Blade Envelopes Part II: Multiple Objectives and Inverse Design

Chun Yui Wong[†,*], **Pranay Seshadri**[‡,*], **Ashley Scillitoe**[*], **Bryn Noel Ubald**[*]
Andrew B. Duncan[‡,*], **Geoffrey Parks**[†]
[†]Department of Engineering, University of Cambridge, U.K.
[‡]Department of Mathematics, Imperial College London, U.K.
[*]Data-Centric Engineering, The Alan Turing Institute, U.K.

Blade envelopes offer a set of data-driven tolerance guidelines for manufactured components based on aerodynamic analysis. In Part I of this two-part paper, a workflow for the formulation of blade envelopes is described and demonstrated. In Part II, this workflow is extended to accommodate multiple objectives. This allows engineers to prescribe manufacturing guidelines that take into account multiple performance criteria.

The quality of a manufactured blade can be correlated with features derived from the distribution of primal flow quantities over the surface. We show that these distributions can be accounted for in the blade envelope using vector-valued models derived from discrete surface flow measurements. Our methods result in a set of variables that allows flexible and independent control over multiple flow characteristics and performance metrics, similar in spirit to inverse design methods. The augmentations to the blade envelope workflow presented in this paper are demonstrated on the LS89 turbine blade, focusing on the control of loss, mass flow and the isentropic Mach number distribution. Finally, we demonstrate how blade envelopes can be used to visualize invariant designs by producing a 3D render of the envelope using 3D modelling software.

1 INTRODUCTION

In the first part of this two-part paper [1], we defined the concept of a *blade envelope*, a visual and computational guideline yielding automatic scrap-or-use decisions for manufactured turbomachinery components. Using the theory of inactive subspaces, a range of geometric designs that are invariant in loss is identified, and geometries from this invariant region can be generated with no additional computational fluid dynamics (CFD) solves. From this, the decision to scrap or keep a measured component reduces down to the computation of the Mahalanobis distance from an aerodynamic knowledge base consisting of invariant designs.

In this second part, we extend blade envelopes beyond the manufacturing stage of production, and describe how they can be used during the design stage as well. During the shape design of a highly-loaded turbine stage, the minimization of loss is often accompanied with constraints to avoid trivial solutions where the blade is unloaded. For example, in [2], the exit flow angle is constrained to be above the baseline value to ensure sufficient work extraction. In [3], the authors put an equality constraint on the mass flow rate while optimizing the loss coefficient to factor out possible reduction in entropy generation due to reduction in flow capacity. Prior work [4, 5] has leveraged active subspaces to construct 2D performance maps for compressor blade design. In the latter work, multiple objectives including the pressure ratio and flow capacity are considered by mapping contours of different objectives onto the active subspace of efficiency. Manufacturing deviations are modeled as constant excursions from the nominal design. The main drawback of this approach is the requirement to run further simulations to map out performance contours in the active subspace. In this work, we incorporate multiple aerodynamic design requirements by interpreting them as additional constraints factored into blade envelopes.

In situations where tighter control over the performance of the component is required, constraints on surface flow characteristics can be implemented. Clark [6] establishes the correlation between aerodynamic features—defined via parts of the surface isentropic Mach number distribution—and aerodynamic performance. Control over these key features can be achieved by factoring the isentropic Mach number distribution as an additional *vector-valued* objective in blade envelopes. This approach is similar in spirit to *inverse design*, where a target distribution is specified on the surface of a blade, and the blade shape is iteratively modified to give a geometry that matches the distribution. While inverse design yields an optimal geometry that fits the design criterion over the entire surface, our approach aims to find designs that satisfy the target distribution in parts of the flow that are most critical to performance. The relaxation of constraints on other locations allows a range of designs to be specified, whose expanse is explicitly quantified by the blade envelope. Moreover, we can combine the control over the surface flow profile with constraints over other scalar objectives to perform inverse design constrained by requirements on other measures of performance.

In addition to computationally quantifying the region of a high-dimensional design space corresponding to designs with invariant objective values, blade envelopes are also amenable to intuitive visualizations. We conclude this work by connecting the numerical output of the blade envelopes workflow to 3D mod-

---

Fig. 1. A CAD representation of the controlled loss and mass flow rate blade envelope.

elling software, producing outputs such as Figure 1. The result is a visualization of the possible designs within the blade envelope that can be easily shared between engineers and made interactive via novel technologies such as 3D printing and augmented reality.

2 COMPUTATIONAL METHODOLOGIES

Blade envelopes demarcate boundaries between regions within the space of manufactured geometries which exhibit similar performance metrics. These envelopes are formed from statistics derived from an aerodynamic database containing geometries sampled from the inactive subspace with respect to a scalar objective. Building upon this framework, we describe two algorithms to identify regions of the design space characterized by invariance in *multiple objectives*, including both scalar- and vector-valued objectives. This enables us to obtain samples invariant in these objectives to form the required database for a multi-objective blade envelope.

Recall from [1] the partitioning of the design space into the *active* and *inactive* subspaces via an orthogonal matrix $Q = [W \ V]$,

$$\mathbf{x} = WW^T\mathbf{x} + VV^T\mathbf{x}, \qquad (1)$$

where we desire that the quantity of interest varies predominantly with changes in the active coordinates $W^T\mathbf{x}$, and are approximately invariant with respect to changes in the inactive coordinates $V^T\mathbf{x}$. Building on this idea of subspace decomposition, we aim to identify an inactive subspace matrix V whose columns form a basis for a subspace within which multiple objectives are approximately invariant.

2.1 Intersection of inactive subspaces

Consider two independent objectives $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ depending on the design variables $\mathbf{x}$ differently. Assuming each objective is differentiable, we can compute the following symmetric positive semi-definite matrices,

$$\boldsymbol{C}_i = \int_{\mathcal{X}} \nabla_{\mathbf{x}} f_i(\mathbf{x}) \nabla_{\mathbf{x}} f_i(\mathbf{x})^T \ \tau \ d\mathbf{x}, \qquad i = 1, 2 \qquad (2)$$

where $\tau = 2^{-d}$ defines a uniform distribution over the design space, which is a d-dimensional hypercube $[-1,1]^d$ as before. Their respective eigenvalue decompositions can be evaluated,

$$\boldsymbol{C}_i = [\boldsymbol{W}_i \ \boldsymbol{V}_i] \begin{bmatrix} \boldsymbol{\Lambda}_{i,1} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Lambda}_{i,2} \end{bmatrix} \begin{bmatrix} \boldsymbol{W}_i^T \\ \boldsymbol{V}_i^T \end{bmatrix}, \qquad i = 1, 2, \qquad (3)$$

where the non-negative eigenvalues are ordered in descending order, and the diagonal terms in $\Lambda_{i,2}$ are sufficiently close to zero, for $i = 1,2$. From (3), the inactive subspace corresponding to $f_i(\mathbf{x})$, i.e., the column span of $\boldsymbol{V}_i$, can be obtained. Let's denote this by $\mathcal{V}_i = \mathrm{colspan}(\boldsymbol{V}_i)$. The idea is to find the *intersection* of the inactive subspaces $\mathcal{V}_{int} = \mathcal{V}_1 \cap \mathcal{V}_2$. If a design perturbation lies within $\mathcal{V}_{int}$, then by definition it lies in both $\mathcal{V}_1$ and $\mathcal{V}_2$, implying that such a perturbation has insignificant effects on both $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$. How do we find this intersection, and how can we sample from it?

