin{bmatrix} X\,X & 0\,0 & 0\,0 \\ \vdots\ \vdots & \vdots\ \vdots & \vdots\ \vdots \\ X\,X & 0\,0 & 0\,0 \end{bmatrix} \\ \begin{bmatrix} 0\,0 & X\,X & 0\,0 \\ \vdots\ \vdots & \vdots\ \vdots & \vdots\ \vdots \\ 0\,0 & X\,X & 0\,0 \end{bmatrix} \\ \begin{bmatrix} 0\,0 & 0\,0 & X\,X \\ \vdots\ \vdots & \vdots\ \vdots & \vdots\ \vdots \\ 0\,0 & 0\,0 & X\,X \end{bmatrix} \end{bmatrix}$$





References

[1] D. Amsallem, M. Zahr and C. Farhat, Nonlinear Model Order Reduction Based on Local Reduced-Order Bases, International Journal for Numerical Methods in Engineering, Vol. 92, pp. 891-916 (2012)

[2] K. Carlberg, C. Bou-Mosleh and C. Farhat, Efficient Nonlinear Model Reduction via a Least-Squares Petrov-Galerkin Projection and Compressive Tensor Approximations, International Journal for Numerical Methods in Engineering, Vol. 86, pp. 155-181 (2011)

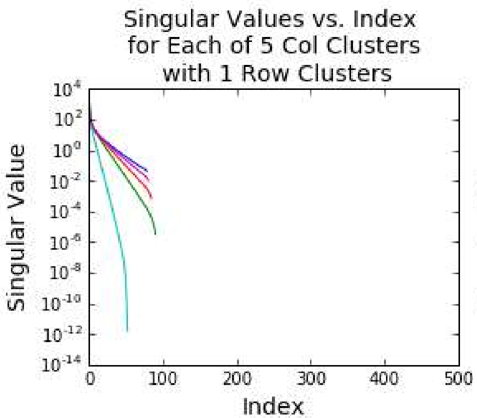
[3] C. Farhat, T. Chapman and P. Avery, Structure-Preserving, Stability, and Accuracy Properties of the Energy-Conserving Sampling and Weighting (ECSW) Method for the Hyper Reduction of Nonlinear Finite Element Dynamic Models, International Journal for Numerical Methods in Engineering, Vol. 102, pp. 1077-1110 (2015)

[4] Farhat, Charbel, Michael Lesoinne, and P. Le Tallec. "Load and motion transfer algorithms for fluid/structure interaction problems with non-matching discrete interfaces: Momentum and energy conservation, optimal discretization and application to aeroelasticity." Computer methods in applied mechanics and engineering 157.1 (1998): 95-114.

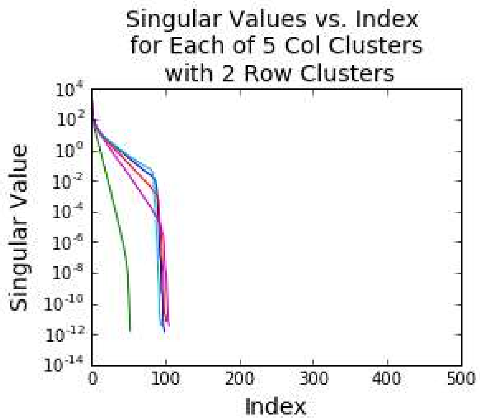
[5] Geuzaine, Philippe, et al. "Aeroelastic dynamic analysis of a full F-16 configuration for various flight conditions." AIAA journal 41.3 (2003): 363-371.

[6] Gottlieb, David, and Chi-Wang Shu. "On the Gibbs phenomenon and its resolution." SIAM review 39.4 (1997): 644-668.

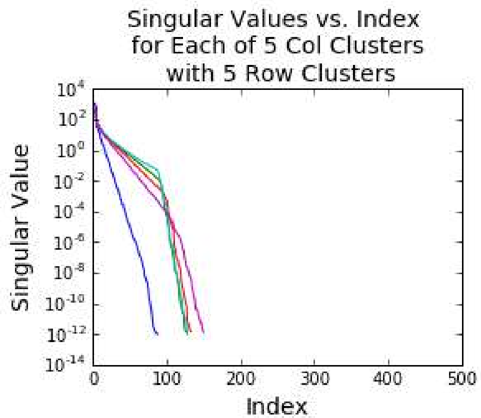
Option 1 Results for Row-Col Clustering



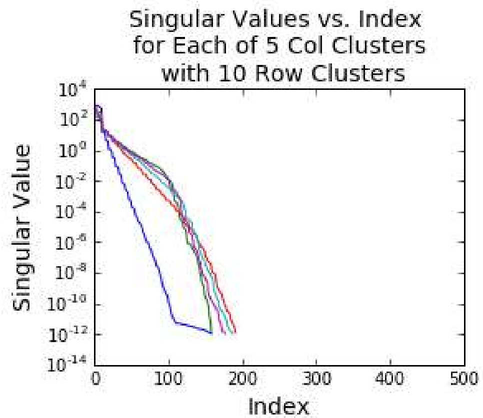
5 Col Clusters with 1 Row Clusters
--- Avg 12.1609 seconds ---
--- Avg Mean Abs Error is 0.02029---
--- Avg Mean Sqrt Error is 0.12083---



