RESEARCH PAPER

Surrogate-based aerodynamic shape optimization with the active subspace method

Jichao Li[1] · Jinsheng Cai[1] · Kun Qu[1]

Abstract

Surrogate-based optimization is criticized in high-dimensional cases because it cannot scale well with the input dimension. In order to overcome this issue, we adopt a snapshot active subspace method to reduce the input dimension. A smoothing operation of samples is used to reduce the demand for snapshots in the construction of active subspaces. This operation significantly reduces the computational cost on the one hand, and on the other hand, it leads to more feasible subspaces. We use a 90~95% energy coverage criterion to define the dimension of the subspace. With this criterion, the surrogate-based airfoil optimization in the active subspace is both efficient and effective. We also validate this optimization approach in an ONERA M6 wing optimization case with 220 shape variables. Compared with original surrogate-based optimization, the new approach reduces the computational time by 70% and obtains a more practical design with a smaller drag.

Keywords Active subspace method · Surrogate-based optimization · High-dimensional optimization

1 Introduction

With the development of numerical computational methods, high-fidelity computational fluid dynamic (CFD) methods like the Reynolds-averaged Navier–Stokes (RANS) equations play an increasingly important role in aerodynamic analysis. Coupled with some optimization algorithms, RANS is more and more widely used in aerodynamic shape optimization and has accelerated the design of aircraft to a large extent.

Aerodynamic shape optimization with RANS is generally composed of two kinds of methods, gradient-based optimization (GBO) and gradient-free optimization. Gradient-based optimization uses gradient information to guide the searching direction. With the adjoint method (Jameson 1988; Mader et al. 2008; Lyu et al. 2013), accurate gradients

can be obtained with a similar cost of the CFD simulation. Thus, gradient-based optimization can efficiently solve aerodynamic shape optimization problems strictly subject to both aerodynamic and geometry constraints. Gradient-free optimization uses gradient-free algorithms to search the design space. Due to the high computational cost of RANS analyses, gradient-free optimization is usually coupled with surrogate models to reduce the cost. In this case, it is also referred to as surrogate-based optimization (SBO) (Queipo et al. 2005; Han and Zhang 2012). Kinds of surrogate models including Kriging (Krige 1951), radial basis function, and support vector method can be used in SBO. These models are trained by some initial samples in advance and updated step by step in the optimization procedure. Gradient information is not necessarily required in SBO unless the surrogate is enhanced by gradients. So this kind of method is popular, especially when the RANS solver is a black-box tool and it is difficult to obtain accurate gradient.

However, both kinds of methods need modifications. On the one hand, GBO is criticized due to the inherent local convergence, in spite of its high efficiency. Chernukhin and Zingg (2013) showed that multimodality existed in the aerodynamic shape optimization of a three-dimensional configuration. Recent researches (Bons et al. 2017; Poole et al. 2017) led by AIAA Aerodynamic Design Optimization Discussion Group also demonstrated multimodality in
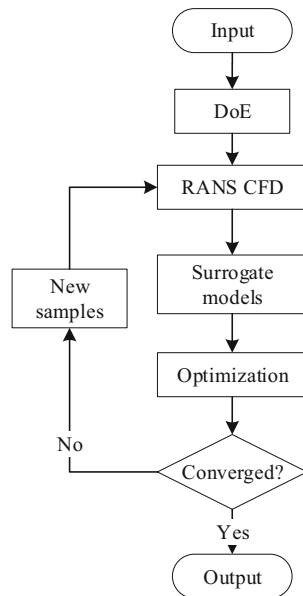

Responsible Editor: Nestor V Queipo

✉ Jichao Li
cfdljc@gmail.com

1 National Key Laboratory of Aerodynamic Design and Research, School of Aeronautics, Northwestern Polytechnical University, 710072 Xi'an, China
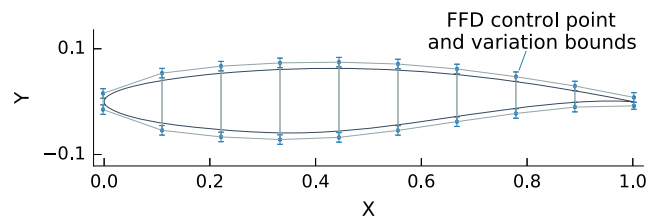
Fig. 1 The flowchart of SBO

a wing design problem. To prevent from being stuck to local minima, GBO could be coupled with a multi-start strategy (Streuber and Zingg 2017; Chernukhin and Zingg
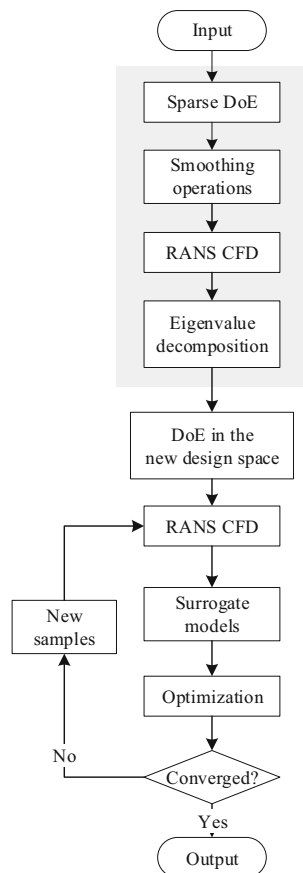
**Fig. 2** The flowchart of ASM-SBO

Fig. 3 The FFD control box for the airfoil optimization

2013), but it seriously affects the efficiency. On the other hand, although SBO has an obvious advantage in global searching capability, it is rather sensitive to the input dimension. SBO can work well in airfoil shape design problems (Leifsson and Koziel 2010; Wu et al. 2017) where tens of design variables are generally required. Li et al. (2018a) constructed a universal surrogate model for almost all existing airfoils to support real-time airfoil design optimization. While for design optimization of aircraft wings and more complex configurations, hundreds of design variables are usually required. In the literature, we can find that Liu et al. (2016) used SBO in a wing shape optimization with 48 variables, and Han et al. (2018) used SBO in a wing-body configuration with 80 design variables. However, it is rare to see that more shape design variables are used in SBO. Because it is difficult for surrogate models to handle so many design variables, and meanwhile the global searching algorithm might be problematic in high-dimensional problems. This issue is called the curse of dimensionality. But a high-dimensional parameterization is always required in engineering optimization, even for preliminary design. So, an approach that can efficiently extend SBO to high-dimensional optimization problems is of great interest.

The active subspace method (ASM) (Constantine et al. 2014) provides an approach to the input dimension reduction, and it has been applied in the aerodynamic shape optimization of an aircraft wing (Lukaczyk et al. 2014) and the aerodynamic shape analysis for a vehicle design (Othmer et al. 2016). In Lukaczyk et al.'s (2014) work, ASM was mapped to the original design space and then coupled with SBO for the optimization of the ONERA M6 wing with 50 design variables. The result was promising but the optimization with ASM was not impressively efficient, because the derivation of ASM vectors in their work was

Table 1 Optimization problem statement in the airfoil case

	Functions	Quantity	Description
Minimize	C_d	1	Drag coefficient
with respect to	α	1	Angle of attack
	$\mathbf{x}_{shape}$	18	y perturbation of FFD control points
Subject to	$C_l \geq 0.82$	1	Lift constraint
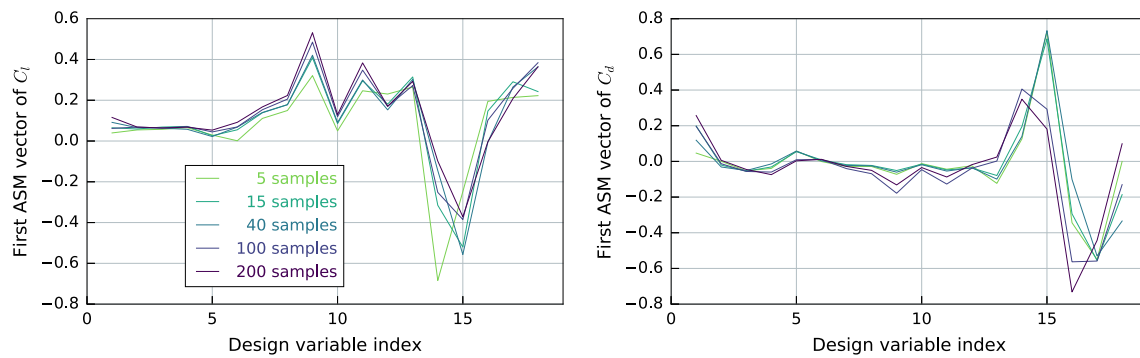
Fig. 4 ASM vectors obtained from samples generated by LHS

time-consuming, which included gradient evaluations of 300 samples. Moreover, mapping the active subspace back to the original design space brought new uncertainty to the optimization and further affected the efficiency.

Namura et al. (2017) proposed a Kriging-estimated method to obtain the ASM vectors in order to reduce the computational cost. It was validated in several problems including an aerodynamic shape optimization problem in a subsonic incompressible viscous flow with XFOIL. This method is especially useful in black-box problems where the accurate gradient cannot be obtained. Nevertheless, its application in the transonic flow regime with hundreds of design variables has not been tested. In this circumstance, the aerodynamic functions are much more nonlinear and it would be difficult to accurately estimate the gradient by surrogate models.

In this paper, we adopt a snapshot ASM in the dimension reduction of shape design variables and make SBO work efficiently in a transonic wing shape design problem with 220 shape design variables. We find that it is vital to smooth the sampling shapes before deriving ASM vectors. After the smoothing operation, the most active vectors are not sensitive to the number of samples, so we can use a small number of samples. This dramatically reduces the computational cost. Unlike former researches, we optimize the aerodynamic shape only in the active subspace in order to improve the optimization efficiency. We use an airfoil design case with 18 shape design variables and a transonic wing design case with 220 shape design variables

to demonstrate the efficiency of the proposed method by comparing it with the regular SBO.

The rest of this paper is organized as follows: First, we introduce the regular SBO with a Kriging model in Section 2.1. Then, we introduce the snapshot ASM used in this work and present a new SBO approach coupled with ASM in Section 2.2. We use a transonic airfoil design case to show the importance of the smoothing operation in Section 3.1, and propose our criterion to guide how many ASM vectors should be used in Section 3.2. Then, we validate this approach in a drag minimization case of a transonic wing with 220 shape design variables in Section 4. We end with the conclusions in Section 5.