It can be shown that $\mathcal{V}_{int}$ is a linear subspace that can be expressed as the column span of a basis matrix (see [7, p. 64]). Let this matrix be $\boldsymbol{V}_{int}$. Once we can evaluate $\boldsymbol{V}_{int}$, the exploration of the subspace $\mathcal{V}_{int}$ subject to constraints reduces to an application of the hit-and-run sampling algorithm as described in Part I [1]. To find this matrix, we require the *active* subspace basis matrices $\boldsymbol{W}_1 \in \mathbb{R}^{d \times r_1}$ and $\boldsymbol{W}_2 \in \mathbb{R}^{d \times r_2}$, and the assumption that the dimension r of the space $\mathrm{colspan}(\boldsymbol{W}_1) \oplus \mathrm{colspan}(\boldsymbol{W}_2)$ is less than d, the ambient design space dimension, where $\oplus$ denotes the direct sum. With these assumptions, the following recipe can be used to find $\boldsymbol{V}_{int}$:

1. Form $\boldsymbol{W}_{both} := [\boldsymbol{W}_1 \ \boldsymbol{W}_2]$.
2. Decompose $\boldsymbol{W}_{both}$ by QR-factorization, $\boldsymbol{W}_{both} = \boldsymbol{QR}$.
3. Set $\boldsymbol{V}_{int}$ to be the last $d - r$ columns of $\boldsymbol{Q}$.

This gives us the basis matrix $\boldsymbol{V}_{int}$ corresponding to the inactive subspace for both $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$. This procedure is straightforwardly generalizable to $n > 2$ objectives via induction, as long as the dimension r of the direct sum of the active subspaces is strictly less than d.

2.2 Vector-valued objectives

Given the method of intersections, a natural extension to find an invariant subspace for a vector-valued objective would be to treat the N components of the objective

$$\begin{aligned} \mathbf{M}(\mathbf{x}) &= [M_1(\mathbf{x}), M_2(\mathbf{x}), ..., M_N(\mathbf{x})] \\ &= [M_1, M_2, ..., M_N] \end{aligned} \qquad (4)$$

2

as independent scalar objectives, and find the intersection of all their respective inactive subspaces $\mathcal{V}_{int} = \mathcal{V}_1 \cap \mathcal{V}_2 \cap ... \cap \mathcal{V}_N$. The challenge with this approach for large N (when compared to d) is the fact that the total number of active coordinates is likely to exceed d. In this situation, the intersection of the inactive subspaces is trivial, i.e., it contains only the zero vector. An alternative approach that provides approximate invariance across the component objectives is required.

We take cues from the work of Wong et al. [8] and Zahm et al. [9] and define the following multivariate generalisation of (2),

$$H = \int_{\mathcal{X}} J(\mathbf{x}) \, R \, J(\mathbf{x})^T \, \tau \, d\mathbf{x}, \tag{5}$$

where

$$J(\mathbf{x}) = \left[\frac{\partial M_1}{\partial \mathbf{x}}, ... , \frac{\partial M_N}{\partial \mathbf{x}} \right] \in \mathbb{R}^{d \times N} \tag{6}$$

is the Jacobian matrix. The matrix R is a user-defined $N \times N$ diagonal *weight matrix*,

$$R = \begin{bmatrix} \omega_1 & & & \\ & \omega_2 & & \\ & & \ddots & \\ & & & \omega_N \end{bmatrix}, \tag{7}$$

where $\omega_i \geq 0$ for $1 \leq i \leq N$. The corresponding active and inactive subspace basis matrices W_M and V_M for M can be found again via the eigenvalue decomposition,

$$H = [W_M \ V_M] \begin{bmatrix} \Lambda_{M,1} & 0 \\ 0 & \Lambda_{M,2} \end{bmatrix} \begin{bmatrix} W_M^T \\ V_M^T \end{bmatrix}, \tag{8}$$

with $W_M \in \mathbb{R}^{d \times r}$ and $V_M \in \mathbb{R}^{d \times (d-r)}$. The dimensionality r is found by identifying the largest eigenvalue gap, as in the scalar case. In [9], the focus is placed on finding the active subspace for vector-valued objectives to achieve dimension reduction. Here, we place the emphasis on the inactive subspace instead, to achieve component-wise invariance in vector-valued quantities.

The power of this method lies in the flexibility in setting the weight matrix R. It can be shown that H is the weighted sum of the gradient covariance matrices corresponding to each component objective [9, Sec. 5.2.2]. That is, if

$$C_i = \int_{\mathcal{X}} \nabla_{\mathbf{x}} M_i(\mathbf{x}) \nabla_{\mathbf{x}} M_i(\mathbf{x})^T \, \tau \, d\mathbf{x}, \tag{9}$$

then

$$H = \sum_{i=1}^{N} \omega_i C_i. \tag{10}$$

The larger the weight for ω_i, the bigger its influence on the eigenvectors W_M and V_M. Samples from the inactive subspace yield output values that are close to nominal at locations with a large prescribed weight; in locations with a smaller weight, output values are allowed to vary more freely. The weights allow us to prescribe different degrees of invariance to various locations indexed by the vector-valued objective.

2.3 Discussion: Distinguishing between inactive subspaces

The problem of finding an orthonormal basis for the intersection of multiple inactive subspaces is equivalent to finding the null space of their direct sum (see 6.4.2 in [7]). In other words, there is a connection between the inactive subspace of H from (10), and last few columns of Q in Section 2.1.

Consider the inactive subspace to be the orthogonal complement of the column span of H, which would correspond to the eigenvectors of H that have eigenvalues of zero. In practice, however, if we have many independent objectives, there is a strong possibility that the span of H will have full dimension, in which case the inactive subspace will be trivial, i.e., $\{0\}$. For practical reasons, this motivates us to consider a relaxed definition of the inactive subspace, which corresponds to the linear span of *small* eigenvalues of H. The threshold between *small* and *not-small* is problem specific, although one can make various scaling arguments about what a reasonable threshold would be. In this setting, the inactive subspace will depend strongly on the choice of the scaling parameters R, which can weight the relative importance of difference subspaces.

