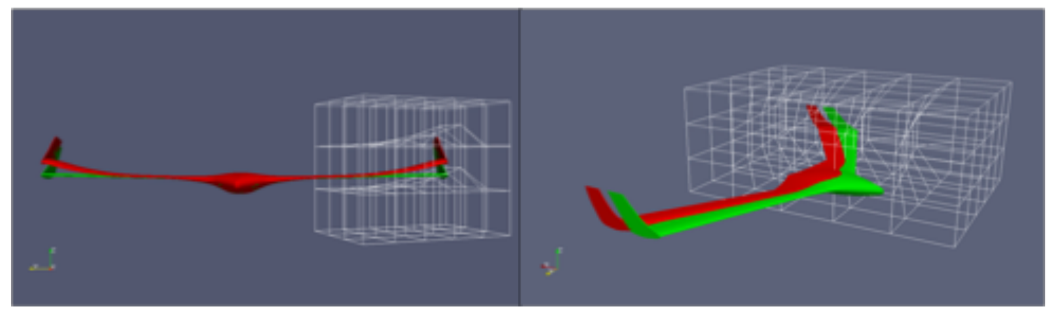
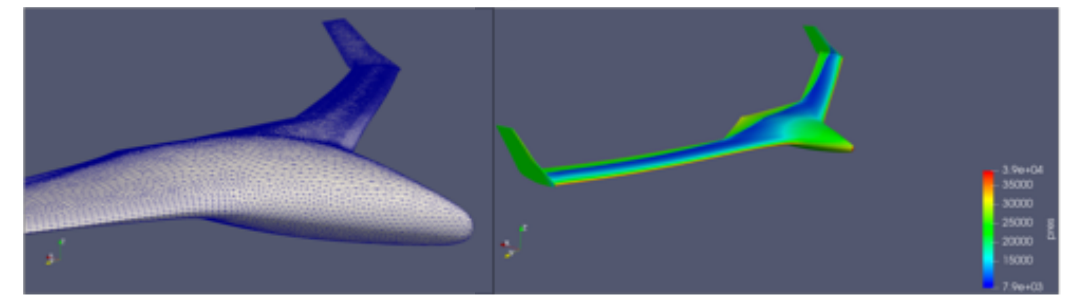
Clustering Reduced Order Models for Computational Fluid Dynamics



Gabriele Boncoraglio and Forest James Charles Fraser

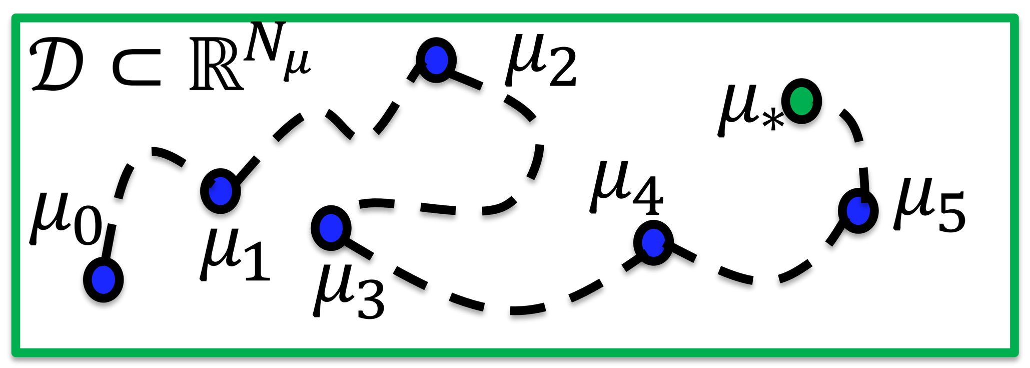
{gbonco, forestf}@stanford.edu



Overview

Improving the design of aircrafts requires solving **PDE-constrained optimization problems** such as maximizing the lift/drag with respect to some parameters, $\boldsymbol{\mu}$. To find the optimal $\boldsymbol{\mu}$, we must update it iteratively, running an expensive computational fluid dynamics (CFD) simulation at each optimization step.

$$\max_{\boldsymbol{\mu}\in\mathcal{D}} \quad \frac{\mathrm{Lift}(\boldsymbol{\mu})}{\mathrm{Drag}(\boldsymbol{\mu})}$$
$$\text{s.t.} \quad \mathrm{Lift}(\boldsymbol{\mu}) \geqslant \mathrm{Lift}_0$$
$$\boldsymbol{\mu}_{\mathrm{lb}} \leqslant \boldsymbol{\mu} \leqslant \boldsymbol{\mu}_{\mathrm{ub}}$$



Objective: Accelerate the optimization process by using clustering/classification techniques to generate and use multiple ***reduced order models (ROMs)*** for less expensive, yet still accurate simulations.

