

Separable shape tensors for aerodynamic design

Zachary J. Grey^{1,*}, Olga A. Doronina² and Andrew Glaws²

¹National Institute of Standards and Technology, Applied and Computational Mathematics Division, Boulder, CO 80305, USA

²National Renewable Energy Laboratory, Computational Science Center, Golden, CO 80401, USA

*Corresponding author. E-mail: zachary.grey@nist.gov

Abstract

Airfoil shape design is a classical problem in engineering and manufacturing. In this work, we combine principled physics-based considerations for the shape design problem with modern computational techniques using a data-driven approach. Modern and traditional analyses of two-dimensional (2D) and three-dimensional (3D) aerodynamic shapes reveal a flow-based sensitivity to specific deformations that can be represented generally by affine transformations (rotation, scaling, shearing, and translation). We present a novel representation of shapes that decouples affine-style deformations over a submanifold and a product submanifold principally of the Grassmannian. As an analytic generative model, the separable representation, informed by a database of physically relevant airfoils, offers: (i) a rich set of novel 2D airfoil deformations not previously captured in the data, (ii) an improved low-dimensional parameter domain for inferential statistics informing design/manufacturing, and (iii) consistent 3D blade representation and perturbation over a sequence of nominal 2D shapes.

Keywords: airfoils, manifolds, generative model, shape tensors, Grassmannian, data-driven

1. Introduction

We begin by reviewing aspects of airfoil and blade/wing design to establish a motivation for the work. What follows is intended to serve as a detailed overview of the theoretical foundations for computations. Implementations and examples are available on GitHub (Doronina et al., 2022).

Many artificial intelligence (AI)-aided design and manufacturing algorithms rely on shape representation methods to manipulate shapes in order to study sensitivities, approximate inverse problems, and inform optimizations. Two-dimensional (2D) cross-sections of aerodynamic structures, such as aircraft wings or wind turbine blades, also known as “airfoils”, are critical engineering shapes whose design and manufacturing can have significant impacts on the aerospace and energy industries. Research into AI and machine learning (ML) algorithms involving airfoil design for improved aerodynamic, structural, and acoustic performance is a rapidly growing area of work (Chen et al., 2019; Glaws et al., 2022a, b; Li et al., 2019; Seshadri et al., 2018; Wang et al., 2022; Yang et al., 2022; Yonekura & Suzuki, 2021; Zhang et al., 2018).

Although airfoil shapes have been studied extensively and can appear relatively benign, their representation and design are complex due to their extreme operating conditions and the highly sensitive relationship between shape deformations and changes in aerodynamic performance. In this context, innovations specifically related to computational domains are of paramount importance for the future of computational fluid dynamics (Slotnick et al., 2014). Improved shape parameter domains will enable future parametrized model reductions (Benner et al., 2015; Willcox & Peraire, 2002) to balance computational costs, improve designs, and make computations more explainable and interpretable.

In this work, we explore a data-driven approach that uses a matrix (2-tensor) manifold framework to parametrize (or learn) a manifold of airfoil shapes. The resulting set of deformations to air-

foil shapes separates important, and often constrained, affine deformations. Modern airfoil design incorporates constrained design characteristics of twist (i.e., angle of attack) and scale, which must be fixed or treated independently of higher order deformations to the shape. Our approach decouples these two aspects of airfoil design and offers new interpretations of a space of shapes not previously considered—i.e., “learning” a manifold of discrete shapes as submanifolds built from parent matrix manifolds. In the following subsections, we provide a brief overview of the airfoil representation scheme and demonstrate its flexibility over current methods, including the capability to extend from 2D airfoils to full three-dimensional (3D) shapes, such as wind turbine blades. The results are predicated on parametrizations over well-understood manifolds, offering an “analytic generative model” for airfoil shapes, in contrast to alternative AI-based generative models and other non-linear dimension reductions (i.e., manifold learning). Implementations and examples are available (Doronina et al., 2022).

1.1. Defining an airfoil

We review general concepts for airfoil design, highlighting the fact that airfoil shapes are defined independent of planar rotations and scaling despite these deformations being highly sensitive to aerodynamic quantities of interest. This presents a challenge to define a general domain of airfoil shapes which is also independent of a chosen basis expansion representation.

Airfoil design seeks an optimal planar shape that satisfies desired design criteria—e.g., a specific lift and drag under particular operating and atmospheric conditions. Quantitatively, a given airfoil is typically characterized by its aerodynamic properties using lift and drag profiles, or “polars,” which are defined as univariate functions of these quantities over a range of planar rotations (or angles of attack). These profiles characterize important operational behavior and sensitivities over a continuous set of planar

rotations. As such, functionals of these univariate profiles are often used to inform quantities of interest for optimization, approximation, and inverse problems—e.g., $\int (\zeta \circ f)(\theta) d\theta$, where f is lift, drag, or the ratio of lift to drag as a function of scalar rotation angle θ ; and ζ acts as a transformation representing the desired characteristics of the polar. Specifying functionals over a planar rotation of the shape to characterize the airfoil operational profile suggests that a given airfoil is best characterized independently of these rotations. The rigid motion is “integrated away” by the definition of operational effectiveness as a functional over the angle of attack.

To study shapes, one approach is to represent airfoils by the class-shape transformation (CST; Kulfan, 2008), which partitions the shape into upper (suction side) and lower (pressure side) surfaces of the airfoil. The upper and lower surfaces are parametrized by the coefficients of two truncated Bernstein polynomial series. The two distinct upper and lower surface polynomials are multiplied by rational “class functions” that ensure at least two roots at zero and one where the distinct curves (as graphs of the two polynomials) connect. Consequently, deformations applied by modifying the coefficients of the expansion retain fixed orientation by design, such that the leading edge point (root at zero) and trailing edge point (root at one) of the polynomial expansions are fixed.

Alternatively, the airfoil can be parametrized using planar Bezier splines, or general B-splines, with specified control points (Hosseini & Moetakef-Imani, 2016). Additionally, shapes can be inferred from noisy data with weighted local least squares (Ghorbani & Khameneifar, 2021). The curve is typically parametrized with complicated (and often subjective) constraints on each control point to help regularize deformations to achieve certain behavior. Previous studies have used splines to design several families of airfoils for modern megawatt-scale wind turbines (Li et al., 2019). These representations are typically fixed to ignore rotations of the shape and scaled to obtain a “unit-chord length” such that the Euclidean distance from the “leading edge” to the “trailing edge” is one.