Utilizing the direct intersection approach in Section 2.1 implies that the inactive subspaces of the various objectives are characterized by eigenvalues that are all exactly zero (albeit with a different formulation of the active subspace). We can interpret this as a *strongly* inactive subspace, and it will be used when considering multiple flow quantities. In contrast, the inactive subspace obtained through H represents a *weakly* inactive subspace, and is used in this paper for studying a single vector-valued quantity, e.g., isentropic Mach number distribution.

2.4 Constrained tuning via active coordinates

Assuming that the active subspaces are linearly independent—a mild assumption for independent objectives, using the method of intersection of n scalar- or vector-valued objectives, we obtain a linear decomposition of the design space. This decomposition can be characterised by gathering the active subspace basis matrices together with the intersection of the inactive subspaces,

$$D = \begin{bmatrix} W_1 & W_2 & ... & W_n & V \end{bmatrix}, \tag{11}$$

yielding a square matrix $D \in \mathbb{R}^{d \times d}$ with independent columns. Corresponding to D, a coordinate transformation on the design $\mathbf{x}$ yields the following

$$D^T \mathbf{x} = \left[(W_1^T \mathbf{x})^T \ (W_2^T \mathbf{x})^T \ ... \ (W_n^T \mathbf{x})^T \ (V^T \mathbf{x})^T \right]^T$$
$$= [\underbrace{\mathbf{u}_1^T \ \mathbf{u}_2^T \ ... \ \mathbf{u}_n^T}_{r_1 + r_2 + ... + r_n} \ \mathbf{z}^T]^T \tag{12}$$

3

The first components of the new coordinates $\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_n$ correspond to the active coordinates of the objectives $f_1, f_2, ..., f_n$; the final coordinates $\mathbf{z}$ correspond to the inactive coordinates. Any perturbation lying in the column space of $\mathbf{V}$ has its first $r_1+r_2+...+r_n$ coordinates equal to zero, implying that it causes no change to the active coordinates for all objectives. Thus, this perturbation does not cause significant change to any objective.

Assuming that each coordinate is independent, the active coordinates can also be treated as tunable parameters that directly control each objective separately. Adjusting the values of $\mathbf{u}_1$ while keeping every other active coordinate constant results in changes in f_1, while maintaining the values of other objectives. Note that we can allow the inactive coordinates $\mathbf{z}$ to float, since those coordinates do not affect any objective.
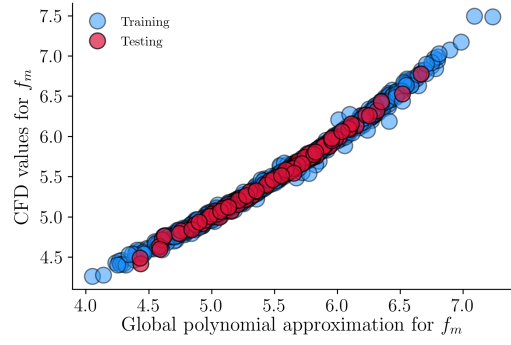

Fig. 2. Validation of polynomial model for the mass flow function. This coincides with the sufficient summary plot because the global polynomial approximation is a linear model.

3 AUGMENTING THE BLADE ENVELOPE FOR LS89

In the following, we apply the methods described above to extend the blade envelope for the LS89 turbine profile to accommodate a constraint on the mass flow function. Next, the method of conditional tuning is demonstrated by identifying geometries with constant loss and mass flow coefficients by their isentropic Mach number distributions.

3.1 Inactive subspace for loss and mass flow

Recall from Equation (2) in Part I [1] the definition of the mass flow function

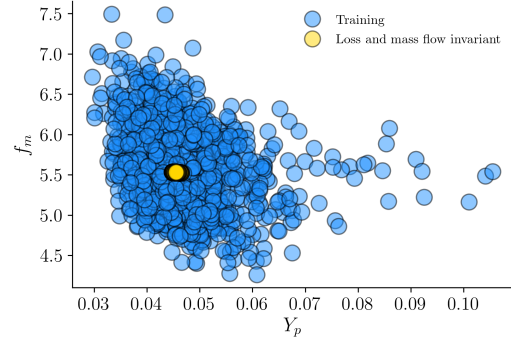$$f_m = \frac{\dot{m}\sqrt{T_{01}}}{p_{01}} \times 10^4. \tag{13}$$

Fig. 3. Loss and mass flow function for random training designs (blue) and invariant designs within the intersection of the inactive subspaces (yellow).

Following a similar approach to the calculation of the inactive subspace for loss, we first model the mass flow function using an orthogonal polynomial series via regression on a data-set consisting of $M = 800$ training samples and 200 validation samples. These are drawn from a 20D design space based on free-form deformation—the same data-set used in part I. In this case, it is found that a linear polynomial ($p = 1$) suffices to yield an R^2 value of 0.993 on the validation data (see Figure 2). Now, we could compute the associated gradient covariance matrix (2) by sampling gradient evaluations, but since the model is linear, the gradient is constant across the input domain. This implies that we can read off the 1D active subspace as the linear polynomial coefficients to the design parameters, i.e.,

$$f_m \approx u_{f_m} = \mathbf{W}_{f_m}^T \mathbf{x}, \tag{14}$$

where $\mathbf{W}_{f_m} \in \mathbb{R}^{20\times1}$ specifies the active subspace, and u_{f_m} is the active coordinate for f_m.

Equipped with the active subspaces for both loss ($\mathbf{W}_{Y_p} \in \mathbb{R}^{20\times1}$) and mass flow ($\mathbf{W}_{f_m} \in \mathbb{R}^{20\times1}$), we can now find the intersection of their inactive subspaces via the method described in Section 2.1. The active subspaces are linearly independent and the intersection $\mathcal{V}_{int}$ is $20-(1+1)=18$-dimensional. Then, using the hit-and-run sampling strategy described in Part I, $H = 5000$

samples are generated from $\mathcal{V}_{int}$. The invariance in both objectives is verified by picking 500 designs and running them through the CFD solver to obtain their true loss and mass flow function values, which we plot against the corresponding values for the training data-set in Figure 3. This confirms that the designs in the intersection are approximately invariant in both objectives.