2 Methodology

2.1 Surrogate-based optimization

Surrogate models provide a low computational cost approach to time-consuming high-fidelity models. In aerodynamic optimization with potential multimodality (Jones et al. 1998; Han et al. 2018; Andrés et al. 2012), they are always coupled with a gradient-free algorithm like the genetic algorithm or particle swarm optimization (PSO) method (Eberhart and Kennedy 1995). For unimodal problems, they can also be coupled with a gradient-based algorithm to accelerate the optimization (Li et al. 2018a). Kriging is a famous surrogate model proposed by Krige (1951), and it
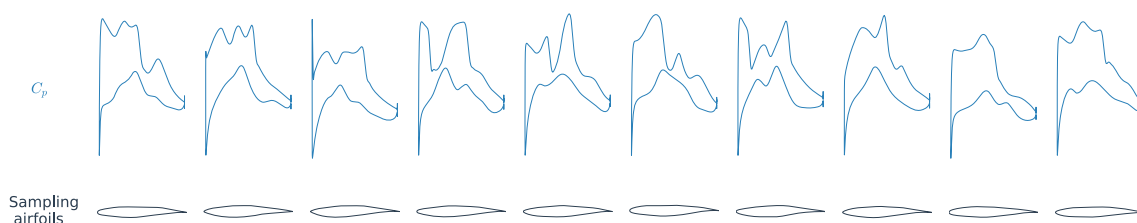
Fig. 5 C_p distributions of ten sampling airfoils

has been widely applied in aerospace design problems (Liu et al. 2016; Han et al. 2018).

For a system with input $\mathbf{x}$ and output y, $\mathbf{y}_s = (y^{(1)}, y^{(2)}, \ldots, y^{(n_s)})^{\mathrm{T}}$ is the vector of observed outputs on input points $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(n_s)}$. Based on the derivation of previous researches (Han and Zhang 2012; Liu et al. 2016), the Kriging prediction $\hat{y}$ of the system output function with respect to the input $\mathbf{x}$ can be expressed as

$$\hat{y}(\mathbf{x}) = \hat{\beta}_0 + \mathbf{r}(\mathbf{x})^{\mathrm{T}} \mathbf{R}^{-1} \left(\mathbf{y}_s - \mathbf{1}\hat{\beta}_0 \right), \tag{1}$$

where $\hat{\beta}_0 = (\mathbf{1}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{1})^{-1}\mathbf{R}^{-1}\mathbf{y}_s$, and $\mathbf{1} \in \mathbb{R}^{n_s}$ is a column vector filled with ones. $\mathbf{R} := [R(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})]_{ij} \in \mathbb{R}^{n_s \times n_s}$, and $\mathbf{r}(\mathbf{x}) := [R(\mathbf{x}^{(i)}, \mathbf{x})]_i \in \mathbb{R}^{n_s}$. $R(\mathbf{x}, \mathbf{x}')$ is a spatial function of $\mathbf{x}$ and $\mathbf{x}'$ that depends on the distance between them, which is the kernel of Kriging.

Apart from the prediction, Kriging models can provide the mean squared error $s^2(\mathbf{x})$ of the prediction.

$$s^2(\mathbf{x}) = \hat{\sigma}^2 \left[1 - \mathbf{r}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{r} + \left(1 - \mathbf{1}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{r} \right)^2 / \left(\mathbf{1}^{\mathrm{T}}\mathbf{R}\mathbf{1} \right) \right], \tag{2}$$

where $\hat{\sigma}^2 = (\mathbf{y}_s - \mathbf{1}\hat{\beta}_0)^{\mathrm{T}}\mathbf{R}^{-1}(\mathbf{y}_s - \mathbf{1}\hat{\beta}_0)/n_s$.

In SBO, the surrogate is trained by initial samples infilling the design space. These samples are generated in the design of experiments (DoE) step using a sampling method, e.g., the Latin hypercube sampling(LHS) method (McKay et al. 1979). Then, they are evaluated by CFD. In this paper, we use a RANS solver coupled with the Spalart-Allmaras turbulence model in CFD simulations.

We use PSO coupled with Kriging models in SBO. Because the surrogate trained by initial samples is usually not accurate enough, we have to modify it and run optimizations cycle by cycle before we eventually find the optimum. In each cycle, new samples are added. Many infilling criteria (Liu et al. 2016) have been proposed to improve the efficiency and global searching capability. We adopt a robust criterion introduced by Jones et al. (1998) in the efficient global optimization (EGO) method. Under this criterion, we add two samples in each cycle—one by minimizing the objective function with the surrogates and the other one by maximizing the expected improvement (EI) function.

$$E[I(\mathbf{x})] = \begin{cases} \left(y_{\min} - \hat{y}(\mathbf{x}) \right) \Phi \left(\frac{y_{\min} - \hat{y}(\mathbf{x})}{s(\mathbf{x})} \right) + \\ s(\mathbf{x}) \phi \left(\frac{y_{\min} - \hat{y}(\mathbf{x})}{s(\mathbf{x})} \right) & \text{if } s > 0 \\ 0 & \text{if } s = 0 \end{cases} \tag{3}$$

This equation can be extended to a constrained form when the Kriging prediction is involved in constraint functions. A detailed introduction can be found in Parr et al. (2012). The procedure of SBO is shown in Fig. 1.

2.2 Surrogate-based optimization with the active subspace method

ASM is an approach to reducing the input dimension of high-dimensional problems. Unlike a qualitative evaluation that finds the most important design variables, ASM provides combination patterns of all design variables that affect the output function much. Each of these combinations can be written as a vector, and perturbing the inputs along these vectors changes the output function more than perturbing inputs along other vectors. These vectors can span the active subspace of the high-dimensional problem, and the output function is more active in this subspace than in others. It is promising to construct a surrogate in the active subspace rather than the original design space, because it protects SBO from the curse of dimensionality.

We follow the derivation of ASM by Constantine et al. (2014). For a scalar output function $f = f(\mathbf{x})$ with n dimensional input $\mathbf{x} \in \mathbb{R}^n$, in order to construct the active subspace, an estimation of the covariance of the gradient in the design space is required.

$$\mathbf{C} = \mathbb{E} \left[\nabla f \nabla f^{\mathrm{T}} \right], \tag{4}$$

where $\mathbb{E}[\cdot]$ is the expectation operator that denotes integration against $\nabla f \nabla f^{\mathrm{T}}$ in the design space. ∇f means the gradient vector of f with respect to the input $\mathbf{x}$. In practice, a snapshot method with a sparse sampling in the design space can be used to approximate the elements in $\mathbf{C}$. First, M samples, $\mathbf{x}_i$, $i = 1, \ldots, M$, are selected in the design space and their gradients to the input are solved. Then, we have

$$\mathbf{C} \approx \frac{1}{M} \sum_{i=1}^{M} \nabla f_i \nabla f_i^{\mathrm{T}}, \tag{5}$$

where $\nabla f_i = \nabla f(\mathbf{x}_i)$.

We conduct an eigenvalue decomposition on $\mathbf{C}$ to identify the active vectors of the original input space, where we have

$$\mathbf{C} = \mathbf{W}\Lambda\mathbf{W}^{\mathrm{T}}. \tag{6}$$

$\mathbf{W}$ is a $n \times n$ matrix of eigenvectors and Λ is a diagonal matrix of eigenvalues. The diagonal elements (eigenvalues) of Λ are in descending order, and the columns of $\mathbf{W}$ are corresponding to eigenvalues from largest to smallest. Generally, the design space is only active along directions of several eigenvectors with maximal eigenvalues, e.g., the first several columns of $\mathbf{W}$. We refer to these eigenvectors as ASM vectors of the output function f in (7).

$$\mathbf{V}_{\mathrm{ASM}}^{f} = \left[\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_{n_{\mathrm{ASM}}} \right], \tag{7}$$

where $\mathbf{w}_i$ is the ith column in $\mathbf{W}$.

These ASM vectors construct the basis of the active subspace, which is denoted as $\mathbf{U}$. For the problem with only one scalar output f, we have

$$\mathbf{U} = \mathbf{V}_{\text{ASM}}^f. \tag{8}$$

For a problem with more than one scalar output, the active subspace space is constructed by the ASM vectors of each output. For example, there are two scalar outputs, coefficients of lift (C_l) and drag (C_d), in the aerodynamic problems discussed in this paper, so the basis of the active subspace $\mathbf{U}$ contains ASM vectors of both outputs, which is expressed in (9).

$$\mathbf{U} = \left[\mathbf{V}_{\text{ASM}}^{C_l}, \mathbf{V}_{\text{ASM}}^{C_d} \right]. \tag{9}$$

The coefficients of ASM vectors, $\mathbf{x}_{\text{ASM}}$, cooperate together to have an global influence on the original design variables $\mathbf{x}$. We use (10) to map the relationship between the subspace and the original design space, and it will not involve unmeasured uncertainty.

$$\mathbf{x} = \mathbf{U}\mathbf{x}_{\text{ASM}}. \tag{10}$$

For a high-dimensional aerodynamic problem, it can be found that the coefficients of lift and drag are mostly active in even one or two directions (Lukaczyk et al. 2014). The initial sparse samples can be generated by a sampling method like LHS, which makes the samples infill the design space evenly. But it is not guaranteed that these samples correspond to reasonable aerodynamic shapes. As shown in Fig. 5, small non-smooth shapes of original sampling airfoils lead to large oscillations in their C_P distributions. It would be noisy to directly use these samples in the derivation of ASM. So we perform a smoothing operation on the samples before evaluating their gradients. The smoothing algorithm with the free-form deformation (FFD) parameterization method is shown in Fig. 6. We discuss the influence of non-smooth sampling shapes and the necessity of the smoothing operation in Section 3.1.

