Clustered Active Subspaces Applied to Aerodynamic Shape Optimisation

Maxime Chapron, Christophe Blondeau, Itham Salah El Din, Denis Sipp, Michel Bergmann

The 9th European Congress on Computational Methods in Applied Sciences and Engineering
ECCOMAS Congress 2024
3–7 June 2024, Lisboa, Portugal

CLUSTERED ACTIVE SUBSPACES APPLIED TO AERODYNAMIC SHAPE OPTIMISATION

Maxime Chapron[1,3], Christophe Blondeau[1], Itham Salah El Din[2], Denis Sipp[2], and Michel Bergmann[3]

[1] ONERA - Department of Aerodynamics, Aeroelasticity, Acoustics (DAAA)
Institut Polytechnique de Paris, 92320, Châtillon, France
e-mail: maxime.chapron@onera.fr

[2] ONERA - Department of Aerodynamics, Aeroelasticity, Acoustics (DAAA)
Institut Polytechnique de Paris, 92190, Meudon, France

[3] INRIA Bordeaux - Memphis Team
200 Avenue de la Vieille Tour, 33405 Talence, France

Key words: Dimension Reduction, Surrogate Modelling, Optimisation, Clustering

Summary. Surrogate-based optimisation in design spaces exceeding several tens of parameters is numerically intractable yet more and more frequent. Dimension reduction is a potent answer to this problem. However, typical dimension reduction methods seek to identify global trends over the entire design space. We propose to use a combination of Active Subspaces in subregions of the design space, where their use of the objective function's gradients will exploit local information to discover specific trends in the objective function. Local surrogate models are then constructed in the subspaces, and recombined to form a global surrogate model, which we can then exploit for optimisation.

The partitioning of the input space is done through a model based Gaussian Mixture Model clustering, where the authors assume the distribution of the joint inputs/outputs to be a mixture of Gaussians. After labelling the points with GMM, we propose to train a supervised Support Vector Classifier. This strategy scales better with high-dimensional parameter spaces and yields an explicit analytical expression of the clusters' boundaries, from which we can formulate an improved recombination strategy with overlapping between the surrogate models. The use of overlapping increases prediction accuracy at the boundaries between clusters, an improvement over recombination methods typically found in the literature.

1 Introduction

Aerodynamic shape optimisation is now a routine part of modern aircraft design. One seeks to minimize a scalar quantity of interest such as drag, under lift and/or moment constraints, with respect to design variables which control the object's shape. This kind of problem is usually solved one of three ways: gradient-based methods [1, 2], stochastic gradient-free methods [3, 4, 5], and surrogate-based methods [6, 7]. These methods have advantages and drawbacks, regarding the number of function evaluations they require, their scalability with the number of design parameters or their robustness to multimodal functions. Our goal is to bridge the gap between gradient-based methods, which are robust to dimension but vulnerable to multimodality, and

surrogate-based methods, which are global methods but subject to the curse of dimensionality. Typically, Surrogate-Based Optimisation (SBO) is assumed to be intractable for design spaces of more than a couple dozen dimensions [8], due to high training cost and low accuracy. Toal et al. [9] have shown that in such cases, it is better to precisely determine a few hyperparameters rather than try to estimate a lot of them. This is the motivation for dimension reduction [10].

Dimension reduction is an active field of research, with many available strategies. In our case, we wish to map an integrated quantity such as drag as a function of an object's shape, with the belief that a lot of engineering quantities of interest exhibit lower dimensionality than the problems they are parameterised as. Therefore, linear embeddings are of particular interest to us [11, 12, 13, 14]. Active Subspaces [15] are especially appropriate since they leverage gradient information to reveal a low dimensional subspace which captures the trends in the quantities of interest. One can then build a regression surface in the reduced space, where the number of hyperparameters is reduced, thus their estimation simplified.

In cases where the underlying substructure of data is nonlinear, global linear embeddings may not be all that accurate. To that end, we build upon the work of Constantine [15], Palacios et al [16] and Bartoli et al. [17] and propose a way to construct a surrogate model for a high dimensional design space, with the goal of later using that surrogate model to drive a global surrogate-based optimisation. Instead of discovering a single, global reduced space, we use a set of local active subspaces to discover local trends in the quantity of interest. Essentially, we are splitting up the problem into several smaller, simpler problems, a common practice in the case of tortuous, possibly multimodal functions [17, 18]. This piecewise linear approach bridges the gap with nonlinear dimension reduction methods [19, 20].

After explaining the reasoning behind the proposed Clustered Active Subspaces method (section 2), from the clustering of the design space, to dimension reduction and the construction of the surrogate model in the reduced spaces, we introduce our novel overlapping strategy (section 3). We will close with a presentation of surrogate modelling results on benchmark CFD test cases.

2 Clustered Active Subspaces

We start with a reminder of our Clustered Active Subspaces method [21], explaining its organisation and the reasoning behind our choices.