The blade envelope for invariance in loss and mass flow is shown in Figure 4. Compared to the blade envelope for loss in Part I, the control zone in this case is narrower from mid-chord to the trailing edge. The typical curvature of the sample profiles is also milder. Using the sample profiles, the ensemble mean and covariance matrix can be calculated and the Mahalanobis distance can be computed for test profiles, as in Section 3.4 of Part I [1]. Figure 5 shows the application of a trained logistic function on profiles from two different design spaces—the 20D design space from which the training data is drawn, and the 30D design space described in Section 4.4 of Part I [1]. The samples from the 30D space are colored according to how close their loss and mass flow values are to nominal values calculated according to the following formula

$$\delta(\mathbf{x}) = \sqrt{(\mathbf{y}(\mathbf{x})-\boldsymbol{\mu}_y)^T \mathbf{S}_y^{-1}(\mathbf{y}(\mathbf{x})-\boldsymbol{\mu}_y)}, \tag{15}$$

where $\mathbf{y}(\mathbf{x}) = [Y_p(\mathbf{x}), f_m(\mathbf{x})]^T$, and $\boldsymbol{\mu}_y$ and $\mathbf{S}_y$ are the ensemble
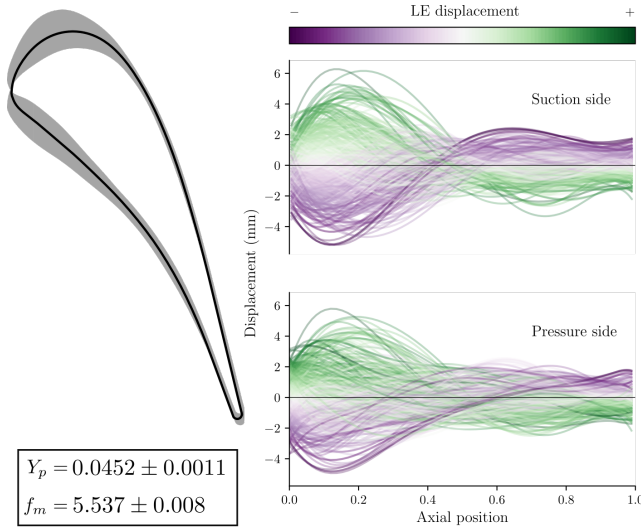
Fig. 4. The blade envelope for loss and mass flow.

mean and covariance matrix of **y** over all training designs respectively. Note that $\delta(\mathbf{x})$ is another Mahalanobis distance, this time defined over the two output objectives. This is analogous to that defined over geometric profiles (see Section 3.4 of Part I [1]). Here, we see similar results as those obtained for a single objective in Sections 4.3 and 4.4 of Part I [1], showing that the Mahalanobis distance defined over geometries is an effective diagnostic for identifying profiles invariant in multiple objectives.

3.2 Setting the peak isentropic Mach number

In addition to control over the total quantity of loss, the distribution of this loss is also of interest to the engineer as a means to judge the quality of a component. The invariant subspace of loss and mass flow contains geometries that yield different flow profiles, resulting in a range of loss distributions, some preferred over others. According to Clark [6], surface flow characteristics defined by primal flow quantities can be used as predictors of aerodynamic performance, differentiating between desirable and undesirable flow profiles. In his work, four features of the surface isentropic Mach number distribution are isolated:

1. the peak isentropic Mach number,
2. the surface fraction of the peak isentropic Mach number,
3. the leading edge isentropic Mach number, and
4. the pressure side isentropic Mach number prior to acceleration.
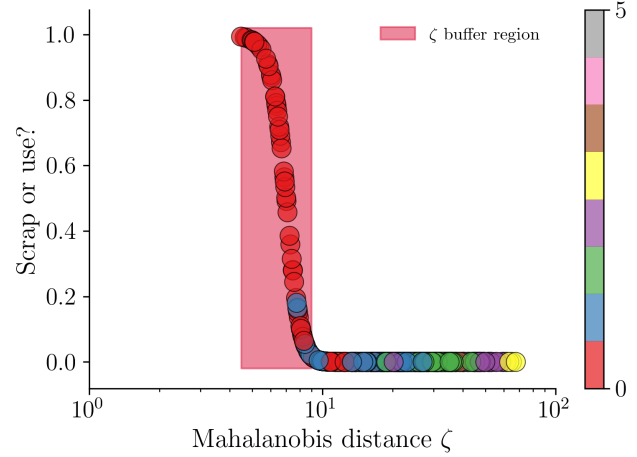
In this section, the method of vector-valued dimension reduction is applied to modulate the peak of this distribution. The isentropic Mach number is defined in Equation (3) in Part I [1] as

$$M(s) = \sqrt{\frac{2}{\gamma-1}\left(\left(\frac{p_{01}}{p(s)}\right)^{\frac{\gamma-1}{\gamma}} - 1\right)}, \quad (16)$$

where s indicates the location on the surface. This can be discretized as in (4) into $\mathbf{M}(\mathbf{x})$.
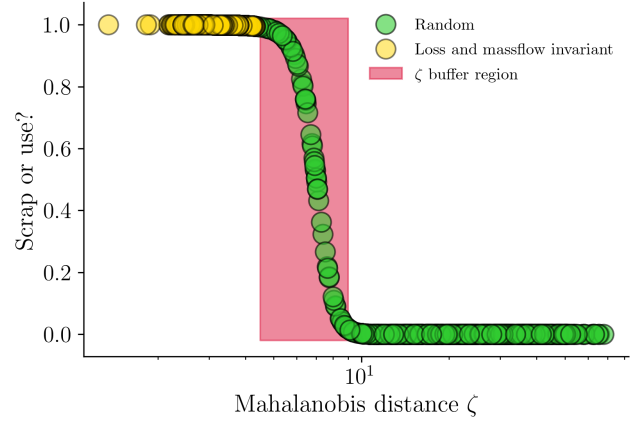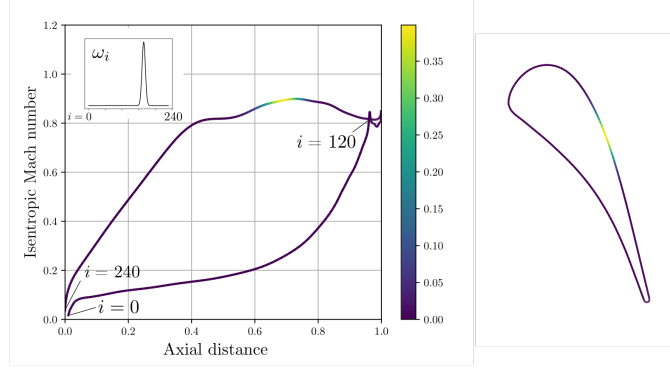
(a)

(b)