Background

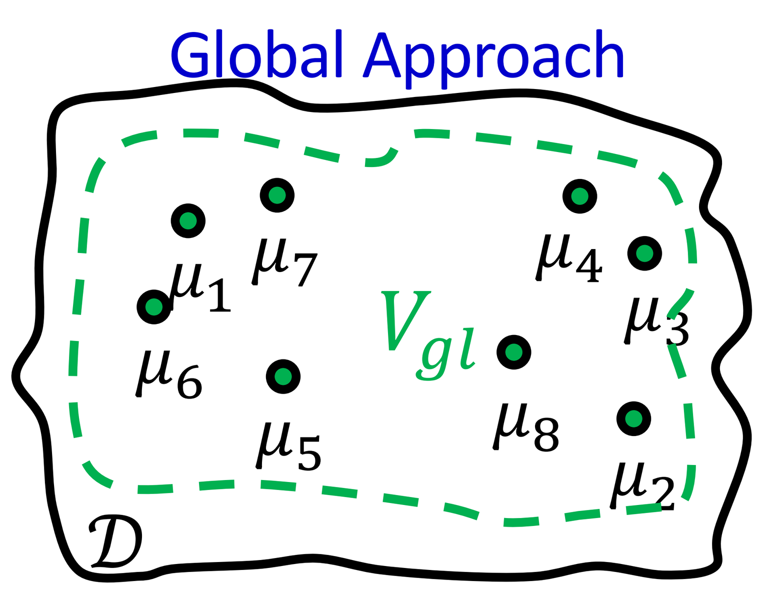
Running a CFD simulation involves solving for the state of the fluid, **w**. To speed up CFD simulations ROMs are used to approximate the results of a full simulation.

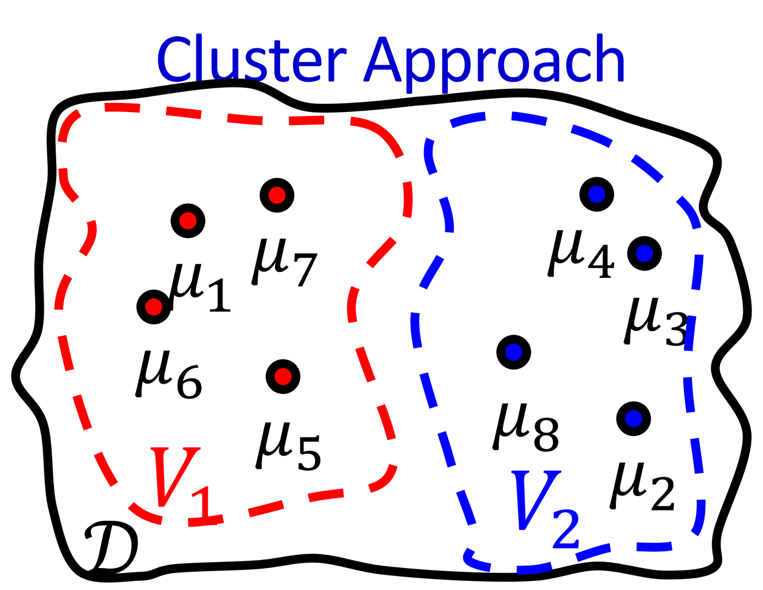
The ROM approximates the fluid state as: $\mathbf{w} \approx \mathbf{V}_{gl}(\boldsymbol{\mu})\mathbf{w}_r$

where the reduced order basis (ROB), $\mathbf{V}_{gl}$, is built using precomputed solution $\{\mathbf{w}_1, \mathbf{w}_2 \ldots, \mathbf{w}_k\}$ computed at $\{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2 \ldots, \boldsymbol{\mu}_k\}$.