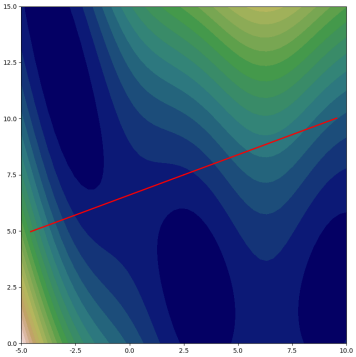
Our starting point was the work of Palacios et al. [22] which presented Active Subspaces for surrogate-based aerodynamic shape optimisation. The Active Subspace (AS) method proposed by Constantine [15] is used to identify a linear combination of input parameters which are most strongly correlated to changes in the objective function, by solving the eigendecomposition of the gradient correlation matrix, and keeping only the eigenvectors associated with the largest eigenvalues, which form a reduced-order basis $\mathbf{U}$:

$$\mathbf{C} = \mathbb{E}_\rho \left[(\nabla_\mathbf{x} f)(\nabla_\mathbf{x} f)^T \right] = \mathbf{W}\mathbf{\Lambda}\mathbf{W}^T, \quad \mathbf{\Lambda} = \mathrm{diag}(\lambda_1, \dots, \lambda_d), \lambda_1 \geq \cdots \geq \lambda_d \geq 0, \quad \mathbf{W}^T\mathbf{W} = \mathbf{I} \tag{1}$$

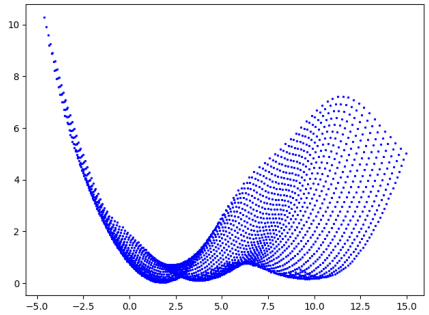
The goal is then to construct a Ridge function $g = g(\mathbf{y} = \mathbf{U}^T\mathbf{x})$ that depends only on the $r \ll d$ active variables, since we assume that the number of parameters d is too large for proper standard surrogate model construction. This function g is approximated by a response surface like Kriging [23] trained in the active subspace.

However, in our experience, complex aerodynamic optimisation cases can be multimodal,

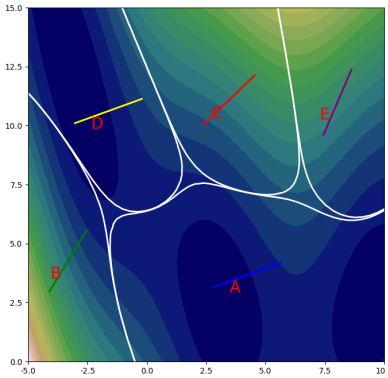
especially as the number of design parameters increases. This is confirmed by the literature [24, 25]. Further, we question the ability of a single global active subspace (a dimension reduction method based on gradient information) to accurately capture trends of multimodal objective functions, especially in high dimensional design spaces. Rather than use a single active subspace to capture information over the entire design space, we use Divide and Conquer principles and seek to locally reduce dimension. This way, local active subspaces capture objective function variations in different areas of the design space. Not only can this allow for smaller active subspaces (which reduces surrogate model complexity and cost), it helps surrogate-based optimisation, since this increases the likelihood that the function will indeed be invariant in the discarded spaces. A visual example is presented in figures 1b and 1d.



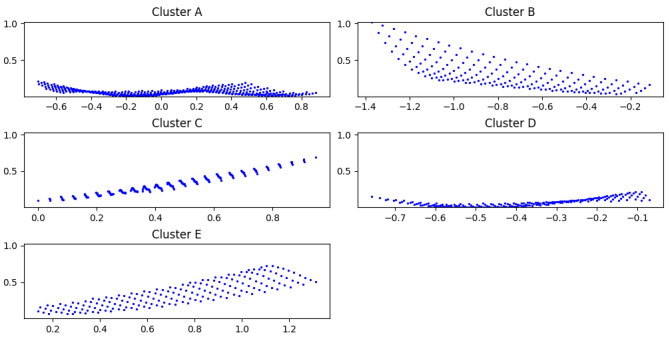
(a) Contour plot of Branin function, with active subspace direction (red).



(b) Summary plot of Branin function values projected onto an active subspace.



(c) Contour plot of the clustered Branin function. White lines represent boundaries between clusters, and colored segments are the local AS directions.



(d) Summary plots of Branin function values projected onto multiple local active subspaces.

Figure 1: Branin function ($f(x_1, x_2) = \left(x_2 - \frac{5.1}{4\pi}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left[\left(1 - \frac{1}{8\pi}\right)\cos(x_1) + 1\right] + \frac{5x_1+25}{15}$), on its defined domain: $x_1 \in [-5, 10]$ and $x_2 \in [0, 15]$, in the case of one and multiple local active subspaces. Cases of high variance (many function values plot to the same reduced coordinate) indicate incomplete capture of function topology.

These two figures represent the same function values, but projected respectively onto a single active subspace (1b) and a set of local active subspaces (1d). Notice how much less noise is present: there are much fewer values of active variables onto which multiple function values are projected. This means the local active subspaces capture function trends much more efficiently, which increases accuracy and simplifies the inverse-mapping back into the full space.