However, the discussed expansion representations (and others) are often coupled with a highly sensitive linear scaling of the shape as inferred from the physics and modeling (Glaws et al., 2022b; Grey & Constantine, 2018; Li et al., 2019; Seshadri et al., 2018). Specifically, general representations of shapes as curves remain coupled to large-scale, affine-type deformations—deformations resulting in significant and relatively well-understood physical impacts on aerodynamic performance like changes in thickness and camber. In contrast, smaller scale undulating perturbations are of increasing interest to airfoil design problems (Glaws et al., 2022b) and to the study of impacts of manufacturing defects and damage (Ge et al., 2019). This coupling between physically meaningful affine deformations and higher order perturbations in shapes confounds deformations of interest in the design process. Further details and examples of this coupling between deformations of interest and affine deformations are presented in Section 2.1.

Moreover, defining appropriate domains and constraints informing meaningful design spaces for the coefficients of Bernstein polynomials or B-splines can be very challenging from one problem to the next. Defaulting to bounded ranges about nominal coefficients may not be expressive enough to cover more diverse classes of airfoils. Restrictive and complicated choices of parameter domains make it challenging to diversify designs and take full advantage of the flexibility offered by AI-aided design and manufacturing. That is, interpreting the dependencies between perturbations to CST or spline parameters and subsequent

changes to the resulting shapes can be challenging. This makes it difficult to select informed prior distributions generating “reasonable” aerodynamic shapes that underpin fundamental AI and inverse problem computations. For example, assumed uniform and Gaussian distributions over shape parameters often translate to complex, non-intuitive distributions over any space of shapes—distributions which may be multi-modal and consisting of several disjoint clusters of shapes.

Constraints imposed by the chosen representation, e.g., CST or splines, tend to fix the orientation of the airfoil in the plane. The intuition is that airfoil shapes are characterized by perturbations that are independent of rotation, given design functionals defined over profiles of rotation angle. The desired 2D rotational invariance and physics-based interpretations motivate our development of a novel separable representation of shapes—independent of a chosen basis expansion representation—for next-generation aerodynamic design.

1.2. Defining a blade

We describe challenges associated with defining a 3D blade/wing design by interpolating a sequence of 2D shapes. The 2D cross-sections are designed and constrained by affine deformations encoding scale, rotation, and position properties that are often fixed by structural constraints or legal regulations. We establish a clear need for a general framework which can accomplish interpolation of 2D shapes independent of prescribed affine deformations. Additionally, new representations should be free from specific 2D basis expansion representations since total parameter count scales poorly for 3D design.

As a natural extension of 2D airfoils, 3D aerodynamic design considers the construction of a blade or wing from landmark airfoils with specific operational characteristics along the length of the blade. Blades and wings are generally represented by 2D cross-sections extruded along a spanwise axis in 3D—e.g., wind turbine blades or gas turbine blades (Hosseini & Moetakef-Imani, 2016). In contrast to the 2D airfoil design problem, the relative orientation, scaling, and position of the airfoil shapes along the span is heavily coupled to the final design of the full 3D shape—which has structural and/or regulatory implications in the specific case of wind turbines. For example, the 2D airfoils are often scaled to achieve an appropriate Reynolds number within the 3D blade, per modeled flow conditions. These shapes are then carefully rotated and translated smoothly along the span axis—imparting a twist and bend to the blade—to achieve specific operational characteristics from the hub to the tip of the 3D design. This procedure amounts to interpolating a sequence of 2D airfoils to define the 3D blade.

The proposed design procedure is challenging with existing airfoil parametrizations as the total number of parameters defining the blade scales with the number of 2D cross-sections and parameters may change dramatically from one airfoil cross-section to the next. It is not clear how conventional, potentially high dimensional, parametrizations can be interpolated from one 2D shape to the next to retain the sought affine characteristics of the blade.

The methods presented here transform existing airfoil definitions in order to (i) inform rotation- and reflection-invariant designs of 2D airfoil shapes and (ii) extend 2D designs to 3D blades by enabling spanwise airfoil interpolations that decouple blade-shape perturbations from specified scalings and rotations. The key characteristic in both the 2D and 3D design tasks is that we seek “separability” between airfoil deformations that scale and rotate the shape—defined by affine deformations—and those that introduce local, high-order undulations in the surface. Introducing

the separability then offers designers the ability to independently select the types of deformations that are most important for their application.

Further, we seek to accomplish these design tasks *free from any specific expansion representation* involving CST or splines—i.e., in a manner that does not parametrize the form of a specific basis expansion. Lastly, we explore a concept of *consistently deforming* the airfoils defining a 3D blade such that the total parameter dimensionality is independent of the number of provided 2D cross-sections. This leads to a novel framework for the design of next-generation aerodynamic shapes over a principled choice of reduced dimension domain.

1.3. Contributions

This work formally develops the use of specific matrix manifolds as underlying parent topologies for defining an affine and expansion independent parametrization of a space of discrete shapes. In general, this reveals a systematic approach to “learning” a sub-manifold of shapes through a vector space of n -by-2 full-rank matrices. Our contributions include:

- (i) The definition of separable forms of discrete shapes independent of affine deformations and designed expansions for 2D and 3D design.
- (ii) A metric space of discrete shapes with an improved notion of distance between shapes over a novel data-driven domain definition.
- (iii) Detailed computational routines for learning a manifold-valued analytic generative model of shapes in seconds.
- (iv) Definition of a 3D blade/wing using affine-independent interpolation of designed 2D airfoil shapes.
- (v) A novel approach for 3D design involving consistent blade/wing deformations—minimizing parameter count and deforming the blade in an intuitive way.

We note that these novel interpretations applied to airfoil design are closely related to the pioneering seminal work of David G. Kendall in 1977 summarized in short (Kendall, 1989) and elaborated in detail (Kendall et al., 2009). As such, modern treatments of discrete shapes are commonly referred to as Kendall shape spaces.

2. Discrete Representation & Deformation

We introduce shapes defined by smooth curves inducing discrete shapes as matrices that are independent of a choice of basis expansion. In detail, we discuss deformations to and standardization of these discrete shapes using linear algebra decompositions. We also analyze the convergence of data preprocessing and provide comparisons to AI/ML generative models. Finally, we cover detailed Riemannian interpretations informing computations necessary to build implementations from scratch.

We begin by developing the discretized representations of 2D airfoil shapes as the foundation for data-driven parametrizations that will facilitate 2D and 3D aerodynamic design. The challenge of interest in modern aerospace shape design is to define shape deformations that are independent of the well-studied and aerodynamically sensitive affine deformations to a shape. Affine deformations over planar coordinate axes, typically parametrized as scaling and rotation, can greatly affect aerodynamic performance (lift and drag profiles) of an airfoil. Additionally, scaling is not limited to simply increasing the shape volume but can also include independent vertical or horizontal scaling as well as the shearing

the shape’s image in the plane. Thus, it is important to decouple affine deformations from higher order deformations as *undulations* and study them independently in 2D design. In this complementary context, we define an undulation in the shape as any remaining deformation that is not represented by a linear deformation—i.e., informally considered higher order or non-linear variations in the shape. Additionally, we require geometric representations that preserve these often carefully chosen affine characteristics for 3D design.