We integrate the ASM method into SBO, which is refered to as surrogate-based optimization with the active subspace method (ASM-SBO) in this paper, and Fig. 2 shows the procedure of this ASM-SBO approach. The first step in ASM-SBO is to determine the active subspace, which contains a sparse DoE, smoothing operations, CFD evaluations, and eigenvalue decompositions using (6). In Fig. 2, the background of this step is marked with gray. Afterwards, the new design space is constructed by the active subspace and flow condition variables, e.g., the angle of attack. $\mathbf{x}_{\text{ASM}}$ is the vector of new shape design variables, which influence the original shape variables by (10). The bounds of each ASM vector are determined by the size of the original design space, i.e., the bounds of FFD control points. We perform another DoE in the new design space to train the surrogate models. Then, we conduct the optimization in this new design space with the same sample infilling criterion. Compared with SBO, ASM-SBO has a much smaller number of design variables. We discuss the selection of the number of ASM vectors in Section 3.2.

3 Transonic airfoil shape optimization

We first test ASM-SBO in an airfoil shape optimization case. In Section 3.1, we show that smoothing the shape of samples in ASM derivations is vital to the success of optimizations. In addition, we find that ASM vectors are not sensitive to the number of samples after the smoothing operation. In Section 3.2, we propose a criterion to select a suitable number of ASM vectors to make ASM-SBO efficient and effective.

The baseline airfoil of this case is RAE 2822. The Mach number is 0.73, and the Reynolds number is 6.5×10^6. The angle of attack is an independent variable, and it can vary among the range of $[2.0°, 3.0°]$ in the optimization. We use a 10×2 FFD box to control the airfoil shape which is shown in Fig. 3. The baseline airfoil and the variation bounds of each FFD control point are also shown in Fig. 3. We make the two FFD points around the leading edge scale together (in the opposite direction) in order to ensure the y coordinate of the leading edge is approximately equal to 0.0, and we also conduct the same operation to the two FFD points around the trailing edge. Thus, there are 18 independent shape variables in this case. The description of this optimization problem is shown in Table 1.

3.1 Smoothing operations in the construction of the active subspace

Among existing sampling methods, the LHS method is a popular one and often used to construct computer experiments for Monte Carlo integration, so we use it to generate snapshots. To fully study the design space, it is better to use a large number of samples. Lukaczyk et al. (2014) used 300 snapshots in a 50 dimensional aerodynamic case. Each sample calls for one CFD analysis and one gradient evaluation for each aerodynamic function. So the more samples used, the more computational cost this task requires, and then the less efficient the optimization will be.

In order to have a deeper insight of how many samples to use, we derive ASM vectors from different numbers of samples and evaluate their differences. We use 5, 15, 40, 100, and 200 samples in this test, and Fig. 4 shows the first ASM vectors for C_l and C_d derived from them. Although outlines of ASM vectors are similar, there is an obvious difference among them. The discrepancy between the results of 100 and 200 samples is not smaller than others,

Algorithm Laplacian smoothing algorithm

    Input: $\mathbf{y}_{\text{sampling}}, \varepsilon_{\mathbf{g}}$

    Output: $\mathbf{y}_{\text{smoothing}}$

Examples

1: **procedure** LAPLACIAN($\mathbf{y}_{\text{sampling}}, \varepsilon_{\mathbf{g}}$)

2:    $\mathbf{y}_{\text{soomthing}} = \mathbf{y}_{\text{sampling}}$

3:    $\varepsilon = 0.0$

4:    $\mathbf{V}_{\text{sampling}} = \mathscr{F}\left(\mathbf{y}_{\text{sampling}}\right)$ ⇨ Compute the shape manipulated by the initial sampling FFD

5:    **while** $\varepsilon < \varepsilon_g$ **do**

6:      **for** each FFD point i **do**

7:        $y^i_{\text{smoothing}} = \theta y^i_{\text{smoothing}} + \frac{1.0-\theta}{n_{\text{neighbor}}}\sum_{j=1}^{n_{\text{neighbor}}} y^j_{\text{smoothing}}$

8:        $(\theta = 0.5$ in this paper$)$ ⇨ Laplacian smoothing

9:      **for** each FFD point i **do**

10:        **if** this point is leading or trailing edge **then**

11:          find the pair of $i \rightarrow k$

12:          $y^i_{\text{smoothing}} + y^k_{\text{smoothing}} = 0.0$ ⇨ Leading/trailing edges

13:        **else**

14:          continue

15:      $\mathbf{V}_{\text{smoothing}} = \mathscr{F}\left(\mathbf{y}_{\text{smoothing}}\right)$

16:      $\varepsilon = \frac{||\mathbf{V}_{\text{smoothing}} - \mathbf{V}_{\text{sampling}}||}{||\mathbf{V}_{\text{sampling}}||}$

17:    **return** $\mathbf{y}_{\text{smoothing}}$
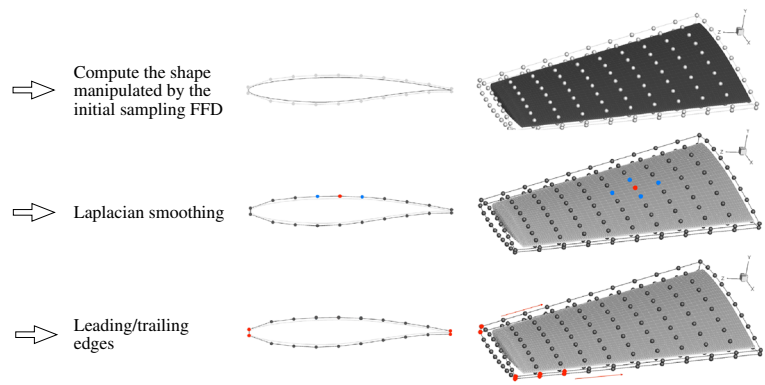
Fig. 6 Laplacian smoothing algorithm of sampling shapes controlled by FFD

so we cannot conclude that it is enough to use 100 or 200 samples in this case.

In the transonic flow regime, the upper surface of an airfoil is usually associated with shock waves. We further investigate the C_p distributions of these samples, and 10 of them are illustrated in Fig. 5. In the lower part of Fig. 5, airfoil shapes are also shown. We can find that the sampling airfoils are a bit non-smooth. This leads to severe oscillations of C_p and complex shock waves on the upper surfaces. Since C_l and C_d are sensitive to shock waves, the gradient to the design variables in the shock waves domain may become abnormally large and the influence of these design variables might be overvalued. Design variables with indexes between 10 and 18 control the shape of the upper surface, and we can see significant larger variances among these variables in Fig. 4. Distinct gradient values are associated to different samples due to oscillations of C_p. We do not see a convergence in ASM vectors even when 200 samples are used. Thus, we could conclude that the gradient information is "polluted" by the noise from complex shock waves due to the non-smooth shapes of the samples.

To address this issue, we conduct a smoothing operation on the samples to ensure that variations of neighbor FFD control points are similar. We use the Laplacian smoothing algorithm in this operation, which is an algorithm usually used to smooth a polygonal mesh. As we can see in Fig. 6, the smoothing algorithm contains two loops, and the outer loop is terminated by ϵ which represents the

change magnitude of the smoothed shape. We use a criterion of $\epsilon_g = 0.035$ to limit this magnitude. In the inner loop, all vertices are smoothed sequentially, and this sequential smoothing is based on the updated values of the adjacent vertices. For each vertex y_i which stands for the y coordinate of the ith FFD point, we update its value by

$$y_i = \theta y_i + \frac{(1.0 - \theta)}{n_{\text{neighbor}}} \sum_{j=1}^{n_{\text{neighbor}}} y_j, \tag{11}$$

where n_{neighbor} is the number of adjacent vertices to vertex y_i, and $0.0 < \theta < 1.0$. n_{neighbor} is two in this two-dimensional case and four in the three-dimensional wing shape optimization case. For the vertices at the leading and trailing edges, we conduct special operations to ensure the corresponding FFD points move in the opposite direction with the same magnitude. Note that the smoothing order in the inner loop can influence the smoothing results, so the same order should be kept in smoothing all samples. The FFD points are in the Plot3d format [1], and we use the consistent I-J-K cycling sequence to update their values in the inner loop.

θ in (11) is a parameter to control the smoothing operation speed. To select a suitable θ, we investigate the smoothing results from five different θ values, 0.1, 0.25, 0.5, 0.75, and 0.9. In the test of smoothing 30 LHS airfoils,