Question remains, how to partition the design space in a way that identifies potential multi-modality, using few queries of the expensive objective function?

Any clustering method needs to fulfill several requirements in order to qualify. (i) It must be an unsupervised method: we are looking for patterns in the data, since we do not know *a priori* how many regions of interest our objective function has. (ii) The method must not be data-hungry, since our function of interest is expensive to compute, and (iii) it needs to work even in large design spaces exceeding a hundred input parameters. Gaussian Mixture Models (GMM) [26] were chosen because their ability to make decisions based on function topology (looking at the joint correlation of inputs and outputs) fits our use-case particularly well. By considering the joint law of the inputs and outputs (X, Y), function values play a key role in splitting up the domain into clusters where the objective function exhibits similar trends.

$$\mathbf{z}_i = (\mathbf{x}_i, y_i) \in \mathbb{R}^{d+1}, \quad Z \sim \sum_{k=1}^{K} \alpha_k \, \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{2}$$

Once the parameters in eq.2 have been identified by the Expectation-Maximisation (EM) algorithm, we can calculate the probability of membership of every point to every cluster, assigning them to the cluster they are most likely to belong to. Local active subspaces are trained using the points of their cluster, and local response surfaces are trained inside these locally reduced spaces. This local dimension reduction was developed to facilitate surrogate model training: dimension reduction reduces the number of hyperparameters that must be determined, which reduces training cost and can even be beneficial to accuracy [9]. Clustering also means that each surrogate model is tasked with predicting a region of the design space, which is easier than fitting a complex function over an entire design space.

The global surrogate model is then assembled by constructing a linear combination of the local experts by adapting the classic formulation for Mixture of Experts [27]:

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^{K} \beta_k(\mathbf{x}) \, \hat{f}_k\big(\mathbf{y}_k = \mathbf{U}_k^T \mathbf{x}\big) \tag{3}$$

where $\beta = (\beta_1, \ldots, \beta_K)$ is the vector of the gating functions. Different gating networks are present in the literature [8, 28, 29, 30].

Previous work [21] showed that surrogate model performance was good, but some accuracy was lost at the boundaries between clusters. We attributed this error to extrapolation: each of the local experts is trained using data belonging to their cluster. It is conceivable that the very edges of each cluster are devoid of training points (illustration in figure 3), leading to extrapolation, which compromises accuracy [8]. In order to increase accuracy without the need to query new points, we developed an overlapping strategy, presented in section 3.

We now introduce the algorithm for Clustered Active Subspaces.

Algorithm 1: Complete Clustered Active Subspaces method

1 Construct an initial dataset of M samples $\mathcal{D} = (\mathbf{x}_i, y_i, \nabla_{\mathbf{x}} y_i)\,,\ \mathbf{x}_i \in \mathbb{R}^d,\ \mathbf{y}_i \in \mathbb{R},\ \nabla_{\mathbf{x}} y_i \in \mathbb{R}^d$.

2 Assume the joint data $Z = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_M, y_M)\}$, $\mathbf{z}_i \in \mathbb{R}^{d+1}$ follows a Gaussian mixture:

$$p\left(\mathbf{z} \mid \alpha_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right) = \sum_{k=1}^{K} \alpha_k \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \text{ with } \begin{cases} \boldsymbol{\mu}_k \in \mathbb{R}^{d+1}, \boldsymbol{\Sigma}_k \in \mathcal{M}_{d+1}(\mathbb{R}) \\ \sum_{k=1}^{K} \alpha_k = 1 \end{cases}$$

4 Identify Gaussian Mixture Model hyperparameters using the Expectation-Maximisation algorithm.

5 Perform hard clustering of training points using highest posterior probability of membership: for

6 $(\mathbf{x}_i, y_i) \in Z$:

$$j^* = \arg \max_{j=1,\dots,K} p\big(k = j \mid (X, Y) = (\mathbf{x}_i, y_i)\big), \quad i = 1, \dots, N$$

7 Train Support Vector Classifier using GMM labels (used in step 12).

8 **for** k *in* $[1, K]$ **do**

10 Compute active subspace by decomposition of local empirical gradient correlation matrix:

$$\mathbf{C}_k = \mathbb{E}_{\rho_k(\mathbf{x})}\left[\nabla f(\mathbf{x}) \nabla f(\mathbf{x})^T\right] \approx \hat{\mathbf{C}}_k = \frac{1}{N} \sum_{j=1}^{N} \nabla f(\mathbf{x}_j) \nabla f(\mathbf{x}_j)^T, \qquad N = |k|$$

11 Decompose matrix $\hat{\mathbf{C}}_k = \mathbf{W}_k \Lambda_k \mathbf{W}_k^T$ with $\mathbf{W}_k^T \mathbf{W}_k = \mathbf{I}_k$ and $\mathbf{W}_k = [\mathbf{U}_k, \mathbf{V}_k]$ from which we define the active variables $\mathbf{y}_k = \mathbf{U}_k^T \mathbf{x}$.