A 2D shape can be represented as a boundary defined by the open (i.e., injective or one-to-one) or closed (i.e., injective, except at endpoints) curve $\mathbf{c} : \mathcal{I} \subset \mathbb{R} \rightarrow \mathbb{R}^2 : s \mapsto \mathbf{c}(s)$, where \mathcal{I} is a compact domain. Without loss of generality, we can assume that $\mathcal{I} = [0, 1]$. In practice, we consider a discrete representation the 2D airfoil shape as an ordered sequence of n landmarks $(\mathbf{x}_i) \in \mathbb{R}^2$ for $i = 1, \dots, n$ and $n \geq 3$ along the curve $\mathbf{c}(s)$. That is, we have landmark points $\mathbf{x}_i = \mathbf{c}(s_i)$ for $0 \leq s_1 < s_2 < \dots < s_n \leq 1$. Moving along the curve, this sequence of planar vectors defining the airfoil shape results in the matrix $X = (\mathbf{x}_1, \dots, \mathbf{x}_n)^T \in \mathbb{R}_*^{n \times 2}$, where $\mathbb{R}_*^{n \times 2}$ refers to the set of full-rank $n \times 2$ matrices (i.e., the non-compact Stiefel manifold). The full-rank restriction ensures that we do not consider degenerate X as a feasible discrete representation of an airfoil shape.

Pivotal in this analysis is that *we do not require a choice of basis expansion as a representation to deform c*. Instead, we assume we have access to a database of discrete shapes generated by a diverse set of potentially different representations defining a variety of airfoil shapes. Then, we work with statistics of discrete shapes as opposed to statistics associated with coefficients in an expansion.

The innovative characteristic of the proposed approach is representing discrete airfoil shapes as elements of a Grassmann manifold (or Grassmannian) $\mathcal{G}(n, 2)$ paired with a corresponding affine transformation defined by an invertible 2×2 matrix plus a translation. This definition of the airfoil shape gives rise to a separable representation, making important subsets of deformations independent and allowing designers to make interpretable, systematic changes to airfoil shapes over either type of deformation. For example, one may seek to preserve average airfoil scale characteristics while independently studying all remaining undulating deformations as perturbations over the Grassmannian. Further, this separability enables the extension of 2D shape parametrizations to 3D blade and wing shapes in a low-dimensional and consistent manner.

2.1. Affine deformations

We discuss affine deformations describing large-scale planar deformations to shapes—namely, the linear term. We present linear deformations that are known to vary aerodynamic performance significantly and/or require specific control informed by design constraints. This motivates the need to separate linear deformations in shape representations from higher order oscillations or undulations in the shapes.

Affine deformations (i.e., scale, rotation, shear, and translation) of an airfoil have the form $M^T \mathbf{c}(s) + \mathbf{b}$, where $M \in GL_2$ is an element from the set of all invertible 2×2 matrices (for brevity, we simply refer to $GL_2(\mathbb{R})$ as GL_2 given all data and computation is over the reals) and $\mathbf{b} \in \mathbb{R}^2$. Note that deformations by rank-deficient M would collapse the shape to a line or to a single point and are not considered physically relevant as they have zero area.

A shape represented by associated boundary $\mathbf{c} : \mathcal{I} \rightarrow \mathbb{R}^2$ can be expressed generally as $\mathbf{c}(s) = (c_1(s), c_2(s))^T$ such that the univariate functions c_i for $i = 1, 2$ admit basis expansions $c_i(s) = \sum_{k \in \mathcal{K}} a_{ki} B_{ik}(s)$ as B-splines of fixed order, Bézier curves, or otherwise. Setting

$(\Phi(s))_{ik} = (B_{ik}(s))$ and $(A)_{ki} = a_{ki}$, we can write this through the lens of linear algebra as $\mathbf{c}(s) = \sum_{j=1}^2 \mathbf{e}_j (\mathbf{e}_j^\top \Phi(s) A \mathbf{e}_j)$ where \mathbf{e}_j is the j th column of the 2-by-2 identity matrix, A is a $|\mathcal{K}|$ -by-2 matrix of the coefficients parametrizing the curve, and $\Phi(s)$ a 2-by- $|\mathcal{K}|$ matrix of evaluated basis functions. By varying A , we can deform the shape as $D(A)$ through some mapping $D(\cdot)$ returning a matrix of appropriate dimension and rank. However, this mapping is unknown in general and could vary dramatically from one representation A to the next. Applying M to the curve as a *scaling/rotation/shearing* deformation then modifies $D(A)$ and vice versa.

As an example, a common choice is to take $c_1(s) = s$ as the identity with all other basis functions in Φ set to 0 along this $i = 1$ horizontal coordinate. Consequently, the vertical direction is parametrized along the horizontal planar-coordinate axis by s . Any deformation $D(A)$ to coefficients of the expansion deform the curve as $\hat{\mathbf{c}}(s) = \sum_{j=1}^2 \mathbf{e}_j (\mathbf{e}_j^\top \Phi(s) D(A) \mathbf{e}_j)$ in the planar-vertical direction which is equivalent to the *graph* of the deformed function \hat{c}_2 over a potentially scaled domain—i.e., $(\hat{a}_{11}s, \sum_k \hat{a}_{k2} B_{k2}(s))$ for $\hat{a}_{11} > 0$ avoiding reflection and maintaining the rank of \hat{A} provided at least one $\hat{a}_{k2} \neq 0$ for $k = 2, \dots, |\mathcal{K}|$. Any deformation of the curve—parametrized in any manner as D —is then realized as a new expansion, $\hat{A} = D(A)$. However, applying a linear scaling M modifies the expansion and vice versa thus coupling deformation types as

$$\begin{aligned} \hat{\mathbf{c}}^\top(s) &= \left(\hat{a}_{11}s, \sum_k \hat{a}_{k2} B_{k2}(s) \right) M \\ &= \left(M_{11} \hat{a}_{11}s + M_{21} \sum_k \hat{a}_{k2} B_{k2}(s), M_{12} \hat{a}_{11}s + M_{22} \sum_k \hat{a}_{k2} B_{k2}(s) \right), \end{aligned}$$

where M_{ij} are the corresponding entries of M . Even in the special case that off-diagonal elements of M are zero (no shearing) and $M_{11} = 1$, M_{22} must be non-zero to maintain $M \in GL_2$ but M_{22} is modified by factoring any non-zero common denominator from the coefficients in the expansion of \hat{c}_2 —i.e., generally, $D(A)$ represents the effect of a vertical scaling by a common factor. Thus, changing \hat{A} changes scaling M and changing M changes our expansion \hat{A} despite the systematic choice of expansion along the horizontal planar-coordinate—which is coupled to M_{22} at a minimum. Of course, this is further complicated if M constitutes a rotation or shearing. In other words, notice that the specific form of Φ does not decouple linear deformations M applied to the shape represented by deforming A via unknown $D(\cdot)$. However, specific choices of $D(\cdot)$ may accomplish scale-invariance subject to corresponding constraints on any given representation—e.g., restricting a function space to a sphere (Hagwood et al., 2013).