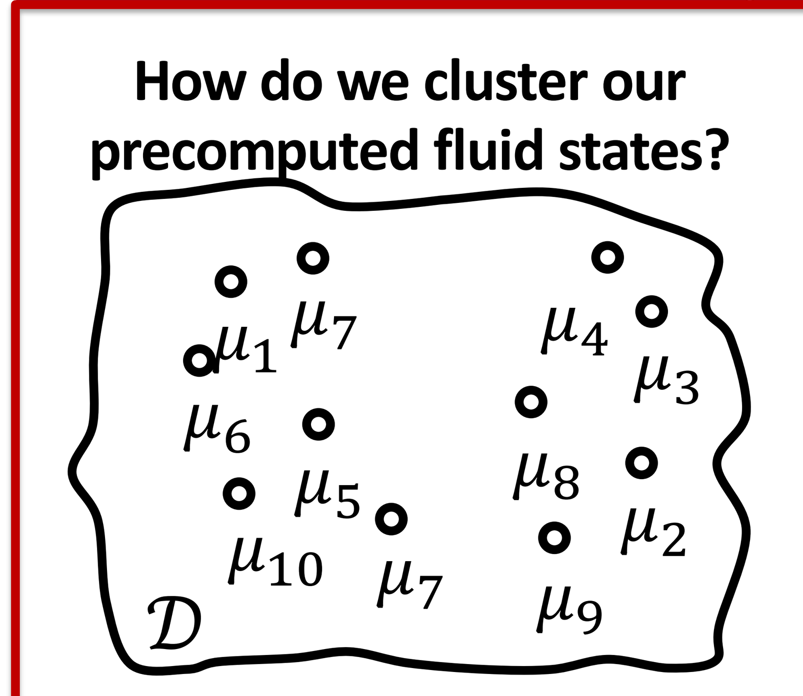
Usually one global ROM is constructed, however we propose building multiple, smaller, more localized ROMs, since they

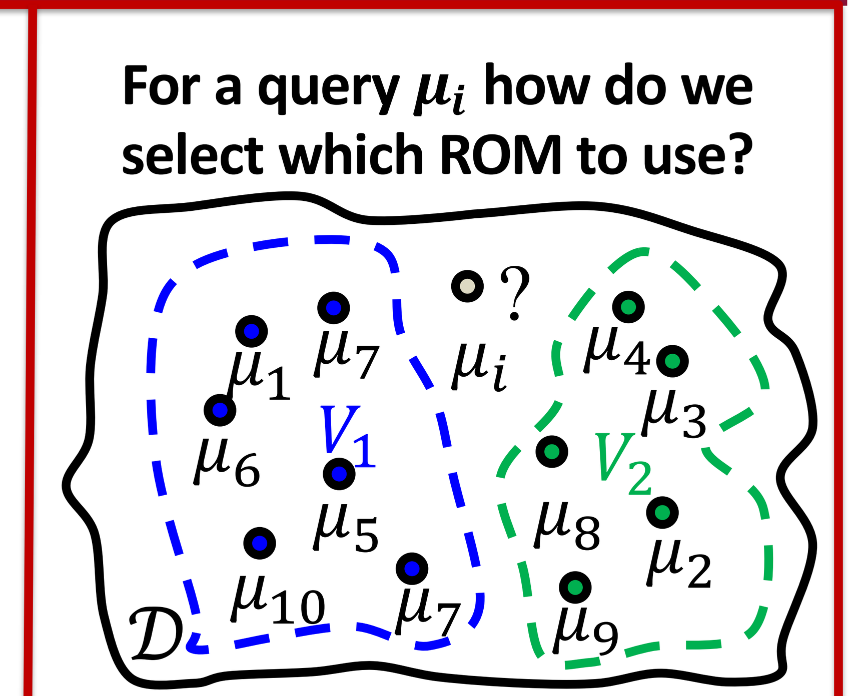
- have fewer unknowns, hence simulations are faster
- may more accurately approximate the full simulation within a sub-region of the design space, $\mathcal{D}$