Fig. 5. Trained logistic function for applying a binary scrap-or-use decision on profiles, requiring invariance in both the loss and mass flow function; (a) shows samples from the 20D design space and (b) shows samples from the 30D design space described in Section 4.4 of Part I [1]. In (b), the samples are colored according to their deviation from nominal loss and mass flow, quantified via (15).
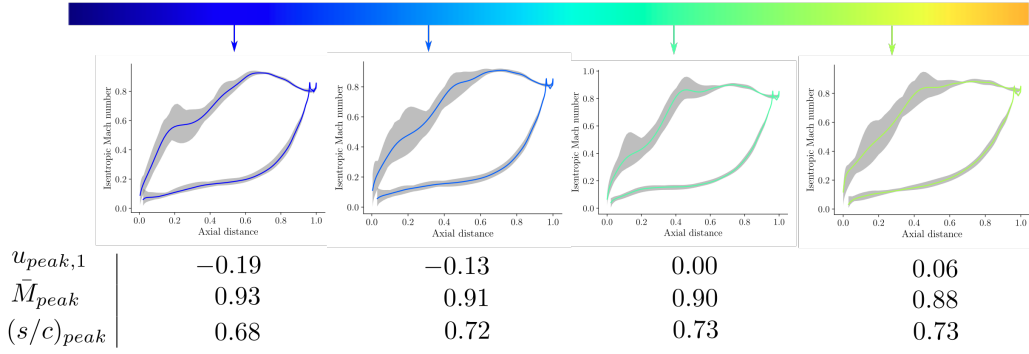
Our goal is to extract a small number of active coordinates which capture the variation of the peak isentropic Mach number. Following the recipe in Section 2.2, the gradient covariance matrix for the isentropic Mach number distribution $\boldsymbol{H}_{peak}$ is required, where the weight matrix $\boldsymbol{R}_{peak}$ is set such that diagonal elements corresponding to the peak are large. This is illustrated in Step I in Figure 6, where the color indicates the magnitude of the corresponding weights. To compute the necessary gradients with respect to each nodal isentropic Mach number, quadratic models are trained at each node using uniformly sampled designs from the full design space. When evaluated on independent validation data, the models on nodes with non-zero weights all achieved an R^2 value above 0.99. Note that we set a non-zero weight on multiple nodes in the vicinity of the peak. As we change the height of the peak, it is observed that the chord-wise location of the peak shifts slightly. This effect is accommodated by smoothing out the weights to a small region near the peak.

I. Set weights on the isentropic Mach number distribution

II. Tune desired peak isentropic Mach number

Increasing $u_{peak,1}$ ⟶

$u_{peak,1}$	-0.19	-0.13	0.00	0.06
$\bar{M}_{peak}$	0.93	0.91	0.90	0.88
$(s/c)_{peak}$	0.68	0.72	0.73	0.73

III. Sample within the inactive subspace to get a blade envelope

$$Y_p = 0.0452 \pm 0.0011$$
$$f_m = 5.537 \pm 0.008$$
$$M_{peak} = 0.901 \pm 0.014$$
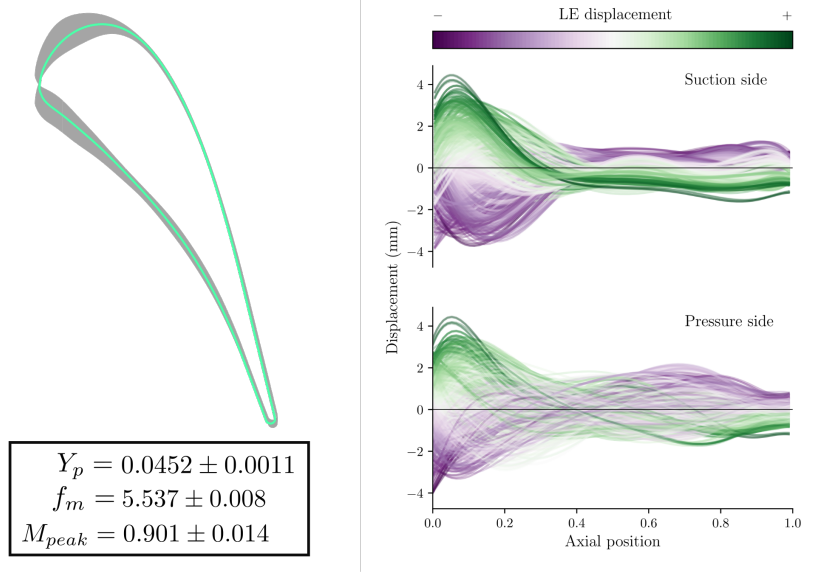
Fig. 6. Conditional tuning of the peak isentropic Mach number.

Carrying out the eigendecomposition of $\boldsymbol{H}_{peak}$,

$$\boldsymbol{H}_{peak} = \begin{bmatrix} \boldsymbol{W}_{peak} & \boldsymbol{V}_{peak} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Lambda}_{peak,1} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\Lambda}_{peak,2} \end{bmatrix} \begin{bmatrix} \boldsymbol{W}_{peak}^T \\ \boldsymbol{V}_{peak}^T \end{bmatrix}. \quad (17)$$

The eigenvalues of $\boldsymbol{H}_{peak}$ are shown in Figure 7. The active coordinates can be defined using the first two eigenvectors for ap-

proximate control over the peak isentropic Mach number, namely,

$$\begin{aligned} \mathbf{u}_{peak} &= \begin{bmatrix} u_{peak,1} & u_{peak,2} \end{bmatrix}^T \\ &= \boldsymbol{W}_{peak}^T \mathbf{x}. \end{aligned} \quad (18)$$

This set of active coordinates can be combined with those corresponding to the loss coefficient and mass flow to form the
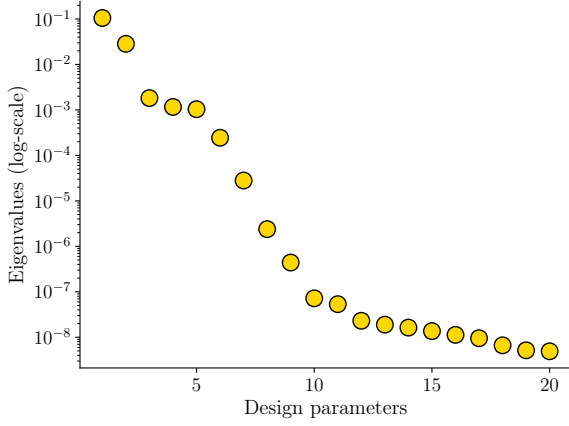
Fig. 7. Eigenvalues of $\mathbf{H}_{peak}$, weighted for the peak isentropic Mach number.

following vector

$$\mathbf{u} = \begin{bmatrix} u_{loss} & u_{fm} & u_{peak,1} & u_{peak,2} \end{bmatrix}^T \qquad (19)$$

To control the peak isentropic Mach number while keeping the loss and mass flow constant, the procedure described in Section 2.4 can be used. In this example, we only tune the first active coordinate $u_{peak,1}$ for illustrative purposes. The effect of changing this variable is shown in Step II in Figure 6. Increasing $u_{peak,1}$ results in the flattening of the isentropic Mach number distribution. The location of the peak moves gradually towards the trailing edge. If the coordinate is increased further, the peak shifts to the junction at the start of the plateau near mid-chord. The uncertainty bands around the distributions indicate the variation in the isentropic Mach number distribution for geometries with the same values of $\mathbf{u}$.