As opposed to working with specific choices of Φ and constrained $D(\cdot)$, we opt to work with discrete shapes X without *designing expansion representations*. Additionally, we have no prudent notion of distance in a high-dimensional space containing A and \hat{A} which complicates selection of domain definitions when a Euclidean distance is presumed insufficient.

For a discrete shape representation, affine deformations can be written as the smooth right action with translation $XM + 1_{n,2} \text{diag}(\mathbf{b})$, where $1_{n,2}$ denotes the $n \times 2$ matrix of ones and $\text{diag}(\mathbf{b}) = \sum_{j=1}^2 (\mathbf{e}_j^\top \mathbf{b}) \mathbf{e}_j \mathbf{e}_j^\top$. Note that the translation of the shape \mathbf{b} does not change the intrinsic characteristics of the shape (i.e., it has no deforming effect) and is generally of little interest for 2D design problems. For 3D blade design, \mathbf{b} locates the landmark airfoils relative to one another and can define the center of rotation.

Previous work on sensitivity analysis of CST parameters representing airfoil shapes has revealed certain shape deformations can dramatically change the coefficients of lift and drag (Glaws

et al., 2022b; Grey & Constantine, 2018). These deformations are similar to affine deformations of simultaneously changing camber and thickness—a result consistent with laminar flow theory. This dominating influence of affine deformations on aerodynamic quantities of interest inhibits the nuanced study of a richer set of perturbations to airfoil shapes, which is becoming increasingly important to continued progress in aerodynamics research. For example, the set of “dents” and “dings” common to damage and manufacturing defects—e.g., leading edge erosion and soiling of an airfoil shape—cannot be described entirely by affine deformations. However, a fundamental understanding of the impact of these features on aerodynamic performance can lead to increased longevity of expensive and difficult-to-replace components, such as offshore wind turbine blades. This motivates the need for a set of parameters that describe deformations independent of those in this dominating class of affine transformations. More precisely, we seek transformations to separately treat smooth right actions over GL_2 . This line of research was initially proposed as an extension of Grey and Constantine (2018) in Grey (2019).

Affine deformations constitute only a subset of the possible important aerodynamic deformations. We contend that aerodynamics will be significantly influenced by any parametrization, composition, or generalization of scaling/rotation as long as $M \in GL_2$. Moreover, these affine deformations are critical for 3D design and are usually constrained or rigorously chosen when selecting nominal definitions of shapes. For example, a useful example parametrization of the linear term is

$$\begin{aligned} L_4 : \mathcal{L} \subset \mathbb{R}^4 &\rightarrow GL_2 \\ \ell &\mapsto \ell_1 \begin{bmatrix} \ell_2 & 0 \\ 0 & \ell_3 \end{bmatrix} \begin{bmatrix} \cos(\ell_4) & \sin(\ell_4) \\ -\sin(\ell_4) & \cos(\ell_4) \end{bmatrix}. \end{aligned} \quad (1)$$

This parametrization is representative of the types of systematic deformations chosen or constrained in blade design. That is, we compute volumetric (ℓ_1) and coordinate-aligned horizontal (ℓ_2) and vertical (ℓ_3) scalings, then rotate the shape into the final angle of attack (ℓ_4) for assembly or modeling. Although (1) is not necessarily a common parametrization of GL_2 , assuming all of GL_2 is aerodynamically significant offers more flexibility for designers to select or fix arbitrary deformations over GL_2 beyond those parametrized by (1) that may be deemed interesting.

We seek to decouple and preserve affine features for blade and wing design through a set of inferred shape deformations over the Grassmannian that are independent of GL_2 . We also discuss separable shape tensors for parametrizing scale variations independent of rotation/reflection for individual airfoil (2D) design.

2.2. Separable shape tensors

We introduce the Grassmannian $\mathcal{G}(n, q)$ as a topology where variations in discrete shapes due to linear deformations are “divided out.” We describe how to map shapes to representative elements of the Grassmannian as landmark-affine (LA) standardizations using: (i) the singular value decomposition (SVD) and (ii) the related polar decomposition. This motivates parametrizations of sections through product manifolds: (i) $\mathcal{G}(n, 2) \times GL_2$ and (ii) $\mathcal{G}(n, 2) \times S_{++}^+$, respectively. These will define the “parent” topologies for submanifolds of separable shape tensors.

2.2.1. LA standardizations

Given a discrete shape representation X and an important affine deformation $XM + 1_{n,2} \text{diag}(\mathbf{b})$, we now develop the necessary

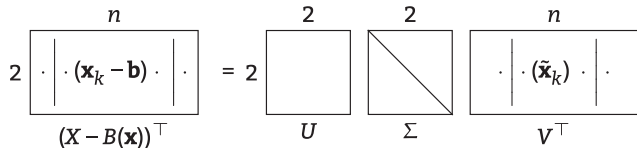


Figure 1: Thin SVD of centered landmarks with appropriate dimensions and corresponding notation. Lines motivate an intuition for a relevant partitioning of any given matrix.

interpretations for deforming shapes independent of the often constrained and notably aerodynamically sensitive affine features. Through this development, we reveal underlying parent matrix manifold topologies which will inform improved non-Euclidean considerations for subsequent computations and 3D blade interpolation.

The Grassmannian $\mathcal{G}(n, q)$ is the space of all q -dimensional subspaces of \mathbb{R}^n . Formally, $\mathcal{G}(n, q) \cong \mathbb{R}_*^{n \times q} / GL_q$, meaning that elements of the Grassmannian are invariant under GL_q transformations where GL_q is the set of all invertible $q \times q$ matrices. Given this invariance, we may consider an element of the Grassmannian $[\tilde{X}] \in \mathcal{G}(n, q)$ to be the equivalence class of all matrices with the same column span as the representative element $\tilde{X} \in \mathbb{R}_*^{n \times q}$ (Absil et al., 2008). That is, the equivalence class $[A] = \{B \in \mathbb{R}_*^{n \times q} : B \sim A\}$ is defined by equivalence relation $A \sim B$ such that $\text{Range}(A) = \text{Range}(B)$. In this way, every element of the Grassmannian is a full-rank matrix modulo GL_q deformations. Thus, deformations over $\mathcal{G}(n, q)$ are independent of affine deformations (ignoring the non-deforming translations)—i.e., $[XM] = [X]$. By representing discretized airfoil shapes as elements of the Grassmannian, we ensure that deformations to shapes or differences between shapes in this space are, by definition, decoupled from the aerodynamically important affine deformations—e.g., linear transformations varying camber, thickness, and length.