Key Issues

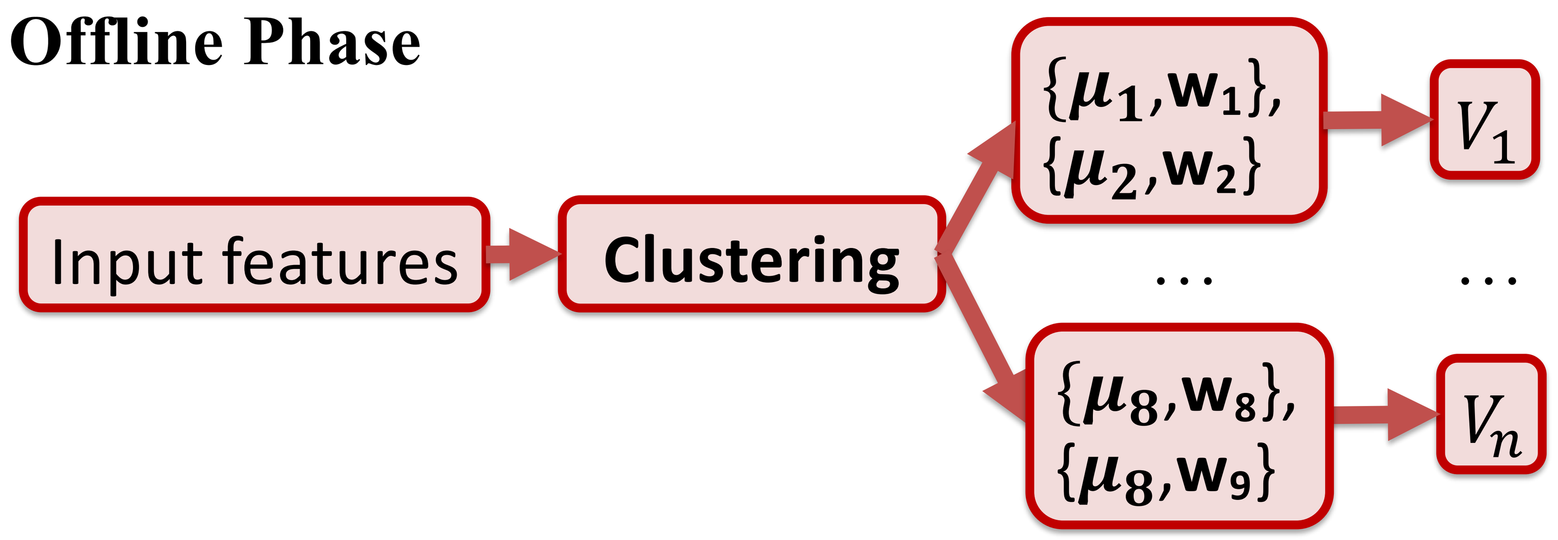




Methodology

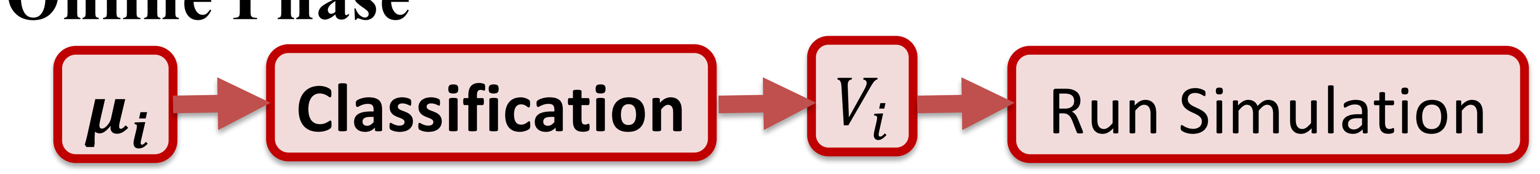
Our proposed methodology solves a PDE-constrained optimization problem in two phases, an offline phase and an online phase.

Offline Phase



In the offline phase, we cluster precomputed training solutions, from which we build our ROMs that are used in the online phase.

Online Phase



In the online phase, we query multiple $\boldsymbol{\mu}$, during the optimization process. For each queried $\boldsymbol{\mu}_i$, we need select which ROM (V_i) to use. Then we run the simulation to compute $\boldsymbol{w}_i(\boldsymbol{\mu}_i)$ and $\frac{lift}{drag}(\boldsymbol{\mu}_i)$.

Sampling

- We sample a set of $\{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots \boldsymbol{\mu}_{90}\}$ for which we calculate the states $\{\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots \boldsymbol{w}_{90}\}$ using full simulations.
- From our set of $\boldsymbol{\mu}$ and $\boldsymbol{w}$ we randomly split our data into training/validation/test sets of size 50, 30 and 10 respectively.