After setting appropriate values for the active coordinates, the hit-and-run sampling algorithm introduced in Part I can be used to sample designs with these coordinates. A blade envelope can be formed from these samples, following the same procedure as the previous examples. Step III in Figure 6 shows the blade envelope with $\mathbf{u} = \mathbf{0}$. Compared with the previous blade envelopes, the tolerance on the profile from mid-chord to the trailing edge is reduced, especially on the suction side. Using the sample profiles, the tolerance covariance matrix and ensemble mean can be calculated as in Part I to arrive at an automatic decision criterion for the scrapping of manufactured blades, taking into account all three objectives.

3.3 Setting the leading edge isentropic Mach number

The procedure can be repeated for a different setting of the weights on the isentropic Mach number distribution. Setting the weights as in Step I of Figure 8, the focus is now placed on the isentropic Mach number near the leading edge. Carrying out the eigendecomposition of the vector gradient covariance matrix

$\mathbf{H}_{LE}$,

$$\mathbf{H}_{LE} = \begin{bmatrix} \mathbf{W}_{LE} & \mathbf{V}_{LE} \end{bmatrix} \begin{bmatrix} \mathbf{\Lambda}_{LE,1} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_{LE,2} \end{bmatrix} \begin{bmatrix} \mathbf{W}_{LE}^T \\ \mathbf{V}_{LE}^T \end{bmatrix}, \qquad (20)$$

the spectrum shown in Figure 9 is obtained. Choosing the first four eigenvectors as the basis for the active subspace $\mathbf{W}_{LE}$, the active coordinates are specified as

$$\begin{aligned} \mathbf{u}_{LE} &= \begin{bmatrix} u_{LE,1} & u_{LE,2} & u_{LE,3} & u_{LE,4} \end{bmatrix}^T \\ &= \mathbf{W}_{LE}^T \mathbf{x}. \end{aligned} \qquad (21)$$

Concatenating this set of coordinates with the active coordinates of the loss and mass flow, we get

$$\mathbf{u} = \begin{bmatrix} u_{loss} & u_{fm} & u_{LE,1} & u_{LE,2} & u_{LE,3} & u_{LE,4} \end{bmatrix}^T. \qquad (22)$$

The effect of tuning the first active coordinate of the leading edge isentropic Mach number $u_{LE,1}$ is shown in Step II of Figure 8. As the coordinate is increased, the isentropic Mach number is reduced towards the leading edge, and the acceleration is milder and more distributed towards mid-chord. Step III shows the blade envelope obtained with samples setting $\mathbf{u} = \mathbf{0}$. The geometric tolerance near the leading edge is tightened, especially on the suction side.

4 PRACTICAL UTILITY AND DISSEMINATION

Thus far in this two-part paper, we have enumerated the core computational techniques required for generating a blade envelope under multiple constraints. We have articulated its utility in design (both forward and inverse) by identifying and exploiting the inactive subspace. Although all our efforts have focused on the LS89, the concept of a blade envelope extends well beyond any given airfoil, and indeed an isolated 2D cross-section. We envision blade envelopes forming a critical part of the design stage, facilitating discussions on trade-offs and tolerances.

Beyond design, a key contribution of our work is a machine learning technique for deciding whether a manufactured blade's tolerance variations are permissible or not. This program can be easily incorporated within an existing coordinate measuring machine protocol and used in manufacturing assembly lines, with the aim of moving away from existing broad spectrum tolerances and opting for a more bespoke, design-centric one, which may offer opportunities for relaxing tolerances—thereby reducing manufacturing costs. Additionally, we see tremendous utility of the blade envelope in health monitoring; a topic we address in detail in a forthcoming paper. When inspecting blades that have gone through several hundred hours of operation, using either detailed GOM scans or borescope images, we can use the blade envelope to query whether a given blade needs to be refurbished or not.
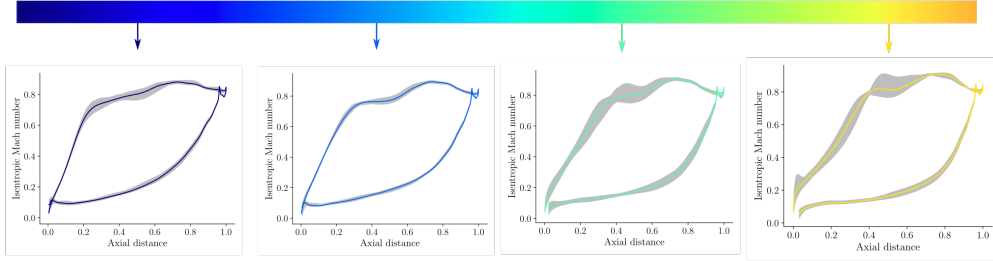
The one high-level challenge that remains is how, in practice, the design, manufacturing and health monitoring teams cohesively converge and take advantage of a blade envelope, given

7

I. Set weights on the isentropic Mach number distribution
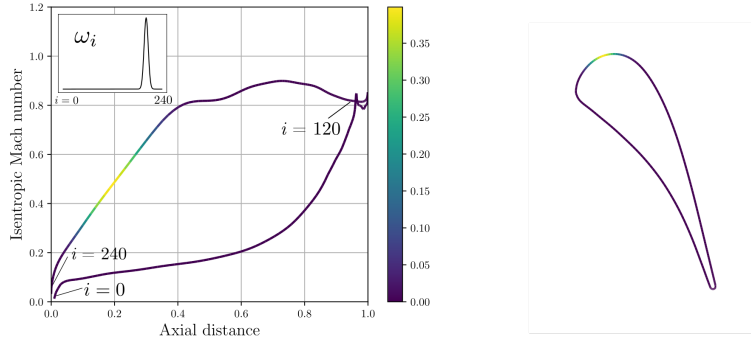
II. Tune desired leading edge isentropic Mach number