It is common to view the Grassmannian as a quotient topology of orthogonal subgroups such that the n landmarks of any representative element \tilde{X} have sample covariance proportional to the identity matrix—i.e., $\tilde{X}^T \tilde{X} = I_q$ (Edelman et al., 1998; Gallivan et al., 2003). In practice, this means that a representative computational element of the Grassmannian is an $n \times q$ matrix with orthonormal columns (Edelman et al., 1998). This perspective offers certain computational advantages and motivates a scaling of airfoil landmark data for computations over $\mathcal{G}(n, 2)$ for airfoil design. In our case, n is equal to the number of landmarks, and $q = 2$ is the dimension of the ambient space where the shape lives for design.

To represent physical airfoil shapes as elements of the Grassmannian, we define the LA standardization (Bryner et al., 2014) as a mapping $\pi : \mathbb{R}_*^{n \times 2} \rightarrow \mathcal{G}(n, 2)$. LA standardization normalizes the shape to have zero sample mean (translation invariance) and identity sample covariance (scale invariance) over the n landmarks defining the shape. The remainder of this section discusses computation of the LA standardization and examines its properties.

Given a discrete airfoil shape with landmarks $X \in \mathbb{R}_*^{n \times 2}$, let $\mathbf{b}(X) = (1/n)X^T \mathbf{1}_{n,1}$ be the discrete center of mass and compute the thin SVD of the centered airfoil $(X - B(X))^T = U\Sigma V^T$, where $B(X) = \mathbf{1}_{n,2} \text{diag}(\mathbf{b}(X))$. Then, define \tilde{M} to be the 2×2 invertible matrix

$$\tilde{M} = \Sigma U^T. \quad (2)$$

The mapping between the airfoil X and its LA-standardized representation \tilde{X} is

$$X - B(X) = \tilde{X}\tilde{M}. \quad (3)$$

As a result, this definition of \tilde{X} provides scale standardization

$$\begin{aligned} \tilde{X}^T \tilde{X} &= \tilde{M}^{-T} (X - B(X))^T (X - B(X)) \tilde{M}^{-1} \\ &= \tilde{M}^{-T} (U\Sigma V^T) (V\Sigma U^T) \tilde{M}^{-1} \\ &= \Sigma^{-1} U^T (U\Sigma^2 U^T) U \Sigma^{-1} \\ &= (\Sigma^{-1} \Sigma) (\Sigma \Sigma^{-1}) \\ &= I_2, \end{aligned}$$

consistent with a “whitening transform” (Hyvärinen & Oja, 2000). From (3), we have $\tilde{X} = V$, with standardized landmarks along the rows as $\tilde{X} = (\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_n)^T \in \mathbb{R}_*^{n \times 2}$. To clarify, Fig. 1 depicts the dimensionality of the various matrices and the notation. The LA standardization (3) satisfies assumptions to apply various intrinsic parametrizations for computing normal coordinates over the Grassmannian (Edelman et al., 1998).

For $[\tilde{X}] \in \mathcal{G}(n, 2)$, \tilde{X} is a *representative* (Stiefel) element of the Grassmannian, defined uniquely up to any GL_2 transformation (Absil et al., 2008; Edelman et al., 1998). Abstractly, we map a given discrete airfoil shape X to an equivalence class $[\tilde{X}]$ via $\pi : \mathbb{R}_*^{n \times 2} \rightarrow \mathcal{G}(n, 2) : X \mapsto [\tilde{X}]$ such that

$$\pi(X) = [\tilde{X}] = [X - B(X)]. \quad (4)$$

Next, we show that $\pi(X) = [\tilde{X}]$ is surjective, thus admitting a parametrizable *right inverse*, and satisfies the desired scale and translation invariance properties. Lastly, we show that π is the canonical projection thus the right inverse parametrizes sections through $\mathbb{R}_*^{n \times 2}$ as a submanifold. The results establish a principled framework for “learning” a (sub)manifold of discrete shapes in $\mathbb{R}_*^{n \times 2}$ with the desired separability.

Proposition 1. π is surjective.

Proof. For any $[X] \in \mathcal{G}(n, 2)$, we can take an arbitrary basis $[X] = \text{span}\{\mathbf{v}_1, \mathbf{v}_2\}$ for linearly independent $\mathbf{v}_1, \mathbf{v}_2 \in \mathbb{R}^n$. Consequently, taking arbitrary $(a\mathbf{v}_1, b\mathbf{v}_2) \in \mathbb{R}_*^{n \times 2}$ for $a, b \neq 0$ implies $\pi((a\mathbf{v}_1, b\mathbf{v}_2)) = [X]$.

Proposition 2. π is scale invariant such that $\pi(XM) = \pi(X)$ for any $M \in GL_2$.

Proof. Defining $B(X) = \mathbf{1}_{n,2} \text{diag}(\mathbf{b}(X))$ where $\mathbf{b}(X) = (1/n)X^T \mathbf{1}_{n,1}$,

$$\begin{aligned} \pi(XM) &= [XM - B(XM)] \\ &= [XM - \mathbf{1}_{n,2} \text{diag}((1/n)(XM)^T \mathbf{1}_{n,1})] \\ &= [XM - \mathbf{1}_{n,2} \text{diag}(M^T \mathbf{b}(X))] \\ &= [XM - \mathbf{1}_{n,2} \text{diag}(\mathbf{b}(X)^T M)] \\ &= [XM - (\mathbf{1}_{n,2} \text{diag}(\mathbf{b}(X)))M] \\ &= [(X - B(X))M] \\ &= [X - B(X)] \\ &= \pi(X). \end{aligned}$$

Proposition 3. π is translation invariant such that $\pi(X + \mathbf{1}_{n,2} \text{diag}(\mathbf{v})) = \pi(X)$ for any $\mathbf{v} \in \mathbb{R}^2$.