12 Compute overlapping region of current cluster (with its neighbours).

13 Add points located in the overlapping region to the training set of the current cluster.

14 Train local Kriging $\hat{f}_k(\mathbf{y}_k)$ using the points in the extended training set.

15 **end**

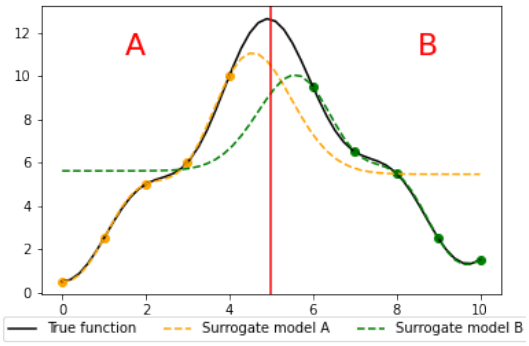
16 Recombine local experts to form a global surrogate model: $\hat{f}(\mathbf{x}) = \sum_{k=1}^{K} \beta_k(\mathbf{x}) \hat{f}_k(\mathbf{y}_k)$.

3 Overlapping

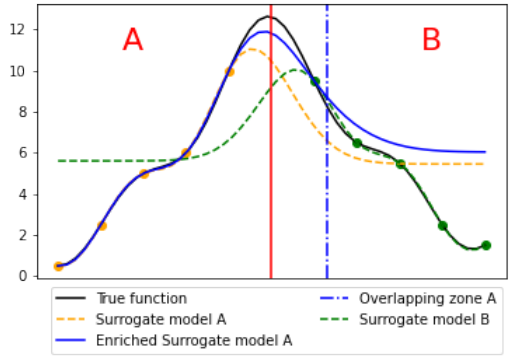
We seek to increase Clustered Active Subspaces surrogate model prediction accuracy without querying new, expensive, training points, by reducing the areas of extrapolation between clusters. We achieve this by extending the training set of each surrogate model using points from neighbouring clusters, such that the training bases now encompass these frontier areas, reducing the need for extrapolation and therefore improving predictive accuracy.

Our first idea was to further utilise the joint probability of membership computed by GMM, by adding the points with the highest probability of belonging to a certain cluster. In practice, setting the threshold for probability of membership proved to be a very sensitive parameter to optimise, since including too many points would increase computation time while increasing the possibility of adding irrelevant or even detrimental points to the training sets.

In order to relieve this limitation, we shift the objective, now seeking to formulate an overlapping method based on the Euclidean distance to the boundary (figure 3). The goal of the method is to add points nearby to the boundaries in order for the local experts' training basis to fully envelop the region they are tasked with predicting.



(a) True function and two local experts, each trained in their respective cluster.



(b) Enriching cluster A by adding a point from cluster B. The blue dash-dot line represents the overlapping zone of cluster A.

Figure 2: Demonstration of overlapping on a 2 cluster toy problem. Note only cluster A has been enriched, but notice how much better the recombined prediction (blue curve) is in the overlapped zone.

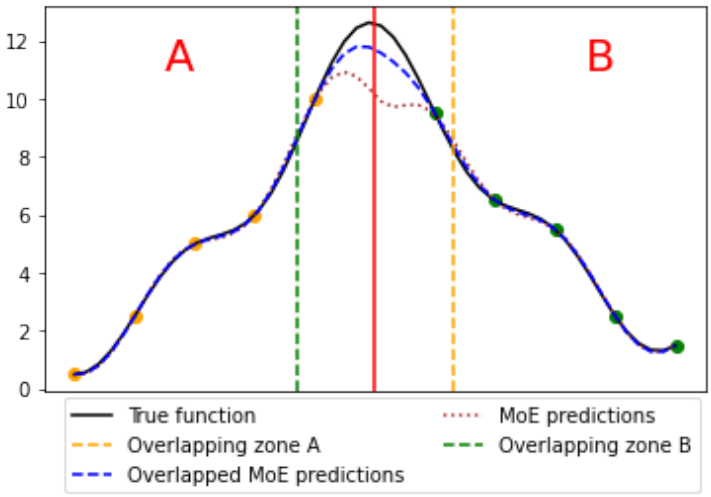


Figure 3: Demonstration of overlapping for Mixture of Experts (MoE) on a 2 cluster toy problem. Note how the overlapped MoE formulation (blue dashed curve) is more accurate than typical MoE (brown dotted curve) in the overlapped zone, while maintaining accuracy outside of it.

To that end, we need an explicit expression of the boundary between clusters, and then a way to compute a distance to said boundaries.