Increasing $u_{LE,1}$ ⟶

$u_{LE,1}$	-1.00	-0.50	0.00	0.50
$\bar{M}_{LE}$	0.63	0.57	0.50	0.44

III. Sample within the inactive subspace to get a blade envelope

$$Y_p = 0.0452 \pm 0.0011$$
$$f_m = 5.537 \pm 0.008$$
$$M_{LE} = 0.504 \pm 0.061$$
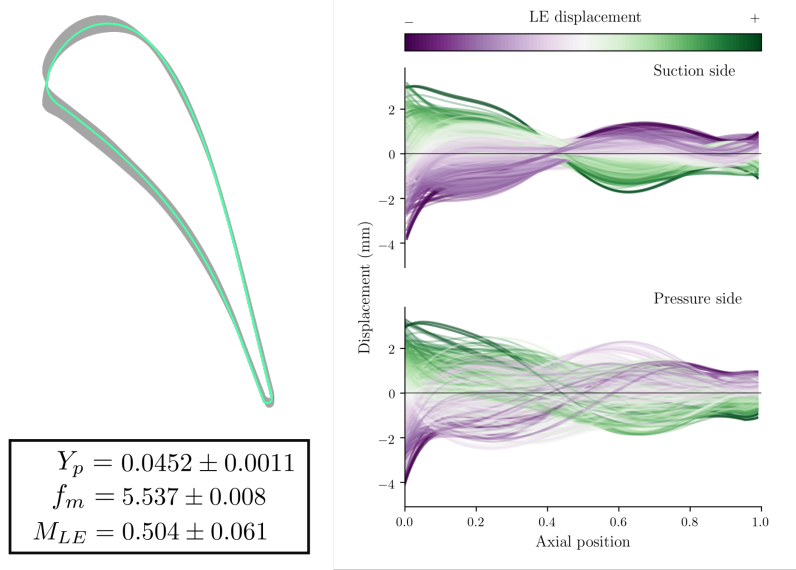
Fig. 8.   Conditional tuning of the leading edge isentropic Mach number.

that these teams are seldom collocated and may even be in different continents. To address this, we offer a simple plug-in from Effective Quadratures (used for doing the computational heavy lifting) [10] to a 3D modelling software package (we opt for Blender [11] given its open source nature). The latter can be easily adapted to industrial computer aided design (CAD) workflows.

We present a CAD representation of the controlled loss and mass flow rate blade envelope, shown in Figure 1, in the Intro-duction. The purple and green airfoils correspond to profiles with negative and positive leading edge displacement respectively, just as plotted before, but now superimposed with the nominal blade (shown in black). Our objective in showing these images is to demonstrate how these envelopes, once frozen for the objectives desired, can be 3D printed—either at scale or as a scaled-up model—and can sit on the desktops of designers, manufacturing engineers, and engine maintenance inspectors (Figure 10). These
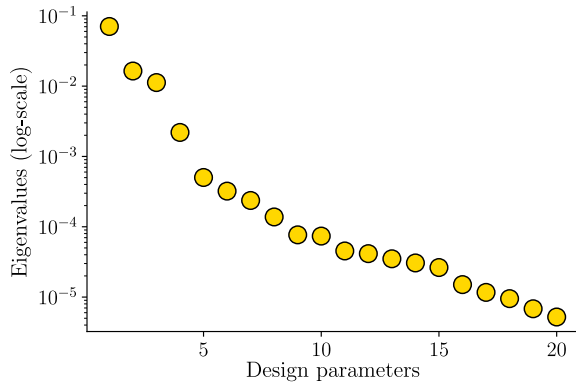
Fig. 9. Eigenvalues of $\boldsymbol{H}_{LE}$, weighted for the leading edge isentropic Mach number.
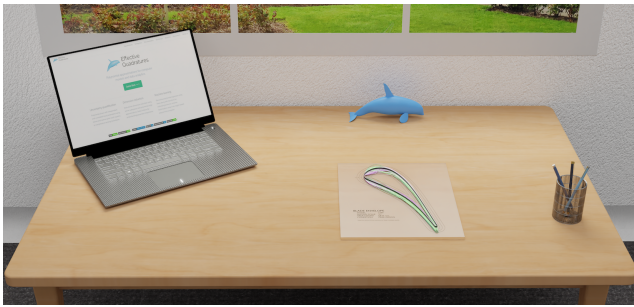

Fig. 10. The blade envelope from Figure 1 can be suitably scaled for 3D printing and placed on desktops.

models can easily be transported, or indeed printed at different sites, and used to foster discussions on design and tolerance decisions. For completeness, close-up and isometric views of this blade envelope are shown in Figures 11 and 12.

5 CONCLUSIONS

This paper extends the blade envelope framework to accommodate multiple objectives. These objectives can either be scalar-valued or a distribution defined on a surface. To take multiple objectives into consideration, an approach based on the intersection of inactive subspaces is described. For profile objectives, we propose a method based on vector-valued dimension reduction to seek an inactive subspace based on weights placed on different parts of the profile. Combining the inactive subspaces of the loss coefficient, mass flow function and the surface isentropic Mach number distribution, we demonstrated use of these methods to derive a blade envelope that delineates a boundary for designs that satisfy the constraints placed on all three objectives.

The collection of active coordinates corresponding to multiple objectives provide independent control over multiple objectives, by changing the active coordinates of one objective and keeping others constant. This is demonstrated through tuning the peak and leading edge isentropic Mach numbers within the inactive subspace of the loss and mass flow.

References

[1] Wong, C. Y., Seshadri, P., Scillitoe, A., Duncan, A., and Parks, G. T. "Blade Envelopes Part I: Concept and Methodology". *Submitted to Journal of Turbomachinery*.

[2] Torreguitart, S., Verstraete, T., and Mueller, L., 2018. "Optimization of the LS89 Axial Turbine Profile Using a CAD and Adjoint Based Approach". *International Journal of Turbomachinery, Propulsion and Power,* **3**(3), Sept., p. 20.

[3] Montanelli, H., Montagnac, M., and Gallard, F., 2015. "Gradient Span Analysis Method: Application to the Multipoint Aerodynamic Shape Optimization of a Turbine Cascade". *Journal of Turbomachinery,* **137**(9), Sept., p. 091006.

[4] Seshadri, P., Yuchi, S., Parks, G., and Shahpar, S., 2020. "Supporting multi-point fan design with dimension reduction". *The Aeronautical Journal,* **124**(1279), p. 1371–1398.

[5] Seshadri, P., Shahpar, S., Constantine, P. G., Parks, G. T., and Adams, M., 2017. "Turbomachinery Active Subspace Performance Maps". *Journal of Turbomachinery,* **140**(4), Apr., p. 041003.

[6] Clark, C. J., 2019. "A Step Towards an Intelligent Aerodynamic Design Process". In ASME Turbo Expo 2019: Turbomachinery Technical Conference and Exposition, American Society of Mechanical Engineers Digital Collection.

[7] Golub, G. H., and van Loan, C. F., 2013. *Matrix Computations*, fourth ed. Johns Hopkins University Press, Baltimore, Apr.

[8] Wong, C. Y., Seshadri, P., Parks, G. T., and Girolami, M., 2020. "Embedded ridge approximations". *Computer Methods in Applied Mechanics and Engineering,* **372**, p. 113383.

[9] Zahm, O., Constantine, P. G., Prieur, C., and Marzouk, Y. M., 2020. "Gradient-Based Dimension Reduction of Multivariate Vector-Valued Functions". *SIAM Journal on Scientific Computing,* **42**(1), Jan., pp. A534–A558. Publisher: Society for Industrial and Applied Mathematics.