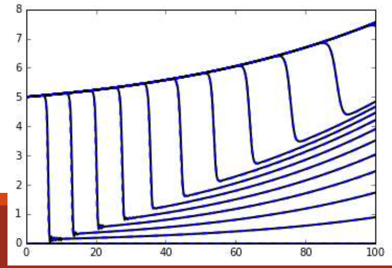
5 Col Clusters with 2 Row Clusters
--- Avg 9.2743 seconds ---
--- Avg Mean Abs Error is 0.00729---
--- Avg Mean Sqrt Error is 0.05053---



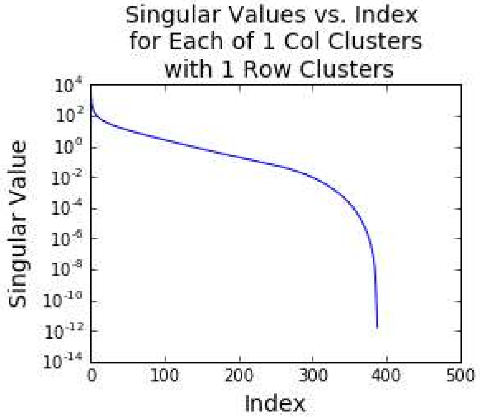
5 Col Clusters with 5 Row Clusters
--- Avg 8.1426 seconds ---
--- Avg Mean Abs Error is 0.01387---
--- Avg Mean Sqrt Error is 0.08759---



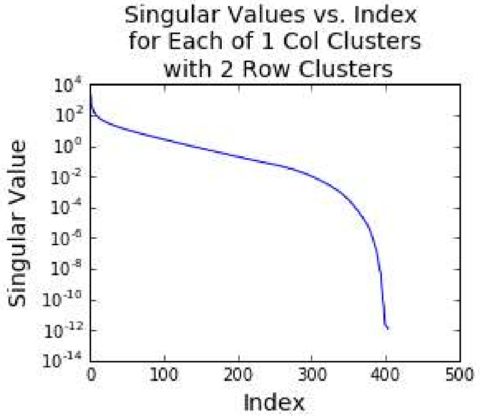
5 Col Clusters with 10 Row Clusters
--- Avg 7.9336 seconds ---
--- Avg Mean Abs Error is 0.01891---
--- Avg Mean Sqrt Error is 0.12421---



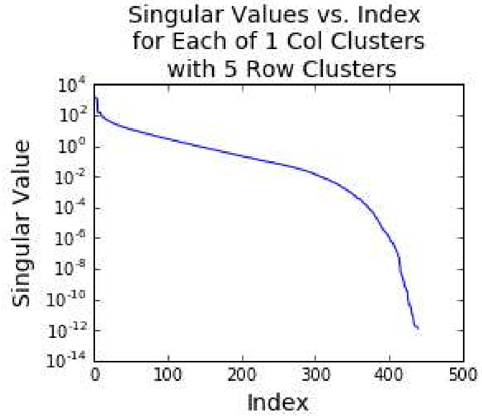
Option 1 Results for Row-Only Clustering



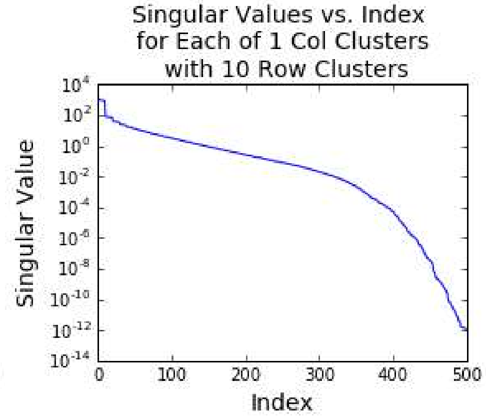
1 Col Clusters with 1 Row Clusters
--- Avg 7.6635 seconds ---
--- Avg Mean Abs Error is 0.05164---
--- Avg Mean Sqrt Error is 0.09729---



1 Col Clusters with 2 Row Clusters
--- Avg 6.3926 seconds ---
--- Avg Mean Abs Error is 0.04983---
--- Avg Mean Sqrt Error is 0.1171---



1 Col Clusters with 5 Row Clusters
--- Avg 5.4625 seconds ---
--- Avg Mean Abs Error is 0.06176---
--- Avg Mean Sqrt Error is 0.19966---



1 Col Clusters with 10 Row Clusters
--- Avg 5.5769 seconds ---
--- Avg Mean Abs Error is 0.09045---
--- Avg Mean Sqrt Error is 0.33677---