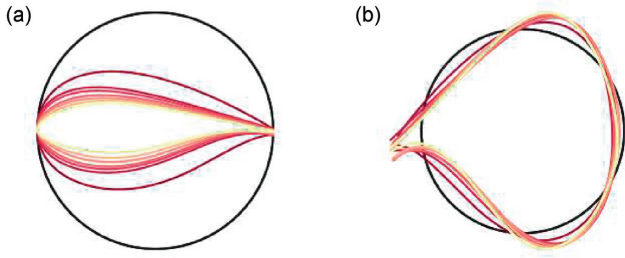


Figure 2: Collection of 10 nominal cross-sectional airfoils defining the International Energy Agency (IEA) 15-MW blade in (a) physical coordinates, X , and (b) LA standardized coordinates, \tilde{X} .

Proof. For arbitrary $\mathbf{v} \in \mathbb{R}^2$,

$$\pi(X + 1_{n,2}\text{diag}(\mathbf{v})) = [X + 1_{n,2}\text{diag}(\mathbf{v}) - B(X + 1_{n,2}\text{diag}(\mathbf{v}))].$$

Then, we recognize that

$$\begin{aligned} B(X + 1_{n,2}\text{diag}(\mathbf{v})) &= 1_{n,2}\text{diag}((1/n)(X + 1_{n,2}\text{diag}(\mathbf{v}))^T 1_{n,1}) \\ &= 1_{n,2}\text{diag}((1/n)X^T 1_{n,1} + (1/n)\text{diag}(\mathbf{v}) 1_{n,2}^T 1_{n,1}) \\ &= 1_{n,2}\text{diag}((1/n)X^T 1_{n,1} + \text{diag}(\mathbf{v}) 1_{2,1}) \\ &= 1_{n,2}\text{diag}((1/n)X^T 1_{n,1}) + 1_{n,2}\text{diag}(\text{diag}(\mathbf{v}) 1_{2,1}) \\ &= B(X) + 1_{n,2}\text{diag}(\mathbf{v}). \end{aligned}$$

Noting that $1_{n,2}^T 1_{n,1} = n 1_{2,1}$, $\text{diag}(\mathbf{v}) 1_{2,1} = \mathbf{v}$, and $\text{diag}(\mathbf{u} + \mathbf{v}) = \text{diag}(\mathbf{u}) + \text{diag}(\mathbf{v})$. Plugging this result into the equation above yields

$$\begin{aligned} \pi(X + 1_{n,2}\text{diag}(\mathbf{v})) &= [X + 1_{n,2}\text{diag}(\mathbf{v}) - B(X + 1_{n,2}\text{diag}(\mathbf{v}))] \\ &= [X + 1_{n,2}\text{diag}(\mathbf{v}) - B(X) - 1_{n,2}\text{diag}(\mathbf{v})] \\ &= [X - B(X)] \\ &= \pi(X). \end{aligned}$$

Proposition 4. π is the canonical projection.

Proof. It is sufficient to show that π is idempotent onto equivalence classes, $\pi(\pi(X)) = \pi(X)$. For $\pi(X) = [\tilde{X}]$ and $(X - B(X))^T = U\Sigma V^T$, representative $\tilde{X} = (X - B(X))U\Sigma^{-1}$ has zero mean over rows, $B(\tilde{X}) = \mathbf{0}$, and $\tilde{X}^T = \Sigma^{-1}U^T U\Sigma V^T = V^T$, which at most rotates and/or reflects the shape after LA standardization informed by the thin SVD $V^T = \tilde{U}I_2\tilde{V}^T$. Consequently, $\pi(\pi(X)) = [\tilde{X}\tilde{U}] = [\tilde{X}] = \pi(X)$.

Proposition 2 motivates an alternative interpretation of π as a GL_2 scale invariance, $\pi(XM) = [\tilde{X}]$. Intuitively, $\pi(X)$ “standardizes” the discretized shape X such that \tilde{X} is as circular as possible. Figure 2 depicts a set of example transformations between these two discrete representations for a collection of wind turbine airfoil shapes. Proposition 2 together with Proposition 3 asserts the sought affine invariance properties of a non-linear mapping (the thin SVD from one discrete shape to the next is computed by an iterative procedure which is, in general, non-linear over a space of changing matrices) onto the Grassmannian, per Proposition 1. With Propositions 1 and 4, we can propose the parametrization of a section through $\mathbb{R}_*^{n \times 2}$ using representative elements \tilde{X} to build a submanifold. Consequently, we can define a separable shape ten-

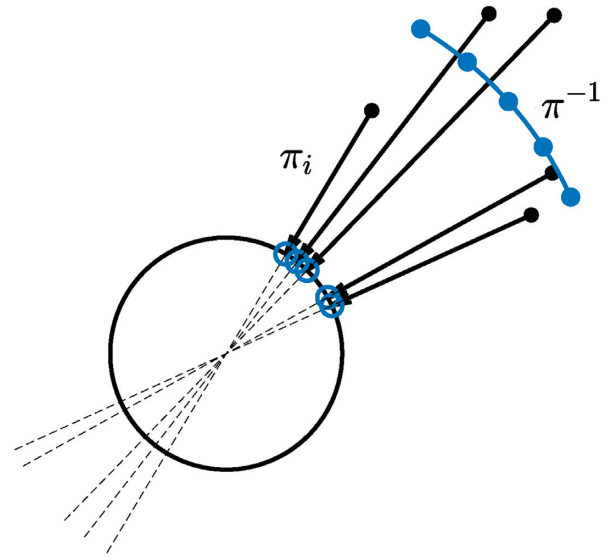


Figure 3: A simplified visualization of five individual projections π_i (black arrows) from elements in $\mathbb{R}_*^{2 \times 1}$ (black dots) onto representative elements of the upper semicircle (blue circles). Elements of $\mathcal{G}(2, 1)$ are shown as dashed lines. A constant section of the fiber bundle (a submanifold of $\mathbb{R}_*^{2 \times 1}$) is shown as the blue curve, with five (uniformly sampled) elements as coincident dots.

or parametrization for discrete shapes,

$$X(\mathbf{t}, \boldsymbol{\ell}) = (\pi^{-1} \circ [\tilde{X}])(\mathbf{t}, \boldsymbol{\ell}) = \tilde{X}(\mathbf{t})M(\boldsymbol{\ell}). \quad (5)$$

In this formalism, π^{-1} is the right inverse of π parametrized by $(\mathbf{t}, \boldsymbol{\ell}) \in \mathcal{T} \times \mathcal{L} \subseteq \mathbb{R}^{r+4}$. This parameter domain will be inferred from data-driven pairs $\{[\tilde{X}_k], M_k\} \subset \mathcal{G}(n, 2) \times GL_2$ given by the thin SVD of discrete shapes, $\{X_k\}$. In practice, $M(\boldsymbol{\ell})$ could be expressed for design as the composition of a fixed nominal scaling with a parametrized affine deformation—e.g., $M(\boldsymbol{\ell}) = \bar{M}L_4(\boldsymbol{\ell})$ possibly with translations where $L_4(\boldsymbol{\ell})$ is defined in (1) and \bar{M} is some fixed nominal scaling like an average. Figure 3 shows a simplified visual analogue of this approach for a constant average scale factor \bar{M} —we further elaborate on the ability to average over GL_2 via separability in the next section. The parametrization of the Grassmannian element $[\tilde{X}](\mathbf{t})$ is inferred from data-driven methods that are also discussed in later sections. We note that the dimension of \mathbf{t} is restricted by the intrinsic dimensionality of $\mathcal{G}(n, 2)$ such that $r \leq 2(n - 2)$ but is practically chosen to be much smaller.