[10] Seshadri, P., and Parks, G., 2017. "Effective-quadratures (EQ): Polynomials for computational engineering studies". *The Journal of Open Source Software,* **2**, pp. 166–166.

[11] Blender Online Community, 2018. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam.
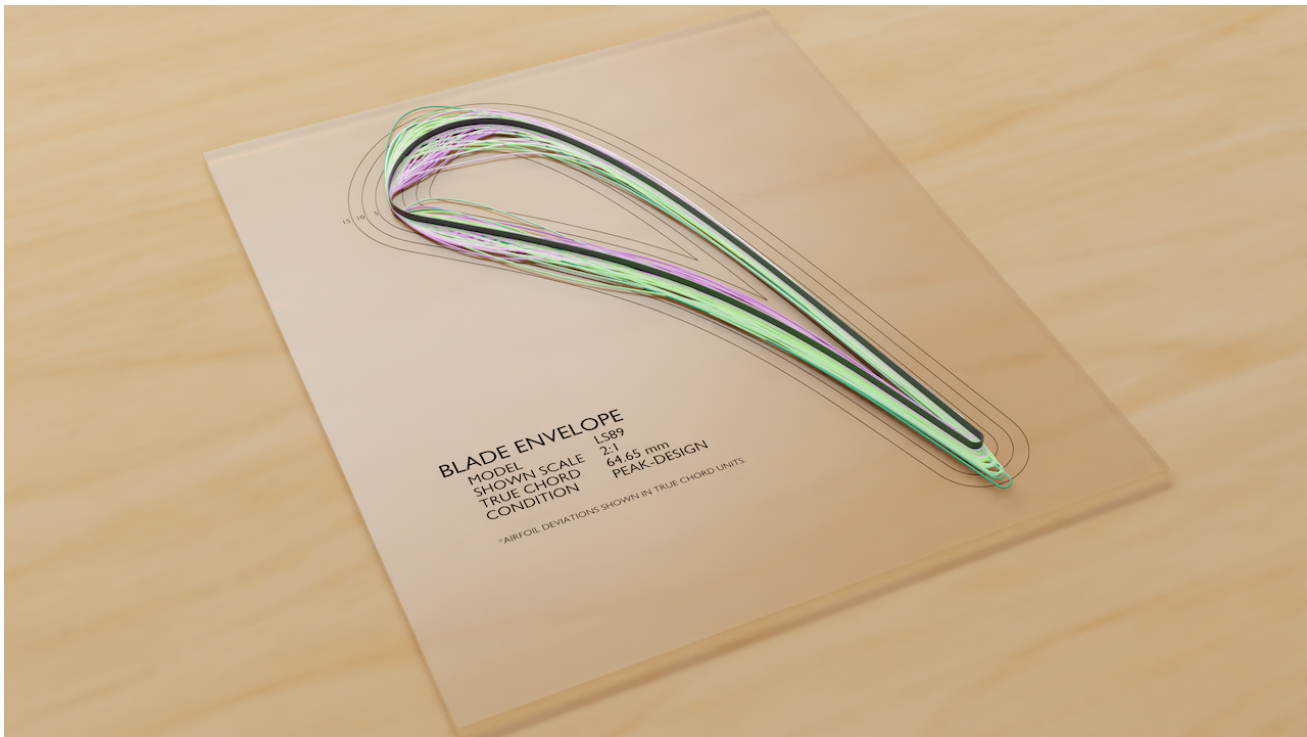
9

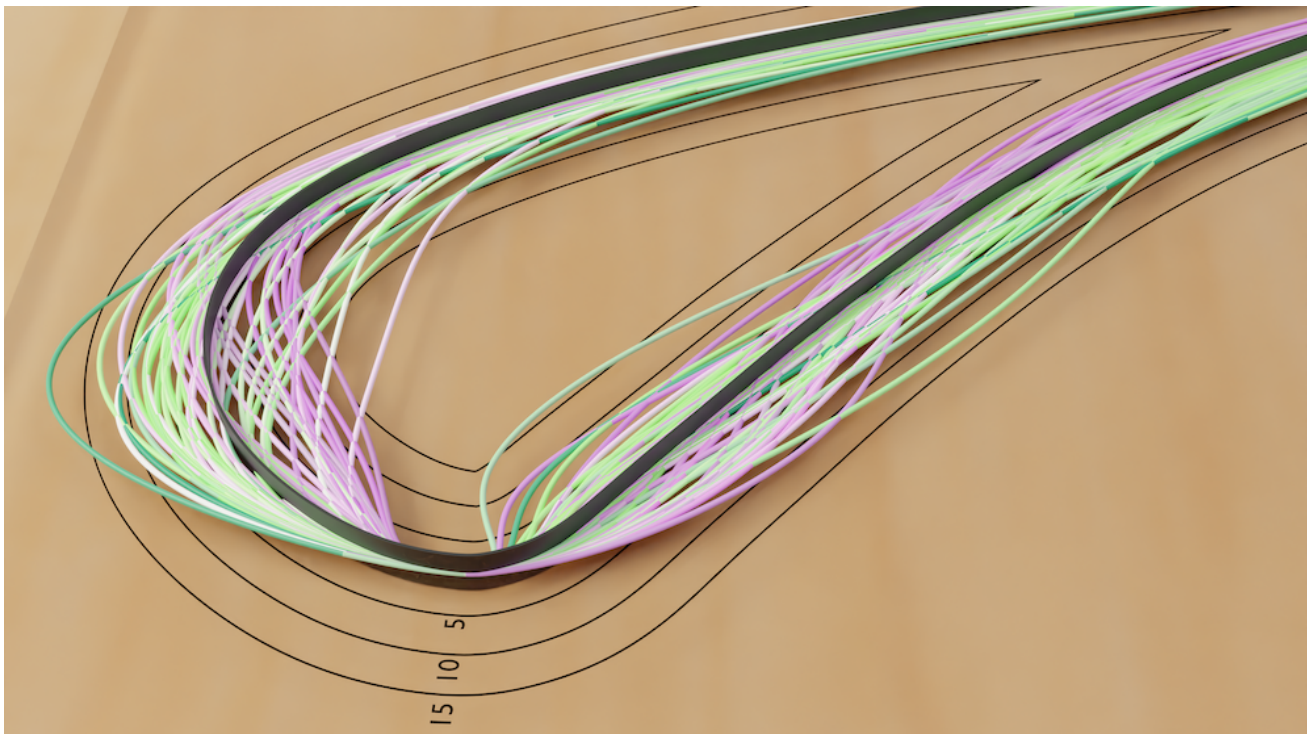Fig. 11. An isometric view of the blade envelope in Figure 1.

Fig. 12. Close-up of the leading edge of the blade envelope in Figure 1. The demarcated iso-contour lines denote displacements at 5, 10 and 15 mm away from the nominal blade.