Experiments & Results

Using our validation set we tested which clustering and classification algorithms performed best using 4 clusters, $\{\boldsymbol{\mu}, \frac{lift}{drag}\}$ as clustering features and $\boldsymbol{\mu}$ as the classification feature.

Clustering algorithm	MSE	Max Error %
K-Means	0.327	23.117
Gaussian mixtures	0.421	28.898
Agglomerative clus.	0.340	23.917

Classification algorithm	MSE	Max Error %
Logistic regression	0.318	29.490
Nearest centroid	0.341	29.490

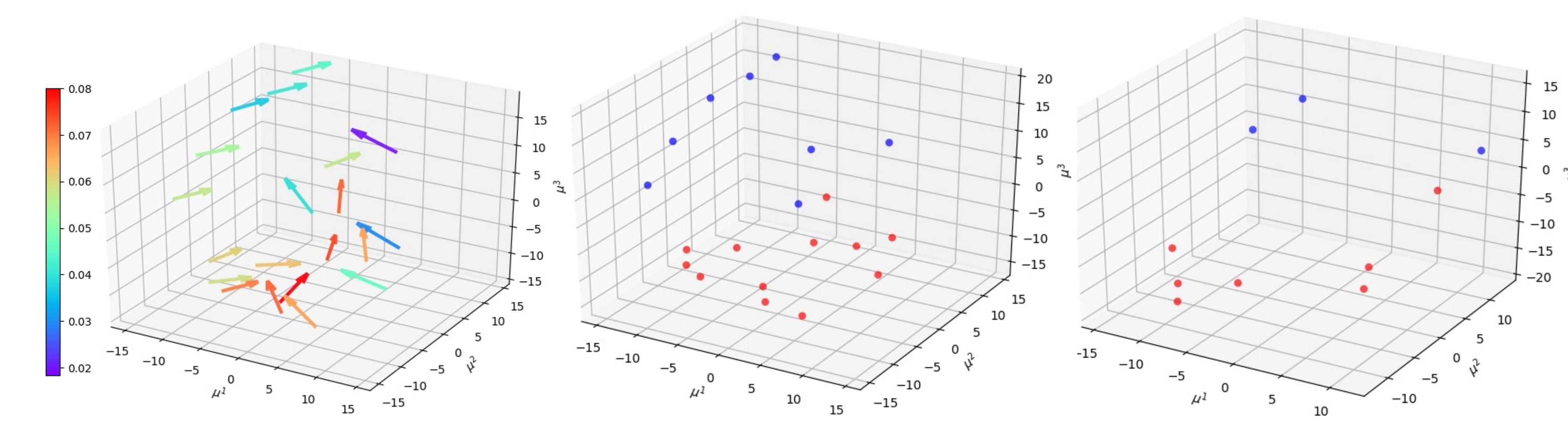
Then using K-Means and logistic regression we tested the performance of different clustering features and cluster sizes.

Number of clusters	MSE	Max Error %
2 clusters	0.255	23.072
3 clusters	0.288	23.520
4 clusters	0.354	26.133
5 clusters	0.417	29.826

Cluster features	MSE	Max Error %
$\left(\boldsymbol{\mu}, \frac{\text{lift}}{\text{drag}}\right)$	0.3543	30.955
$\left(\boldsymbol{\mu}, \frac{\partial \frac{\text{lift}}{\text{drag}}}{\partial \boldsymbol{\mu}}\right)$	0.3145	30.955
$\left(\boldsymbol{\mu}, \frac{\text{lift}}{\text{drag}}, \frac{\partial \frac{\text{lift}}{\text{drag}}}{\partial \boldsymbol{\mu}}\right)$	0.3409	30.955

Finally we tested the performance of the clustered ROMs on the test set using the optimal parameters found from validation.

Training Gradients **Training Clusters** **Testing Classification**



Cluster vs Global ROM Comparison

Method	Clusters	ROM Sizes	MSE	Max Error %
Cluster ROM	2	(12, 8)	0.051	7.726
Cluster ROM	4	(5, 7, 5, 3)	0.200	17.439
Global ROM	-	14	0.194	21.156

Conclusions and Future Work

- Using our methodology allows us to either accelerate the optimization process or achieve a higher simulation accuracy when compared with a global ROM.
- In the future, we would like develop an accurate predictor for determining the optimal parameters for clustering/classification.