3.1 Support Vector Classifiers

Support Vector Classifiers (SVM) [31, 32] aim to identify the optimal separating hyperplane between points of different sets. SVC is traditionally a binary classifier, but extension to the

multiclass scenario is straightforward, ensuring SVC can be used for our situation, where $y_i \in [1, K]$ are the labels stemming from our GMM based classifier. Another valuable feature of SVMs is that they can be extended to deal with non-linearly separable data, by projecting the input data into a higher dimensional space. In that case, the *decision function* is of the form:

$$\mathcal{D}(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + b \tag{4}$$

where ϕ is a nonlinear mapping. This trick, known as the Kernel trick, allows for much better classification in the case of data which is tough to separate linearly. Classification is induced by the sign of $\mathcal{D}$, $G(\mathbf{x}) = \text{sign}\left[\phi \mathbf{w}^T(\mathbf{x}) + b\right]$. The boundary between clusters is where the decision function $\mathcal{D}(x)$ is null, and this is the definition of a multivariate implicit curve: $Z(\mathcal{D}) = \{\mathbf{x} : \mathcal{D}(\mathbf{x}) = 0\}\, \mathbf{x} \in \mathbb{R}^d, \mathcal{D} : \mathbb{R}^d \to \mathbb{R}$.

3.2 Computing the distance to the boundary

The distance between a DoE point $p \in \mathcal{X} \subset \mathbb{R}^d$ and this boundary is defined as the minimum of the distances from p to points in the zero set: $\delta\big(p, Z(\mathcal{D})\big) = \min\big(\|p - q\|; \mathcal{D}(q) = 0\big)$. Let p be a point such that $\|\nabla \mathcal{D}(p)\| \neq 0$ and let us expand $\mathcal{D}(q)$ in Taylor series up to first order in a neighbourhood of p:

$$\mathcal{D}(q) = \mathcal{D}(p) + \nabla \mathcal{D}(p)^T (q - p) + \mathcal{O}(\|q - p\|^2) \tag{5}$$

The generalisation to the k^{th} order is explained in [33]. For our purposes, the second order is sufficient, which amounts to solving for the positive root of equation 6, since a distance will always be positive.

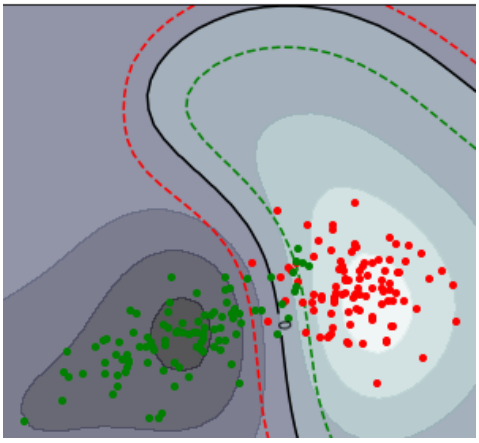
$$|\mathcal{D}(p)| - \|F_1\|\delta_2 - \|F_2\|\delta_2^2 = 0 \quad \longrightarrow \quad \delta_2 = \sqrt{\frac{-\|F_1\|^2}{4\|F_2\|^2} + \frac{|\mathcal{D}(p)|}{\|F_2\|}} - \frac{\|F_1\|}{2\|F_2\|} \tag{6}$$

$$\text{where } F_i = \left\{ \frac{D^\alpha \mathcal{D}(p)}{\sqrt{\alpha!}\sqrt{i!}} : |\alpha| = i,\ i = 1, 2 \right\}$$

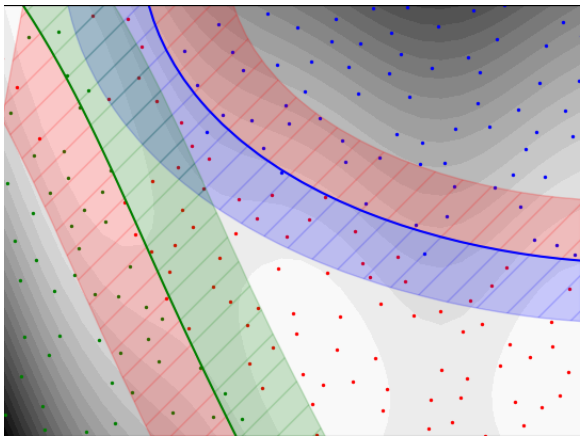
$\alpha = (\alpha_1, \alpha_2, ..., \alpha_n)$ is the multi-index which contains the collections of partial derivatives orders which sum to i, $n = A_d^i = \frac{d!}{(d-i)!}$, $\alpha! = \alpha_1! \times \alpha_2! \times ... \times \alpha_n!$, and F_i is the vector of all partial derivatives of order i in dimension d.

3.3 Illustration of supervised overlapping

Figures 4a and 4b illustrate supervised overlapping on 3 class problems. Figure 4a is intended to show how the overlapping width follows only the Decision Boundary and not the other isovalues of the Decision Function, and figure 4b showcases its use on the Branin function.



(a) Isovalues of the SVM Decision Function, with the black line marking Decision Boundary ($\mathcal{D}(\mathbf{x}) = 0$). The dashed lines follow the Decision Boundary, offset by the distance computed in eq.(6), representing the overlapping zones of each cluster.



(b) Overlapping regions are color-coded to show which clusters they are adding points to. For example, all green and blue points within the red shaded regions will be added to the training set of red cluster's surrogate model.