[1] https://www.grc.nasa.gov/www/wind/valid/plot3d.html

Fig. 7 Comparison of
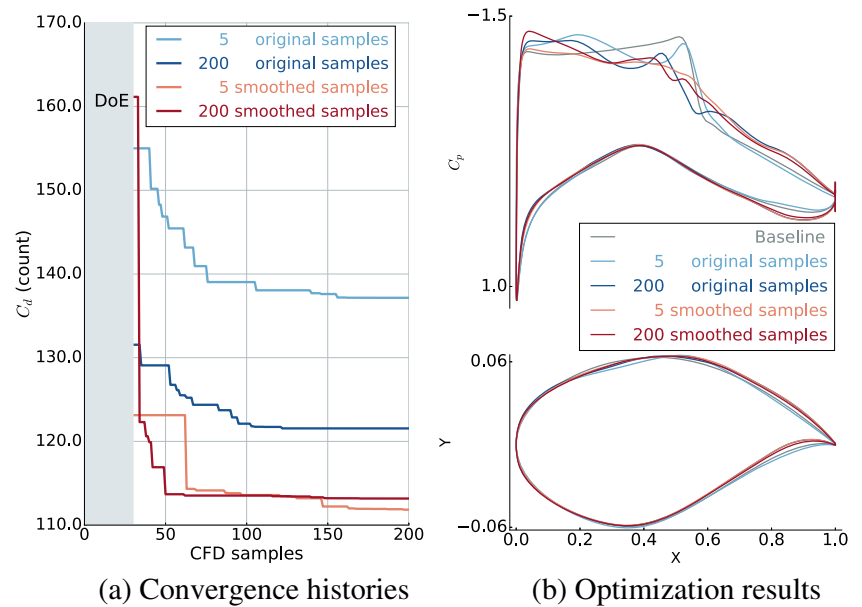smoothing results with different
θ in (11)
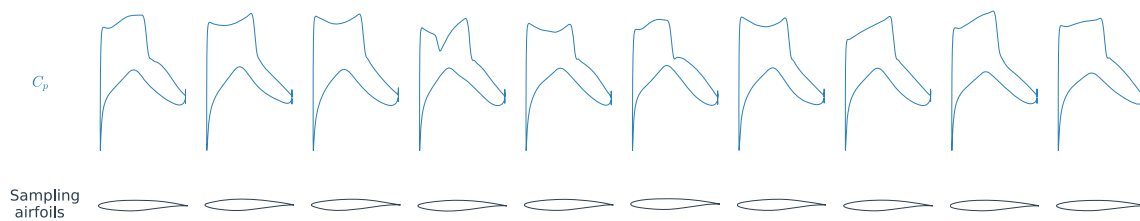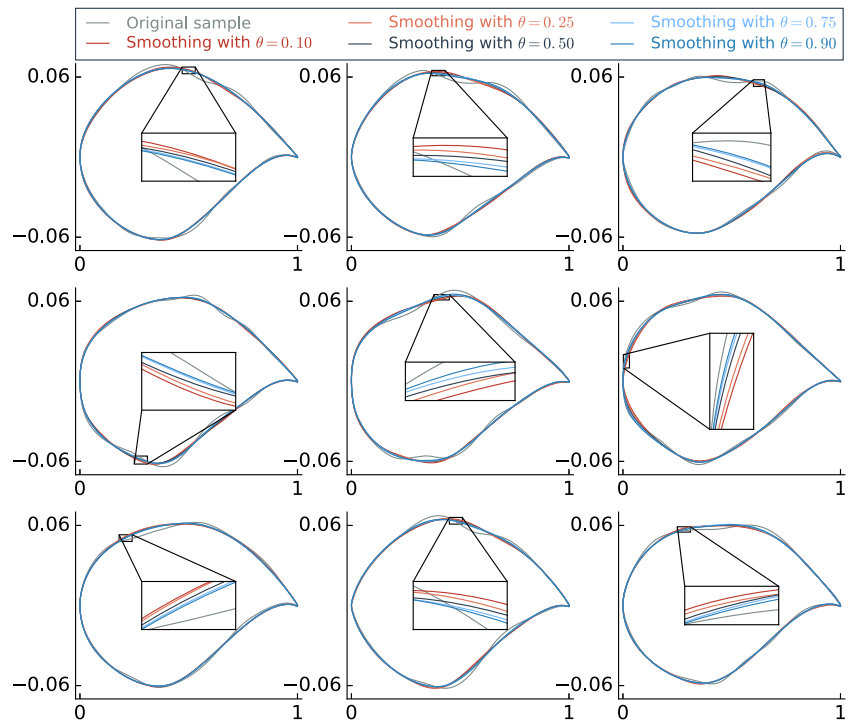
Fig. 8 C_p distributions of the ten smoothed airfoils. Smoothing operation remarkably reduces the oscillations.
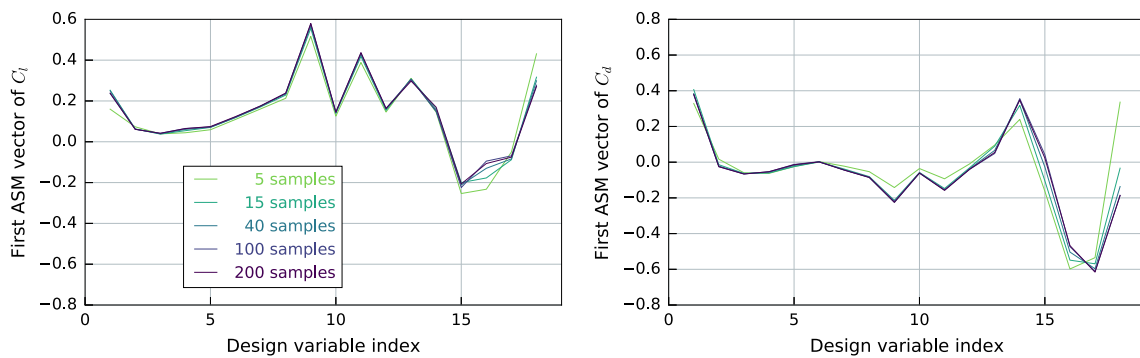
Fig. 9 ASM vectors obtained from LHS samples with smoothing operations

Fig. 10 **a** Convergence histories and **b** optimization results with different ASM vectors

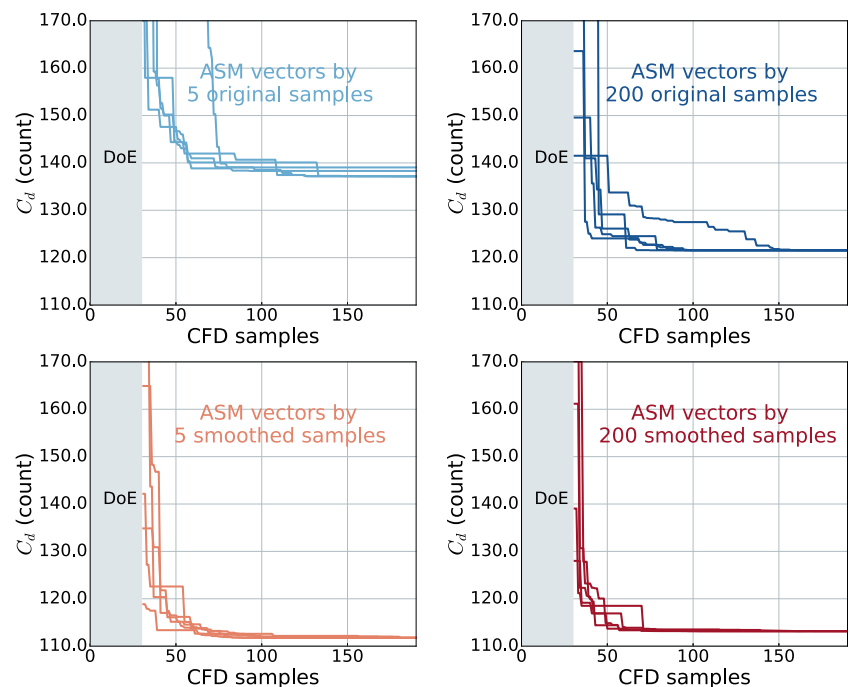(a) Convergence histories

(b) Optimization results

the average iteration numbers in the outer loop with the five θ values are 1.3, 1.6, 2.6, 5.6, and 14.2, and Fig. 7 shows the comparison of nine smoothed airfoils. A large θ, e.g., 0.9 or 0.75, slows down the operation so more smoothing iterations are used. We also find that the smoothed shapes with a larger θ are closer to the original sampling shapes, which means that the smoothing operation is a bit less effective. However, using a small θ, e.g., 0.1 or 0.25, accelerates the smoothing operation too much. In some cases, it only takes one iteration in the outer loop. This is too aggressive, because the change of the smoothed airfoils
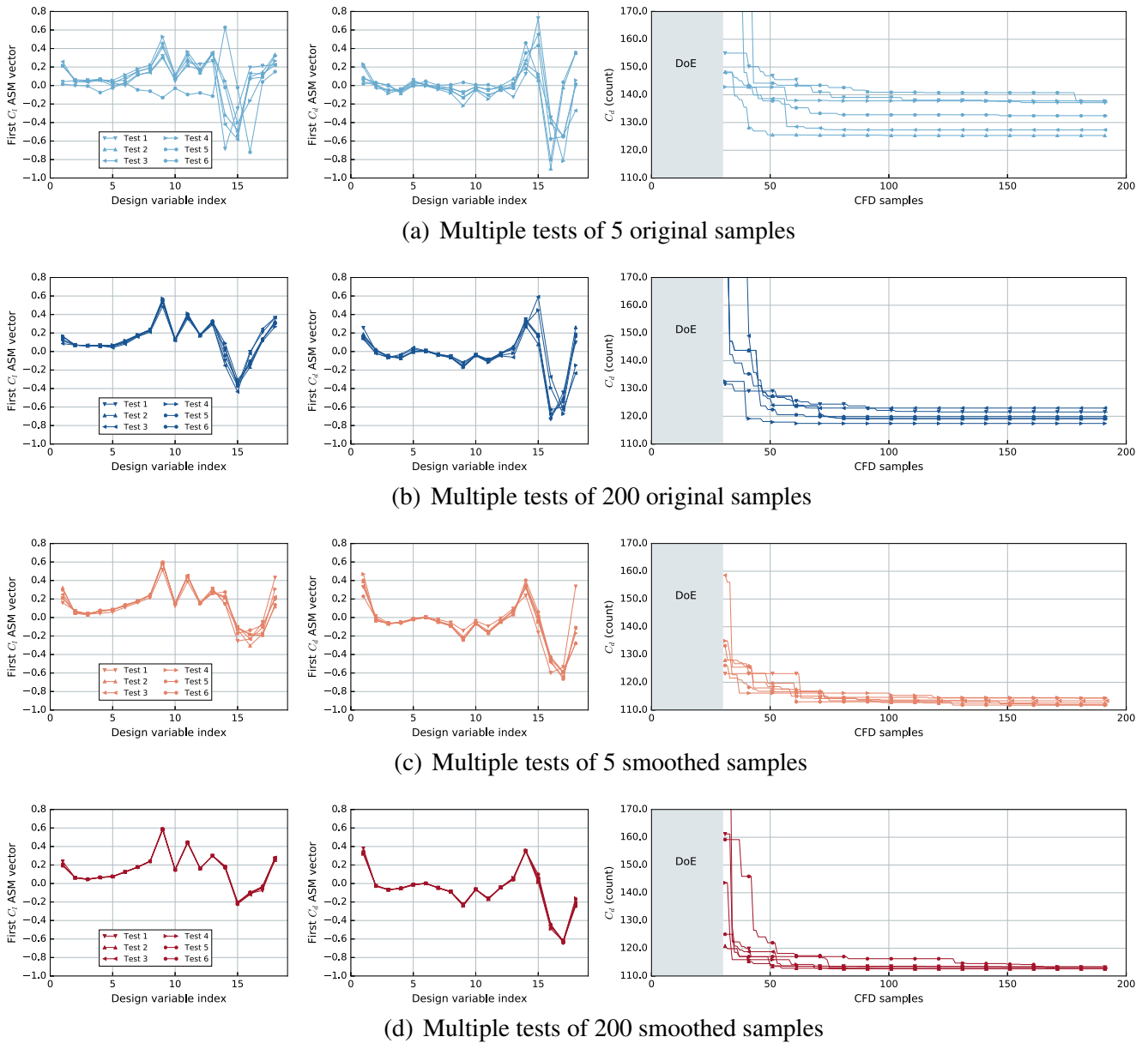
can be much larger than the limitation. Considering both concerns, we use $\theta = 0.5$ in this paper.