The utility of the representation in (5) is the *separable form* of the airfoil representation such that the effects of changing \mathbf{t} are separate from the effects of changing $\boldsymbol{\ell}$. The lingering question is: Given a database of discrete airfoils $\{X_k\}$, how can we infer parameter distributions of $(\mathbf{t}, \boldsymbol{\ell})$? Alternatively, how should we define $\mathcal{T} \times \mathcal{L}$ for subsequent design tasks? The key will be utilizing data-driven approaches involving the underlying Riemannian geometry—described in the sections to follow.

2.2.2. Mean scales of random airfoils

When considering an “ensemble” of airfoil shapes $\{X_k\}$, an average notion of scale can be used to remove the dependencies on $M(\boldsymbol{\ell})$. For example, we could define the constant extrinsic estimate $\bar{M} = (1/N) \sum_{k=1}^N \tilde{M}_k$, where \tilde{M}_k is computed as in (2) for the corresponding X_k . Assuming these airfoils implicitly define some distribution over \mathcal{L} , this offers a notion of “average scale” for parametrizing a “local section of the fiber bundle” through “total space” $\mathbb{R}_*^{n \times 2}$ as $(\pi^{-1} \circ [\tilde{X}])(\mathbf{t}; \bar{M})$ (Grey, 2019). Figure 3 depicts a

simplified visual analogue for this choice of constant scaling—represented by the blue curve.

When designed or inferred affine deformation subgroups, or arbitrary parametrizations such as (1), are combined with translations, they can then be applied independently to $\tilde{X}(\mathbf{t})\bar{M}$ as a systematic design schema. As an aerodynamic interpretation, for unknown domain \mathcal{L} weighted by unknown probability measure ρ , order-dependent compositions of camber, chord, twist, and/or thickness deformations can be independently applied to Monte Carlo approximations of average-scale shapes, $\tilde{X}(\mathbf{t})\bar{M} \approx \int_{\mathcal{L}} \tilde{X}(\mathbf{t})M(\boldsymbol{\ell})d\rho(\boldsymbol{\ell}) = \tilde{X}(\mathbf{t}) \int_{\mathcal{L}} M(\boldsymbol{\ell})d\rho(\boldsymbol{\ell})$. The challenge is ensuring that $\bar{M} \in GL_2$ and that \bar{M} does not arbitrarily inflate average length scales of the shape—i.e., \bar{M} can result in an inflated determinant. This may require control through additional nonlinear transformations (e.g., a set of shape constraints) applied to $\tilde{X}(\mathbf{t})\bar{M}$ by the shape designer. Alternatively, we could pose an intrinsic mean scale over GL_2 . In either case, the ability to average scales $\{\bar{M}_k\}$ —or separately compute higher order moments of important and highly sensitive scale variations over alternative metric spaces—is enabled by the separability in (5).

2.2.3. Equivalent polar decomposition

We next develop an alternative method for mapping discrete airfoil shapes to the Grassmannian based on a rotation-invariant polar decomposition. We denote the resulting product manifold of shapes as $\mathcal{G}(n, 2) \times S_{++}^2$, where S_{++}^2 denotes the set of 2×2 symmetric positive definite (SPD) matrices. First, we define an equivalence relationship $\tilde{X} \sim_{\mathcal{O}} \tilde{X}O$ for all orthogonal 2×2 matrices $O \in \mathcal{O}(2)$ such that $[X]_{\mathcal{O}} = \{A \in \mathbb{R}_*^{n \times 2} : A \sim_{\mathcal{O}} X\}$. Taking the polar decomposition of the linear transformation as $M = P_2R_2$, we have $[\tilde{X}M]_{\mathcal{O}} = [\tilde{X}P_2R_2]_{\mathcal{O}} = [\tilde{X}P_2]_{\mathcal{O}}$. Thus, we can parametrize a set of rotation/reflection-invariant shapes as $\tilde{X}(\mathbf{t})P(\boldsymbol{\ell})$ for all $P(\boldsymbol{\ell}) \in S_{++}^2$. That is, we retain the scale variations of shapes over SPD matrices and “divide out” deformations resulting in rotations and reflections.

Given a discrete shape X and its corresponding thin SVD $(X - B(X))^T = U\Sigma V^T$, the polar decomposition becomes $(X - B(X))^T = PR$ such that $P = U\Sigma U^T$ is unique and $R = UV^T$. Given V as an $n \times 2$ rectangular matrix with orthonormal columns and U orthogonal implies that $RR^T = I_2$. Moreover, $P^{-1} = U\Sigma^{-1}U^T$ is SPD. Hence, $\tilde{X} = (X - B(X))P^{-1}$ defines an equivalent normalization of scale,

$$\begin{aligned} \tilde{X}^T \tilde{X} &= P^{-1}(X - B(X))^T (X - B(X))P^{-1} \\ &= P^{-1}(PR)(R^T P)P^{-1} \\ &= (P^{-1}P)(PP^{-1}) \\ &= I_2. \end{aligned}$$

Equivalently, $\tilde{X} = R^T = VU^T$, which rotates/reflects the original non-unique LA-standardized shape V back into the original view by U^T .

For an ensemble of shapes $\{X_k\}$, we can compute the corresponding P_k from the approximated thin SVD and use the data-driven pairs to construct a submanifold from $\{[\tilde{X}_k], P_k\} \subset \mathcal{G}(n, 2) \times S_{++}^2$ —i.e., a set of rotation/reflection-invariant subgroups of discrete shapes. The separable form of rotation/reflection-invariant physical airfoils becomes

$$X(\mathbf{t}, \boldsymbol{\ell}) = (\pi^{-1} \circ [\tilde{X}]) (\mathbf{t}, \boldsymbol{\ell}) = \tilde{X}(\mathbf{t})P(\boldsymbol{\ell}) \quad (6)$$

for parameters $(\mathbf{t}, \boldsymbol{\ell}) \in \mathcal{T} \times \mathcal{P} \subset \mathbb{R}^{4+3}$. Notice also that $P(\boldsymbol{\ell})$ serves to parametrize the eigenspaces of the landmark sample covariance, i.e., $(X - B(X))^T (X - B(X)) = U\Sigma U^T = P$. In other words,

parametrizing linear scale variations in the shape is equivalent to parametrizing the sample covariance of landmarks, establishing the utility of examining and parametrizing the range of U .