Figure 4: Supervised overlapping on a three class problem. Toy problem (left) and Branin function (right). Notice how the boundary of the overlapping region is at a constant distance from the cluster boundary, but does not follow the other isovalues of the nonlinear Decision Function.

This is exactly why we resorted to a Euclidean distance based method: the nonlinearity in the probability of membership function of GMM and the Decision Function of SVM do not allow for consistent overlapping regions, but this does. It is worth noting that this method is different from the one proposed in [34], as these authors were concerned with finding the largest possible zone devoid of points between clusters, whereas we seek to add points near the boundaries.

4 Results

Although most of the work so far has focused on methodology, we present some interesting results we have accumulated along the way. The two test cases are datasets featured in the Active Subspaces Toolbox for Python [35]. The first is a set of 1756 inviscid CFD runs on the 2D NACA0012 airfoil, with 18 design variables. The quantities of interest are lift and drag coefficients, as well as their gradients with regards to the 18 shape parameters. The second is a set of 300 CFD runs around the ONERA M6 3D wing, this time with 50 design variables. Again, the dataset contains lift and drag coefficients, as well as their respective gradients. In both cases below, the studied quantity of interest is the drag coefficient.

We use both test cases to compare the overlapped Clustered Active Subspaces method presented in this paper with two reference methods: a Kriging model trained in the original space, and a Kriging model trained in a global active subspace, as presented in the works of Constantine [15] and Lucaczyk et al. [22]. For comparison's sake, we also include the first iteration of the Clustered Active Subspaces method, without overlapping, as presented in [21].

4.1 NACA0012 airfoil with 18 design parameters

The first interesting conclusion we draw is that active subspaces are beneficial to kriging when used as a preprocessing method. We attribute this to better numerical conditioning, with active subspaces providing an isotropisation of the design space, which can help the maximimation of the likelihood of kriging hyperparameters, even without reducing dimension.

While 18 dimensions is not considered a large number of design parameters for aerodynamic shape optimisation, it is approaching the upper limit for standard kriging construction. This is corroborated by our results (table 1), which show kriging to be the worst performing method, especially for small training sets: there are not enough training points to accurately determine the hyperparameters of the kriging model. As the size of the training set grows, we see kriging perform comparatively better, since the ratio of training points to hyperparameters increases.

Method	**50 TP**	**100 TP**	**200 TP**	**500 TP**	**750 TP**
Kriging	$5.583{\cdot}10^{-3}$	$5.868{\cdot}10^{-3}$	$3.516{\cdot}10^{-3}$	$2.494{\cdot}10^{-3}$	$2.502{\cdot}10^{-3}$
AS + Kriging	$2.516{\cdot}10^{-3}$	$2.9{\cdot}10^{-3}$	$3.549{\cdot}10^{-3}$	$2.628{\cdot}10^{-3}$	$3.119{\cdot}10^{-3}$
CAS (no overlap)	$4.849{\cdot}10^{-2}$	$4.323{\cdot}10^{-3}$	$3.288{\cdot}10^{-3}$	$2.837{\cdot}10^{-3}$	$3.305{\cdot}10^{-3}$
CAS (overlap)	$4.402{\cdot}10^{-3}$	$3.323{\cdot}10^{-3}$	$2.903{\cdot}10^{-3}$	$2.389{\cdot}10^{-3}$	$2.212{\cdot}10^{-3}$

Table 1: Comparison of RMSE generalisation error on the 18 dimensional NACA airfoil CFD drag dataset.

Even on a relatively simple test case, it is beneficial to accept a small loss of information due to dimension reduction, in order to better estimate the parameters of the kriging model.

The Clustered Active Subspaces method is more accurate than standard kriging on small training sets, though slightly less effective than globally reduced kriging. However, its performance improves significantly as more data is added. Over the entire range of training set sizes, the new CAS method with overlapping presented in this paper outperforms its previous version, with improvements between 15% and 90%.

Note: The automatic selection of the optimal number of clusters is work in progress for the authors. However, while for the purposes of this paper we have constrained the CAS methods to have at least 2 clusters, we absolutely will let the method decide for itself if a single global active subspace is the way to go for certain datasets.

4.2 M6 wing with 50 design parameters

On the M6 wing dataset, which has a much lower number of training points compared to the number of dimensions, kriging is now comfortably the worst performing method. There is not enough information to accurately train the model. In addition, on this more complicated test case, the benefit of using several active subspaces instead of a single global active subspace becomes apparent.

For the entire range of training set sizes, both CAS versions outperform the single AS+Kriging method. The reason for this is that the CAS methods are more flexible, enabling the construction of active subspaces more suited to local function trends, which then allows for more accurate kriging prediction. Furthermore, the CAS methods allows for smaller local active subspaces,

further reducing the number of kriging hyperparameters to be optimised.

The only caveat is for very small numbers of training points, the local active subspaces and krigings do not have enough information to accurately train the different models.