Figure 8 shows the smoothed sampling airfoils and their C_p distributions. We can see that their shapes become smoother and the C_p distributions are more reasonable and closer to practical ones. We conduct this smoothing operation to all the samples used in Fig. 4. Then, we perform CFD analyses and gradient evaluations. ASM vectors derived from smoothed samples are shown in Fig. 9. Compared with those derived from original samples, ASM vectors become much closer to each other, which implies

Fig. 11 Optimization with different DoEs using different ASM vectors

(a) Multiple tests of 5 original samples

(b) Multiple tests of 200 original samples

(c) Multiple tests of 5 smoothed samples

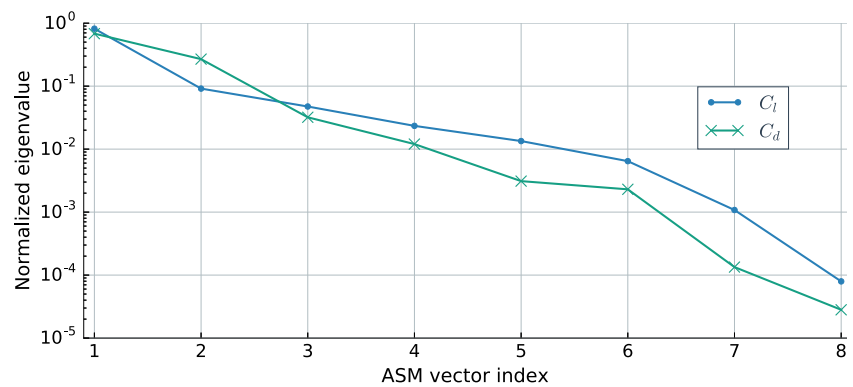(d) Multiple tests of 200 smoothed samples

Fig. 12 ASM vectors with different samples and corresponding optimization histories. **a** Multiple tests of 5 original samples. **b** Multiple tests of 200 original samples. **c** Multiple tests of 5 smoothed samples. **d** Multiple tests of 200 smoothed samples

Fig. 13 Eigenvalue decay for C_l and C_d active subspaces

**Table 2** The number of C_l and C_d ASM vectors used in each combination

	Combo 1	Combo 2	Combo 3	Combo 4	Combo 5
No. of C_l ASM vectors (energy)	1 (81.7%)	2 (90.8%)	1 (81.7%)	2 (90.8%)	2 (90.8%)
No. of C_d ASM vectors (energy)	1 (68.2%)	1 (68.2%)	2 (95.0%)	2 (95.0%)	3 (98.0%)

an early convergence in the derivation of ASM vectors. The most active vectors are not sensitive to the number of samples after the smoothing operation. This allows us to use a small number of samples to compute ASM vectors, which is highly beneficial to the efficiency of ASM-SBO.

In order to show the feasibility of the smoothing operation, we use ASM-SBO to solve the optimization problem in Table 1 with ASM vectors derived from both original samples and smoothed samples. Four sets of ASM vectors are investigated in this test, and they are from (i) 5 original samples, (ii) 200 original samples, (iii) 5 smoothed samples, and (iv) 200 smoothed samples.

Each set is composed of one C_l ASM vector and two C_d ASM vectors. The angle of attack is an independent variable in all cases, so there are four design variables in ASM-SBO. The EGO infilling strategy described in Section 2.1 is used, and the number of initial samples used to train surrogate models is 30.

Figure 10 shows the convergence histories and optimization results. Without the smoothing operation, the optimization with ASM vectors derived from 200 original samples is better than that with ASM vectors derived from 5 original samples. But both optimizations cannot effectively reduce the shock waves, so their optimized drag coefficients are both higher than 120 counts. Using more samples in ASM derivation might be helpful to the optimization, but the computational cost of deriving ASM vectors will become even higher than the optimization itself. So using original samples directly in ASM-SBO is not feasible. After the smoothing operation, we find that both optimizations converge to about 113 counts, and the shock waves have been reduced significantly. This means that the smoothing operation not only accelerates the convergence of the ASM computation but also makes ASM-SBO more effective. Furthermore, we conduct multiple tests to investigate the influence of DoE in active subspaces on optimization results. Figure 11 shows

Fig. 14 Accuracy comparison of surrogate models in different design spaces. **a** Error of C_l (1 C_l count equals 0.001). **b** Error of C_d (1 C_d count equals 0.0001)
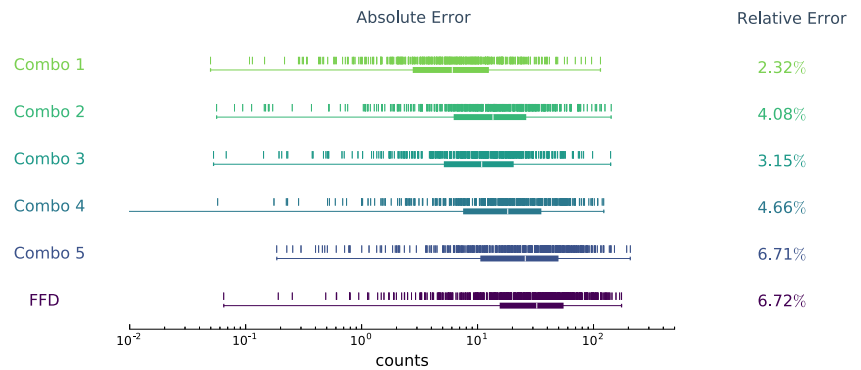
(a) Error of C_l (1 C_l count equals 0.001)
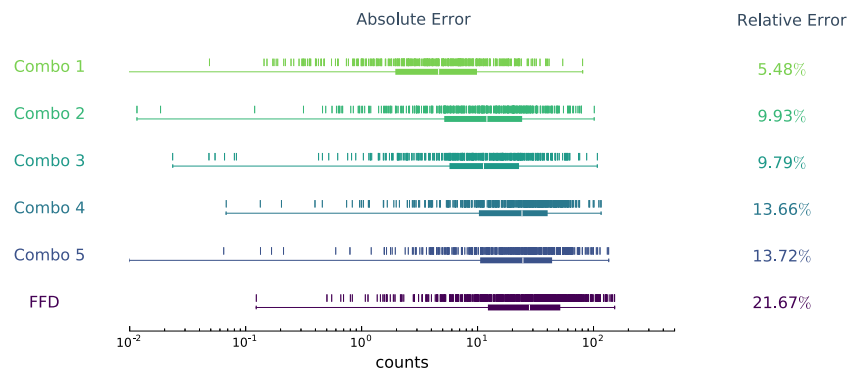
(b) Error of C_d (1 C_d count equals 0.0001)

Fig. 15 **a** Convergence histories and **b** optimization results with six surrogate models
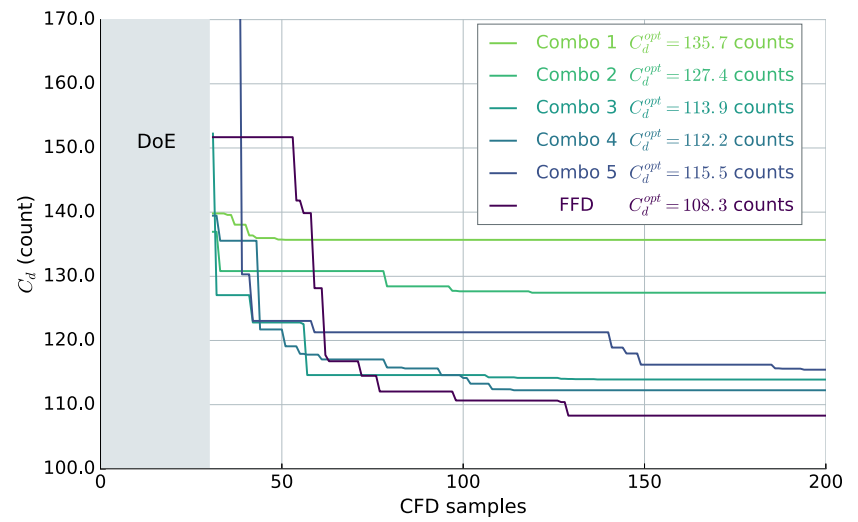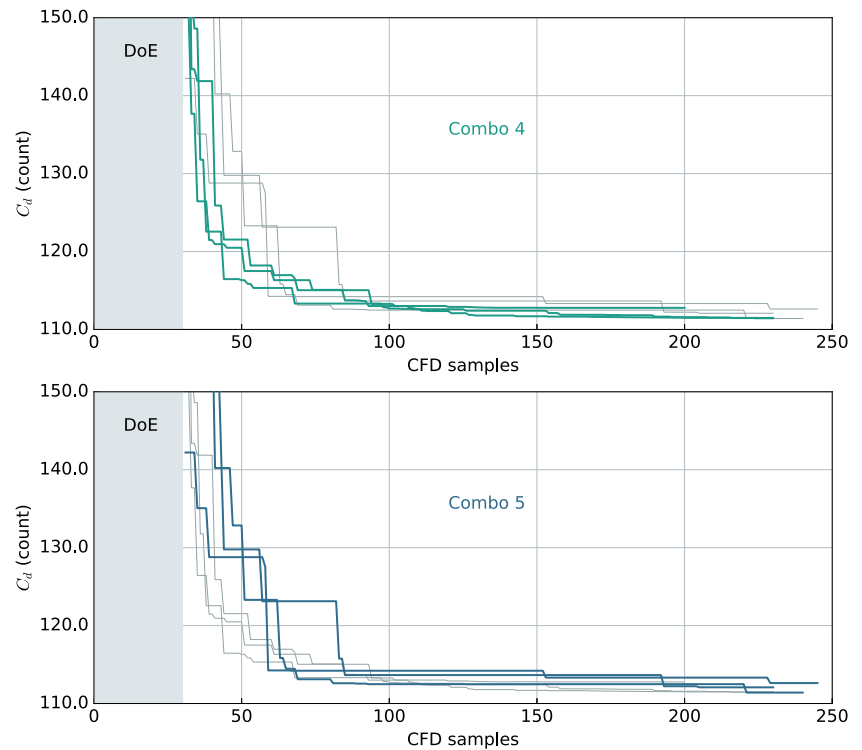