Method	**50 Training points**	**100 Training points**	**200 Training points**
Kriging	$5.396{\cdot}10^{-3}$	$4.873{\cdot}10^{-3}$	$3.157{\cdot}10^{-3}$
AS + Kriging	$2.177{\cdot}10^{-3}$	$2.891{\cdot}10^{-3}$	$2.911{\cdot}10^{-3}$
CAS (no overlap)	$1.792{\cdot}10^{-2}$	$2.750{\cdot}10^{-3}$	$2.407{\cdot}10^{-3}$
CAS (overlap)	$1.190{\cdot}10^{-2}$	$2.296{\cdot}10^{-3}$	$2.378{\cdot}10^{-3}$

Table 2: Comparison of RMSE generalisation error on the 50 dimensional M6 wing CFD drag dataset.

5 Conclusion

This paper presents an improved Clustered Active Subspaces method applied to the high-dimensional objective functions of two aerodynamic shape optimisation problems. The method uses both clustering and dimension reduction in order to discover local trends in the quantity of interest, and facilitate surrogate model construction.

Unsupervised Gaussian Mixture clustering is used to partition the design space based on function topology, which then allows the discovery of local active subspaces to be built as linear combinations of the most influential input parameters. We train surrogate models in each of these locally reduced spaces, then combine them into a smooth global surrogate model of the quantity of interest. Our results show the method has good promise, being both more accurate and faster on two datasets of 18 and 50 design variables respectively.

Later work will couple this surrogate modelling method with a hybrid Bayesian/gradient-based enrichment strategy in order to perform global constrained aerodynamic shape optimisation in high dimensional design spaces.

References

[1] Marco Carini et al. "Towards industrial aero-structural aircraft optimization via coupled-adjoint derivatives". In: *AIAA Aviation 2021 Forum*. 2021. DOI: 10.2514/6.2021/3074.

[2] Gaetan K. W. Kenway, Joaquim R. R. A. Martins, and Graeme J. Kennedy. "Aerostructural optimization of the Common Research Model configuration". In: *15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*. 2014. DOI: 10.2514/6.2014-3274.

[3] Warren Hare Charles Audet. *Derivative Free and Bloackbox Optimization*. Springer Cham, 2017. ISBN: 978-3-319-68912-8. DOI: https://doi.org/10.1007/978-3-319-68913-5.

[4] K. Deb et al. "A fast and elitist multiobjective genetic algorithm: NSGA-II". In: *IEEE Transactions on Evolutionary Computation* 6.2 (2002), pp. 182–197. DOI: 10.1109/4235.996017.

[5] Nikolaus Hansen, Sibylle D. Müller, and Petros Koumoutsakos. "Reducing the Time Complexity of the Derandomized Evolution Strategy with Covariance Matrix Adaptation (CMA-ES)". In: *Evolutionary Computation* 11.1 (2003), pp. 1–18. DOI: 10.1162/106365603321828970.

[6] Jonas Močkus. "On Bayesian methods for seeking the extremum". In: *Optimization Techniques IFIP Technical Conference: Novosibirsk, July 1-7, 1974*. Springer. 1975, pp. 400–404.

[7] Michael J Sasena, Panos Y Papalambros, and Pierre Goovaerts. "The use of surrogate modeling algorithms to exploit disparities in function computation time within simulation-based optimization". In: *Constraints* 2.5 (2001).

[8] David Ginsbourger et al. "A note on the choice and the estimation of Kriging models for the analysis of deterministic computer experiments". In: *Applied Stochastic Models in Business and Industry* 25.2 (2009), pp. 115–131. DOI: 10.1002/asmb.741.

[9] David J. J. Toal, Neil W. Bressloff, and Andy J. Keane. "Kriging Hyperparameter Tuning Strategies". In: *AIAA Journal* (2008). DOI: 10.2514/1.34822.

[10] Paul G. Constantine. "Active Subspace Methods in Theory and Practice: Applications to Kriging Surfaces". In: *SIAM Journal on Scientific Computing* 36.4 (2014), pp. 1500–1524. DOI: 10.1137/130916138.

[11] Rohit Tripathy, Ilias Bilionis, and Marcial Gonzalez. "Gaussian Processes with Built-in Dimensionality Reduction: Applications in High-dimensional Uncertainty Propagation". In: *Journal of Computational Physics* 321.15 (2016), pp. 191–223.

[12] Paul Saves et al. "High-Dimensional Efficient Global Optimization Using Both Random and Supervised Embeddings". In: *AIAA Aviation Forum*. 2023. DOI: 10.2514/6.2023-4448.

[13] Mohammed Amine Bouhlel et al. "Improving Kriging surrogates of high-dimensional design models by Partial Least Squares dimension reduction". In: *Stuctural and Multidisciplinary Optimisation*. 53rd ser. (2016), pp. 935–952. DOI: 10.1007/s00158-015-1395-9.

[14] Mohammed Amine Bouhlel et al. "An Improved Approach for Estimating the Hyperparameters of the Kriging Model for High-Dimensional Problems through the Partial Least Squares Method". In: *Mathematical Problems in Engineering* 216 (2016). DOI: http://dx.doi.org/10.1155/2016/6723410.