(a) Convergence histories
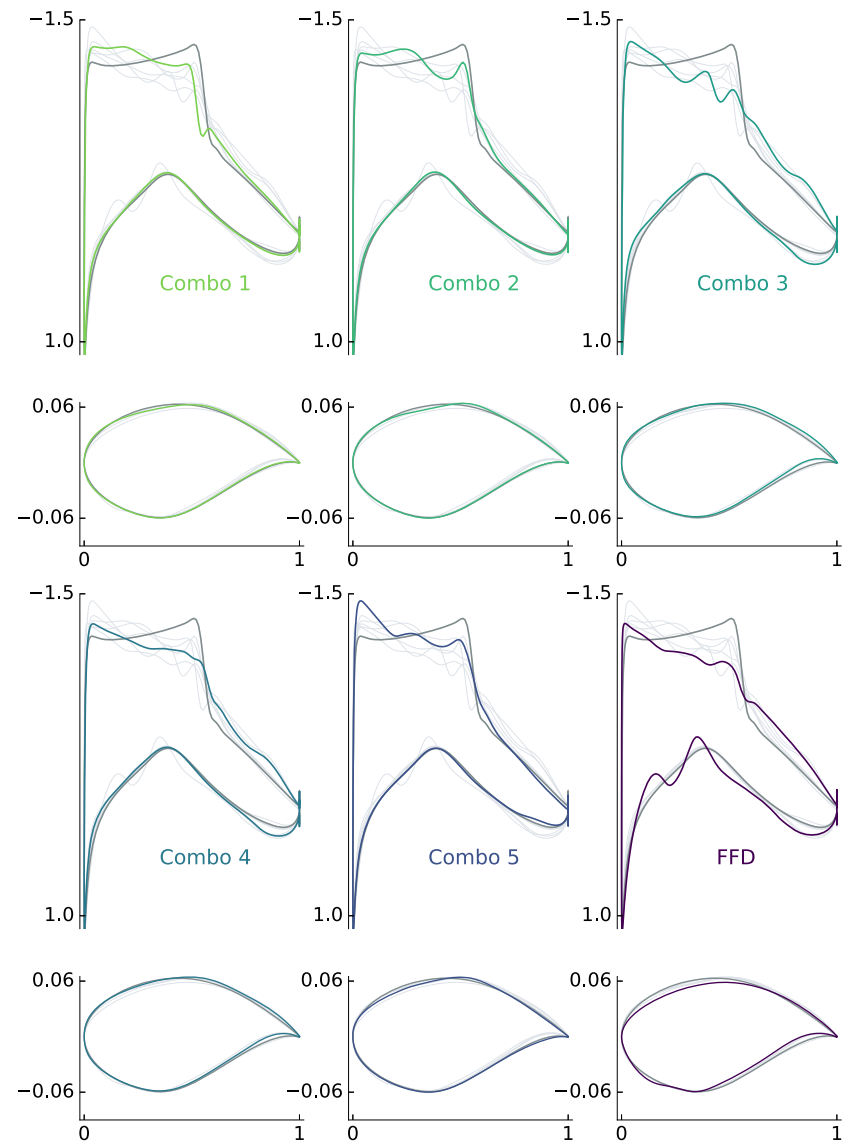
(b) Optimization results

Fig. 16 Multiple DoEs for ASM-SBO with Combo 4 and Combo 5

the optimization convergence histories with different DoEs. We can see that DoE does not affect the final result obviously, and this reinforces the robustness of the proposed ASM-SBO.
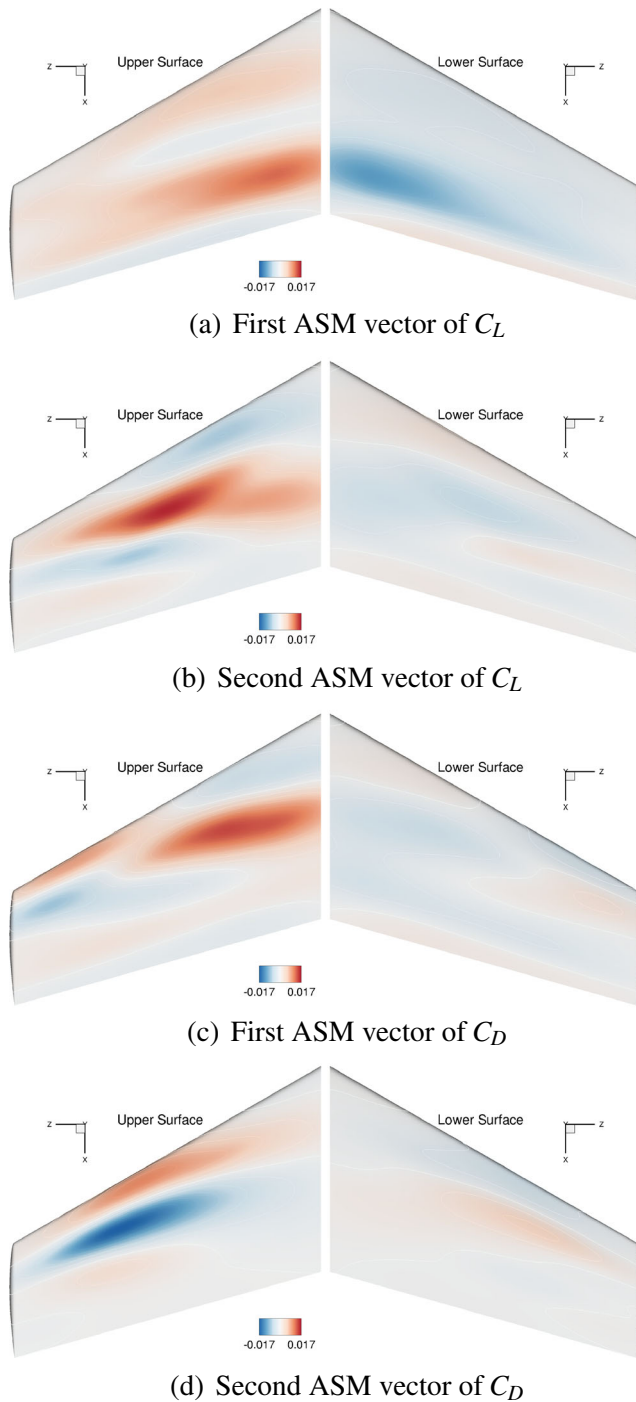
In Figs. 10 and 11, we also notice that the optimization with 5 smoothed samples is slightly better than that with 200 smoothed samples. This is due to the influence of initial samples used in the ASM derivation. We conduct multiple tests to investigate this phenomenon. We derive the four sets of ASM vectors from different initial samples, and then we conduct the corresponding ASM-SBO. In Fig. 12, we show the first ASM vectors of C_l and C_d on the left side and the corresponding optimization convergence histories on the right side. In Fig. 12a, b, the ASM vectors derived from different LHS samples without smoothing operations are obviously different, and this reinforces the fact that the ASM vectors are sensitive to the initial samples without smoothing operations. These optimization results vary a lot from one test to another, but no test is effective. However,

after the smoothing operations, the ASM vectors are much closer to each other in Fig. 12c, d. The smoothing operations make the ASM vectors less dependent on the initial samples. Although slight difference in ASM vectors results in slight difference in optimization results, the optimization after smoothing operations are all much more effective than the others. These features make ASM-SBO robust and efficient.

Smoothing operations reduce the demand on samples in the derivation of ASM vectors. The tests in Figs. 10 and 12 have shown that using 5 initial samples with the smoothing operations is effective to derive ASM vectors in the transonic airfoil shape optimization. In Section 4, we will show that using 10 smoothed samples is effective and efficient for the optimization of a transonic wing with 220 shape variables. Although a larger number of smoothed samples could provide a bit more consistent results as shown in Fig. 12d, a reasonable small number of (about 10) smoothed samples can be used in order to reduce the computational cost.

Table 3 Optimization problem statement in the wing case

	Functions	Quantity	Description
Minimize	C_D	1	Drag coefficient
with respect to	α	1	Angle of attack
	$\mathbf{x}_{shape}$	220	y perturbation of FFD control points
Subject to	$C_L \geq 0.26$	1	Lift constraint

(a) First ASM vector of C_L

(b) Second ASM vector of C_L

(c) First ASM vector of C_D

(d) Second ASM vector of C_D

Fig. 17 M6 wing shape deformations corresponding to ASM vectors. The color counter indicates the surface normal displacement magnitude. **a** First ASM vector of C_L. **b** Second ASM vector of C_L. **c** First ASM vector of C_D. **d** Second ASM vector of C_D

3.2 Number of ASM vectors in ASM-SBO

In order to figure out how large the active subspace should be, we further study the influence of the number of ASM vectors on the results of ASM-SBO. Ten smoothed LHS

samples are used to derive ASM vectors for both C_l and C_d. The normalized eigenvalues are shown in Fig. 13. We can see that the first several ASM vectors take the majority of the total energy in eigenvalues.

Five combinations of C_l and C_d ASM vectors are selected and listed in Table 2. The energy of ASM vectors used in each combination is listed as well, which is computed by the relative magnitude of eigenvalues. For example, Combo 1 uses the first ASM vectors of C_l and C_d, which take 81.9% and 68.2% energies of the two functions, respectively.

We first investigate the accuracy of surrogate models in the five active subspaces and the original FFD design space. Five groups of DoE samples in each design space are generated by LHS and evaluated by the RANS CFD solver, and each DoE has 30 samples. We use the leave-one-out strategy to train the Kriging model with samples in one DoE and validate the model with samples in other DoEs. In Fig. 14, we provide the absolute error with scatter plots and box plots to compare the accuracy of surrogate models in different design spaces, and we also provide the relative error computed by (12) to make a clear comparison. It can be seen that the accuracy of the surrogate model in each active subspace is higher than that in the original FFD design space. This implies that conducting optimization in the active subspace might be more efficient. It can be noticed that the accuracy of the surrogate in the active subspace decreases with the increase of the number of ASM vectors, and this would affect the performance of ASM-SBO, which will be discussed later.

$$\text{Relative error} = \frac{\|\mathbf{f}_{\text{Kriging}} - \mathbf{f}_{\text{CFD}}\|_2}{\|\mathbf{f}_{\text{CFD}}\|_2} \times 100\%. \qquad (12)$$

Then, we conduct five ASM-SBOs with them, and we also conduct a SBO using original FFD variables for comparison. Figure 15 shows the convergence histories and results of six optimizations. From Fig. 15a, we find that ASM-SBOs converge faster than SBO, except Combo 5, and using an insufficient number of ASM vectors (Combo 1 and Combo 2) is ineffective although the convergence is fast. However, it does not mean that using more ASM vectors is always better, because the efficiency of ASM-SBO would suffer from the less accurate surrogate model with the increase of ASM vectors, such as Combo 5. To reduce the influence of the particular DoE on the results, we conduct multiple tests with different training samples to compare the optimization performance with Combo 4 and Combo 5, which are shown in Fig. 16. The best result ($C_d = 111.24$ counts) is obtained in one test with Combo 5, but the largest iteration numbers are also used by tests with Combo 5. In addition, the benefit from one more ASM vector is not obvious (about 0.2 C_d count). It can be concluded that ASM-SBO with Combo 5 could find a
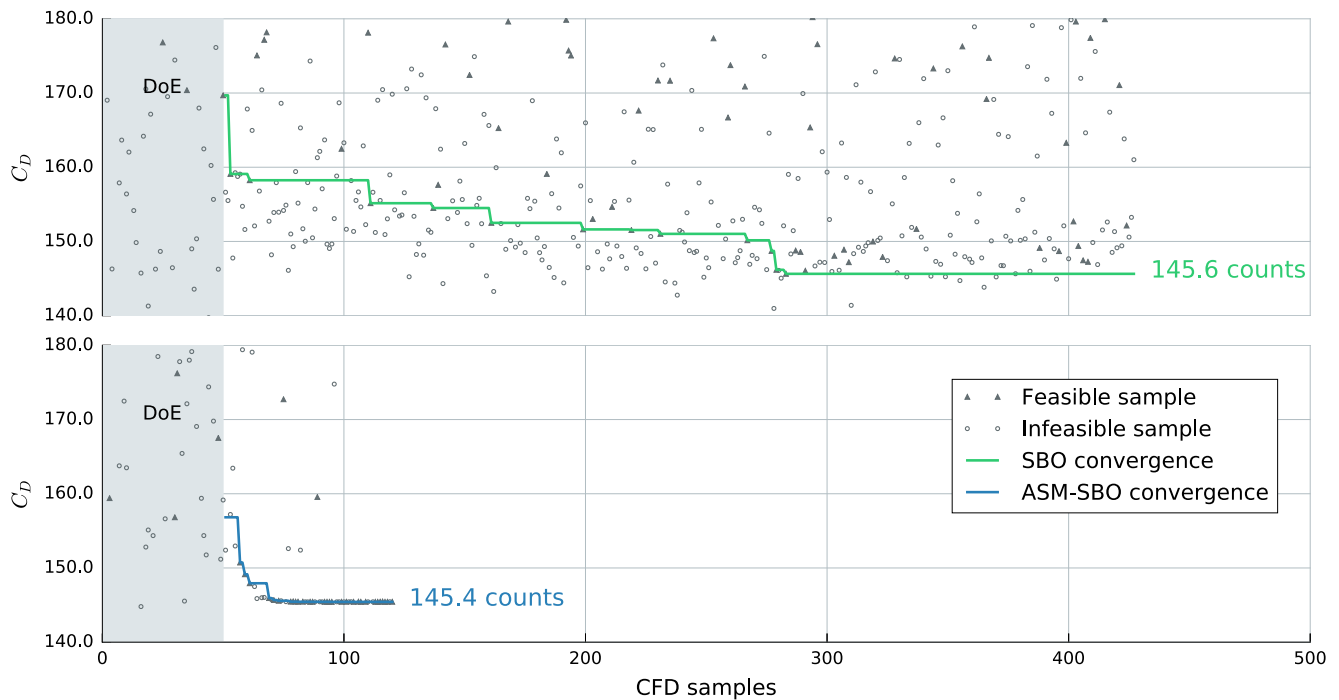
Fig. 18 Optimization convergence histories of SBO (upper) and ASM-SBO (lower) starting with 50 initial samples

slight better result but it requires more iterations. If we only allow a computational budget with 200 CFD evaluations, the average value of optimized C_d with Combo 4 in these tests is 111.7 counts, while the mean with Combo 5 is 112.1 counts. So using one more ASM vector, like Combo 5, is not always beneficial to the optimization. One combo that compromises the benefit from a larger design space with more ASM vectors and the drawback of corresponding lower accuracy in the surrogate models is desired.

A suitable number of ASM vectors like Combo 3 and Combo 4 make ASM-SBO effective and more efficient than regular SBO. Based on the comparison these optimizations, we recommend using a number of ASM vectors that can cover 90~95% energy. In Section 4, we conduct ASM-SBO with this criterion and obtain an efficient result.

In this case, SBO finds the best optimum because SBO works well with this number of design variables (19 in total). Nevertheless, it is still promising to use ASM-SBO. For example, ASM-SBOs with Combo 3 and Combo 4 converge to comparable results with lower computational cost, and the optimized shapes are smoother than that optimized by SBO, especially in the lower surface. This advantage will be more obvious in high-dimensional problems like a wing shape optimization in Section 4.
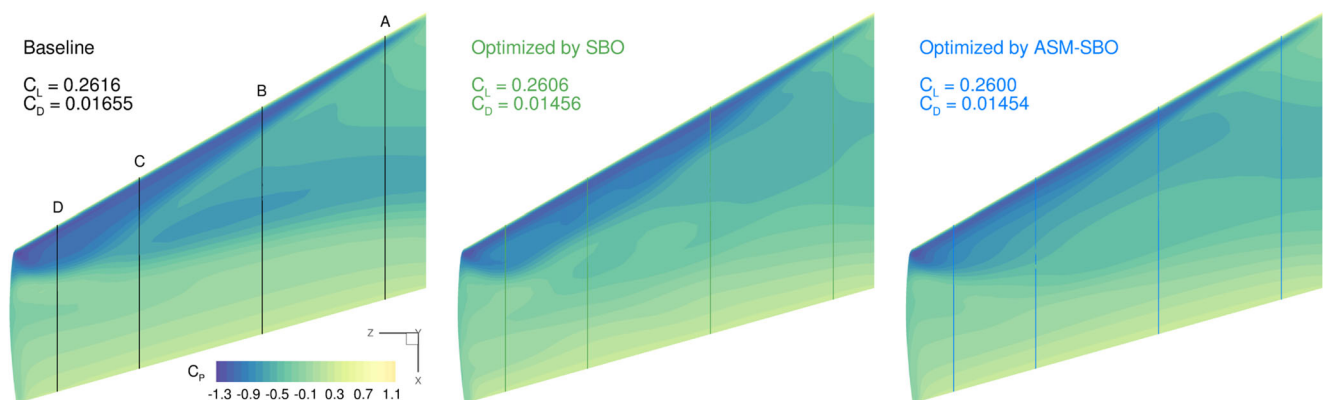
Fig. 19 Contour comparison of SBO and ASM-SBO
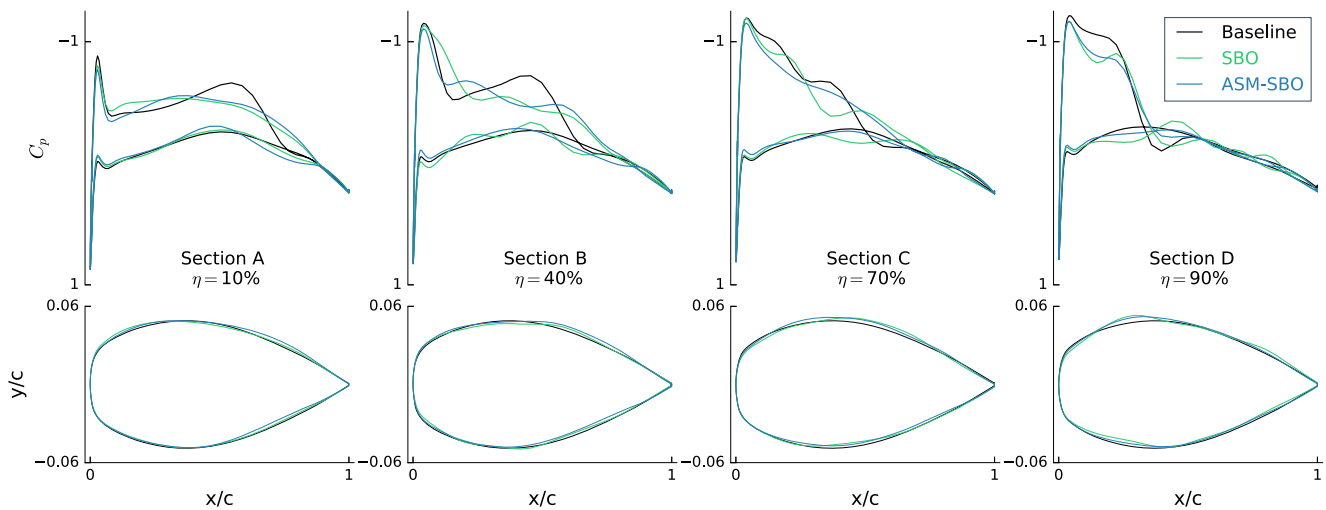
Fig. 20 Section airfoil and C_P comparison of SBO and ASM-SBO

4 Transonic wing shape optimization

We minimize the drag of the ONERA M6 wing by manipulating its shape using ASM-SBO and regular SBO in this case. The Mach number is 0.8395, and Reynolds number is 11.72×10^6. At the initial angle of attack ($\alpha = 3.06°$), the wing is associated with complex transonic flow phenomena, e.g., local supersonic flow and shock waves. So there is a space for optimizers to improve its performance. We use a CFD mesh with 1.5 million cells in the simulation, and the height of the first mesh layer is small enough to ensure $y^+ < 1.0$. All the computations are conducted in a server with 20 Xeon cores.