[15] Paul G. Constantine. *Active Subspaces: Emerging ideas for Dimension Reduction in Parameter Studies. Emerging ideas for Dimension Reduction in Parameter Studies*. SIAM Spotlights, 2015.

[16] Francisco Palacios et al. "Stanford University Unstructured (SU2): An open-source integrated computational environment for multi-physics simulation and design". In: 2013-01. DOI: 10.2514/6.2013-287.

[17] Nathalie Bartoli et al. "Adaptive Modeling Stategy for Constrained Global Optimization with Application toAerodynamic Wing Design". In: *Aerospace Science and Technology* 90 (2019), pp. 85–102. DOI: https://doi.org/10.1016/j.ast.2019.03.041.

[18] Yoel Tenne. "An Optimization Algorithm Employing Multiple Metamodels and Optimizers". In: *International Journal of Automation and Computing* 10 (2013), pp. 227–241. DOI: 10.1007/s11633-013-0716-y.

[19] Sam Roweis and Lawrence Saul. "Nonlinear Dimensionality Reduction by Locally Linear Embedding". In: *Science* 290 (5500 2000), pp. 2323–2326. DOI: 10.1126/science.290.5500.2323.

[20] Gabriele Boncoraglio and Charbel Farhat. "Piecewise-Global Nonlinear Model Order Reduction for PDE-Constrained Optimization in High-Dimensional Parameter Spaces". In: *SIAM Journal on Scientific Computing* 44.4 (2022), pp. 2176–2203. DOI: 10.1137/21M1435343.

[21] Maxime Chapron et al. "Scalable Clustered Active Subspaces for Kriging in High Dimension". In: *EUROGEN 2023. 15th ECCOMAS Thematic Conference on Evolutionary and Deterministic Methods for Design, Optimization and Control*. Ed. by N. Gauger et al.

[22] Trent Lukaczyk et al. "Active Subspaces for Shape Optimization". In: 2014-01. DOI: 10.2514/6.2014-1171.

[23] C.E.Rasmussen and K. I. Williams. *Gaussian Processes for Machine Learning*. 2006.

[24] Nicolas P. Bons et al. "Multimodality in Aerodynamic Wing Design Optimization". In: *AIAA Journal* 57.3 (2019). DOI: https://doi.org/10.2514/1.J057294.

[25] Gregg Streuber and David W. Zingg. "A Parametric Study of Multimodality in Aerodynamic Shape Optimization of Wings". In: *2018 Multidisciplinary Analysis and Optimization Conference*. 2018. DOI: 10.2514/6.2018-3637. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2018-3637. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2018-3637.

[26] Douglas Reynolds. "Gaussian Mixture Models". In: *Encyclopedia of Biometric Recognition*. Springer, 2008. ISBN: 978-1-4899-7487-7. DOI: https://doi.org/10.1007/978-1-4899-7488-4-196.

[27] Roger Grosse and Nitish Srivastava. "Lecture 16 : Mixture Models". 2017.

[28] Rhea P. Liem and Joaquim R. R. A. Martins. "Surrogate Models and Mixtures of Experts in Aerodynamic Performance Prediction for Mission Analysis". In: *Aerospace Science and Technology* 43 (2015), pp. 126–151. DOI: https://doi.org/10.2514/6.2014-2301. URL: https://www.sciencedirect.com/science/article/pii/S1270963815000760.

[29] Dimitri Bettebghor et al. "Surrogate modeling approximation using a mixture of experts based on EM joint estimation". In: *Structural and Multidisciplinary Optimization* 43 (2011), pp. 243–259. DOI: 10.1007/s00158-010-0554-2.

[30] Junda Xiong, Xin Cai, and Jinglai Li. "Clustered active-subspace based local Gaussian Process emulator for high-dimensional and complex computer models". In: *Journal of Computational Physics* 450 (2022), p. 110840. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2021.110840. URL: https://www.sciencedirect.com/science/article/pii/S002199912100735X.

[31] Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik. "A training algorithm for optimal margin classifiers". In: *Proceedings of the fifth annual workshop on Computational learning theory*. 1992, pp. 144–152.

[32] Christopher Bishop. "Pattern Recognition and Machine Learning". In: vol. 16. Jan. 2006, pp. 140–155. DOI: 10.1117/1.2819119.

[33] Gabriel Taubin. "Distance Approximations for Rasterizing Implicit Curves". In: *ACM Transactions on Graphics* 13.1 (1994), pp. 3–42.

[34] Dimitri Bettebghor and François-Henri Leroy. "Overlapping radial basis function interpolants for spectrally accurate approximation of functions of eigenvalues with application to buckling of composite plates". In: *Computers & Mathematics with Applications* 67.10 (2014), pp. 1816–1836. ISSN: 0898-1221. DOI: https://doi.org/10.1016/j.camwa.2014.03.020. URL: https://www.sciencedirect.com/science/article/pii/S0898122114001539.

[35] Paul Constantine et al. *Python Active-subspaces Utility Library*. Ed. by Zenodo. 2016. DOI: https://doi.org/10.5281/zenodo.158941.