We use a $10 \times 2 \times 11$ FFD box to parameterize the wing shape. Only y coordinates of the FFD points can move during the optimization, which results in 220 shape variables. We add a constraint for C_L, so we consider the angle of attack as a design variable as well. The optimization statement is listed in Table 3.

We generate 10 samples in the 221 dimensional space and smooth their shapes using the algorithm in Fig. 6 before the CFD simulation and gradient computation. Then, we derive ASM vectors of C_L and C_D from the gradient of these samples. Based on the criterion proposed in Section 3.2, we use two ASM vectors of C_L and two ASM vectors of C_D in this case, which take 95.1% and 94.4% energies, respectively. Figure 17 shows their effects on deforming the wing surface. The first ASM vector of C_L (Fig. 17a) has an obvious influence on both the upper and lower surfaces. Basically, it tends to make the wing, especially near the root, more flattened on the upper surface and more cambered in the aft section. This deformation helps to generate more lift. In this flow condition, the upper surface is associated with shock waves, which increase

the drag severely. Manipulating the upper surface can promisingly decrease the drag. So we find in Fig. 17c, d that most manipulations by C_D ASM vectors are on the upper surface.

We conduct both ASM-SBO and SBO with 50 initial training samples in corresponding design spaces. Figure 18 shows the infilling history of samples along the optimization convergence. The upper and lower figures are SBO and ASM-SBO, respectively. We can find a much higher efficiency in ASM-SBO, which uses 78 CFD simulations (including training samples) to converge to $C_D = 145.4$ counts. SBO takes 283 CFD simulations and obtains a C_D of 145.6 counts.

In each iteration, ASM is still costlier than ASM-SBO. First, training surrogate models in ASM-SBO is much easier due to its lower dimension. In this case, it always takes less than one second to train C_L and C_D surrogates in ASM-SBO, while about ten minutes are used in the end of SBO. Moreover, the average CFD simulation time per iteration in SBO is about one hour, which is higher than that (50 minutes) in ASM-SBO. The reason is that the shapes of infilling samples in SBO are more likely to have oscillations due to the inaccuracy of surrogates, whereas, in ASM-SBO, benefiting from the lower dimension, surrogates are more accurate and shapes of infilling samples during the optimization are more reasonable. So, CFD simulation is faster in ASM-SBO.

In Fig. 18, we can find that most of infilling samples at the end of ASM-SBO are along the convergence line. This demonstrates that the surrogates in ASM-SBO are accurate and implies a clear convergence of the optimization. Whereas, the infilling samples are still "randomly" distributed at the end of SBO due to inaccurate surrogates. If we continue the optimization in SBO, the

surrogates might become more accurate and the drag might be further reduced, but it requires a much larger computation budget. Since both ASM-SBO and SBO are suitable to and more likely used in preliminary design or rough optimization, it is vital to obtain a reasonable design with a computational budget as small as possible. They could further be coupled with a detailed optimization, for example, in a two-step optimization strategy (Duan et al. 2012; Li et al. 2018b).

Including the cost of the ASM derivation, the total computational cost of ASM-SBO is only 30% of that in SBO. The optimization results are compared in Figs. 19 and 20. From the contour comparison in Fig. 19, we find reductions of shock waves in both results, which account for their drag reductions. Although they have similar drag reductions, the optimized shapes are of a difference which can be seen in Fig. 20. The optimized shape by ASM-SBO is smoother than the optimized shape by SBO, which leads to smoother C_P distributions. This feature is beneficial for manufacture, flight control, and further detailed optimization. Thus, from the view of practice, the efficient and effective convergence of ASM-SBO is remarkably helpful to high-dimensional aerodynamic shape design optimization problems.

5 Conclusions

We present an efficient approach to applying ASM in high-dimensional aerodynamic shape optimization problems. Benefiting from the smoothing operation, only a sparse sampling of the design space is required to derive ASM vectors, which significantly reduces the computational cost. To make the optimization efficient and effective, a suitable number of ASM vectors should be used. This number can be determined by the 90~95% energy coverage criterion. In a transonic wing optimization case with 220 shape design variables, we derive ASM vectors from 10 smoothed samples and use 2 C_L ASM vectors plus 2 C_D ASM vectors in ASM-SBO. Compared with regular SBO, ASM-SBO converges to a better design using only 30% computational cost.

ASM-SBO proposed in this work effectively addresses the issue of SBO in high-dimensional aerodynamic shape optimization. It would be promising to use this method in other high-dimensional computational engineering problems like structural and multidisciplinary optimization for preliminary design.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

Andrés E, Salcedo-Sanz S, Monge F, Pérez-Bellido A (2012) Efficient aerodynamic design through evolutionary programming and support vector regression algorithms. Expert Syst Appl 39(12):10700–10708. https://doi.org/10.1016/j.eswa.2012.02.197

Bons N, He X, Mader CA, Martins JRRA (2017) Multimodality in aerodynamic wing design optimization. In: 35th AIAA Applied Aerodynamics Conference, American Institute of Aeronautics and Astronautics. https://doi.org/10.2514/6.2017-3753

Chernukhin O, Zingg DW (2013) Multimodality and global optimization in aerodynamic design. AIAA J 51(6):1342–1354. https://doi.org/10.2514/1.j051835

Constantine PG, Dow E, Wang Q (2014) Active subspace methods in theory and practice: applications to Kriging surfaces. SIAM J Sci Comput 36(4):A1500–A1524. https://doi.org/10.1137/130916138

Duan Y, Cai J, Li Y (2012) Gappy proper orthogonal decomposition-based two-step optimization for airfoil design. AIAA J 50(4):968–971. https://doi.org/10.2514/1.j050997

Eberhart R, Kennedy J (1995) A new optimizer using particle swarm theory. In: Proceedings of the Sixth International Symposium on Micro Machine and Human Science. IEEE. https://doi.org/10.1109/mhs.1995.494215

Han ZH, Zhang KS (2012) Surrogate-based optimization. In: Real-World Applications of Genetic Algorithms. InTech. https://doi.org/10.5772/36125

Han ZH, Abu-Zurayk M, Görtz S, Ilic C (2018) Surrogate-based aerodynamic shape optimization of a wing-body transport aircraft configuration. In: Notes on Numerical Fluid Mechanics and Multidisciplinary Design, Springer International Publishing, pp 257–282. https://doi.org/10.1007/978-3-319-72020-3_16

Jameson A (1988) Aerodynamic design via control theory. J Sci Comput 3(3):233–260. https://doi.org/10.1007/bf01061285

Jones DR, Schonlau M, Welch WJ (1998) Efficient global optimization of expensive black-box functions. J Global Optim 13(4):455–492. https://doi.org/10.1023/A:1008306431147

Krige DG (1951) A statistical approach to some basic mine valuation problems on the Witwatersrand. J Chem Metallurgical Mining Soc 52:119–139

Leifsson L, Koziel S (2010) Multi-fidelity design optimization of transonic airfoils using physics-based surrogate modeling and shape-preserving response prediction. J Comput Sci 1(2):98–106. https://doi.org/10.1016/j.jocs.2010.03.007

Li J, Bouhlel MA, Martins JRRA (2018a) Data-based approach for fast airfoil analysis and optimization. AIAA Journal (In press)

Li J, Cai J, Qu K (2018b) Adjoint-based two-step optimization method using proper orthogonal decomposition and domain decomposition. AIAA J:1–13. https://doi.org/10.2514/1.j055773

Liu J, Song WP, Han ZH, Zhang Y (2016) Efficient aerodynamic shape optimization of transonic wings using a parallel infilling strategy and surrogate models. Struct Multidiscip Optim 55(3):925–943. https://doi.org/10.1007/s00158-016-1546-7

Lukaczyk TW, Constantine P, Palacios F, Alonso JJ (2014) Active subspaces for shape optimization. In: 10th AIAA Multidisciplinary Design Optimization Conference, American Institute of Aeronautics and Astronautics. https://doi.org/10.2514/6.2014-1171

Lyu Z, Kenway GK, Paige C, Martins JRRA (2013) Automatic differentiation adjoint of the Reynolds-averaged Navier-Stokes equations with a turbulence model. In: 21st AIAA Computational Fluid Dynamics Conference, American Institute of Aeronautics and Astronautics. https://doi.org/10.2514/6.2013-2581

Mader CA, Martins JRRA, Alonso JJ, van der Weide E (2008) ADJoint: an approach for the rapid development of discrete adjoint solvers. AIAA J 46(4):863–873. https://doi.org/10.2514/1.29123

McKay MD, Beckman RJ, Conover WJ (1979) A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics 21(2):239. https://doi.org/10.2307/1268522

Namura N, Shimoyama K, Obayashi S (2017) Kriging surrogate model with coordinate transformation based on likelihood and gradient. J Glob Optim 68(4):827–849. https://doi.org/10.1007/s10898-017-0516-y

Othmer C, Lukaczyk TW, Constantine P, Alonso JJ (2016) On active subspaces in car aerodynamics. In: 17th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, American Institute of Aeronautics and Astronautics. https://doi.org/10.2514/6.2016-4294

Parr JM, Keane AJ, Forrester AI, Holden CM (2012) Infill sampling criteria for surrogate-based optimization with constraint handling. Eng Optim 44(10):1147–1166. https://doi.org/10.1080/0305215x.2011.637556

Poole DJ, Allen CB, Rendall T (2017) Global optimization of multi-modal aerodynamic optimization benchmark case. In: 35th AIAA Applied Aerodynamics Conference, American Institute of Aeronautics and Astronautics. https://doi.org/10.2514/6.2017-4365

Queipo NV, Haftka RT, Shyy W, Goel T, Vaidyanathan R, Tucker PK (2005) Surrogate-based analysis and optimization. Prog Aerosp Sci 41(1):1–28. https://doi.org/10.1016/j.paerosci.2005.02.001

Streuber GM, Zingg DW (2017) Investigation of multimodality in aerodynamic shape optimization based on the Reynolds averaged Navier-Stokes equations. In: 35th AIAA Applied Aerodynamics Conference, American Institute of Aeronautics and Astronautics. https://doi.org/10.2514/6.2017-3752

Wu X, Zhang W, Song S (2017) Robust aerodynamic shape design based on an adaptive stochastic optimization framework. Struct Multidiscip Optim 57(2):639–651. https://doi.org/10.1007/s00158-017-1766-5