2.2.4. Convergence to discrete representatives

The interpretation and use of the Grassmannian for the purpose of describing a topology of discrete shapes is a relatively unique application of the manifold. The motivation stems from the application-driven need for separable representations to study or systematically control distinct affine deformations of known physical importance. There are concerns about how this treatment of discrete shapes may change when subject to reparametrizations as diffeomorphisms, i.e., $\xi : \mathcal{I} \rightarrow \mathcal{I}$ defining new landmarks $\mathbf{x}_i = \mathbf{c}(\xi(s_i))$ that constitute the transposed rows of X . Modern definitions of shape spaces (Michor & Mumford, 2006; Welker, 2021) are typically defined modulo such diffeomorphisms, and alternative frameworks (Dogan et al., 2015; Joshi et al., 2007; Klassen et al., 2004) take advantage of a “pre-shape space” with “square root velocity functions”, inducing a notion of distance between shapes.

In shape design, ξ is often defined in an effort to best identify sequences (\mathbf{x}_i) that are distributed along the arc-length of the shape with increased concentration around regions of high curvature or are distributed uniformly with high resolution (e.g., $n \geq 1000$ in airfoil design) to improve meshing in a flow solver. We propose fixing ξ such that it induces a specific distribution over chosen parameter s to generate corresponding landmark refinements, $X \in \mathbb{R}_*^{n \times 2}$.

Data sets of discrete shapes rarely share a common choice of generating landmark distribution, particularly if the shapes are gathered from multiple sources. Further, the number of landmarks in the discrete shapes n_c may vary across the data set. We consider a simple “preprocessing” of data before working with tensor representations. This preprocessing proceeds as follows. Given landmark data (\mathbf{x}_i) , we first compute normalized cumulative Euclidean lengths:

$$s_i = \begin{cases} 0, & i = 1 \\ \sum_{k=1}^{i-1} \|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2 / \sum_{k=1}^{n_c-1} \|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2, & i = 2, \dots, n_c. \end{cases} \quad (7)$$

We then interpolate the entries of (\mathbf{x}_i) over $\xi(s_i)$ to construct a continuous representation $\hat{\mathbf{c}}(s)$ as an approximation. Finally, we generate a fixed “ n -refinement” $X \in \mathbb{R}_*^{n \times 2}$ as n (typically greater than n_c) landmarks generated by interpolations and LA standardize the n -refinement to produce $\tilde{X} \in \mathbb{R}_*^{n \times 2}$. This procedure also offers control over the “landmark gauge”, $\bar{h}_n = \max_{k \in \{1, \dots, n-1\}} \{\|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2\}$, of the n -refinement for subsequent meshing and simulation. This preprocessing motivates the following result:

Lemma 1. Given sparse data $(\mathbf{x}_i)_{i=1}^{n_c}$ from assumed $\mathbf{c} \in C^4(\mathcal{I})$, n -refinements $X \in \mathbb{R}_*^{n \times 2}$ generated by cubic splines of $(\mathbf{x}_i)_{i=1}^{n_c}$ with fixed reparametrization ξ converge to elements $[\tilde{X}] \in \mathcal{G}(n, 2)$ as $O(h_{n_c}^4)$ where $h_{n_c} = \max_{k \in \{1, \dots, n_c-1\}} |s_{k+1} - s_k|$ is the gauge of an arc-length parametrization.

Proof. By the polar decomposition $(X - B(X))^T = PR$, P is unique and thus $\tilde{X} = R^T = (X - B(X))P^{-1}$ is unique, provided X is full rank, which is true by assumption. Consequently, $[\tilde{X}] = [R^T]$ is unique.

In the most general case, an interpolation $\bar{\mathbf{c}}$ must be evaluated over an appropriately weighted domain to maintain a correspondence of landmarks consistent with a fixed reparametrization $\xi : \mathcal{I} \rightarrow \mathcal{I}$ over arc-lengths $s \in \mathcal{I}$. Otherwise, inconsistent land-

marks are encoded in the rows of the n -refinement. We argue convergence over the worst-case error across rows of X computed with $\hat{\mathbf{c}} = \hat{\mathbf{c}} \circ \xi$, necessitating a choice and/or construction of fixed ξ with pre-image as arc-lengths $s \in \mathcal{I}$. We define

$$\|\mathbf{e}\|_{\infty,2} = \max_s \{\|\mathbf{e}(s)\|_2 : s \in \mathcal{I}\}$$

as the “worst-case” Euclidean norm over any row encoded in a corresponding matrix. For fixed ξ and h_{n_c} as the gauge of the n_c -mesh, it follows that the worst-case row converges for decreasing h_{n_c} by applying results from Birkhoff and De Boor (1964; sharpened by Hall & Meyer, 1976), asserting

$$\|\hat{\mathbf{c}} - \mathbf{c}\|_{\infty,2} \leq K_4 h_{n_c}^4$$

which bounds an equivalent matrix norm applied to a corresponding difference in n -refinements. The aforementioned result derives constant K_4 which depends on the largest fourth derivatives of the component functions of \mathbf{c} .

We experiment with the common CST representation of airfoils to generate 100 random airfoil shapes according to a cosine domain distribution (Chebyshev nodes), resulting in a non-uniform arc-length distribution of landmarks. The cosine distributed landmarks are a common choice for airfoil designers because landmarks concentrate around the “leading edge” and “trailing edge” features of an airfoil; in this case, it reflects an intuitive choice of non-uniform landmark distribution. The CST representation utilizes 18 total coefficients (nine upper surface coefficients and nine lower surface coefficients), each uniformly sampled over $[0,0.45]$. However, the CST shapes are represented by a partition into upper and lower surface thus potentially compromising the assumed differentiability. Moreover, in general, we often do not have true arc-lengths along the shape as data. Instead, we supplement by using (7) as a parametrization of spline(s) with corresponding landmark gauge. We numerically study the effects of abusing the underlying assumptions of the theory to demonstrate empirically that our necessary preprocessing of shapes remains convergent.

Given a known CST shape and an appropriate (design-informed) sampling scheme, we compute a refinement at $n = 10000$, defining LA-standardized shape \tilde{X}^* with corresponding cumulative Euclidean lengths $(s_i^*)_{i=1}^n$ inferred from X^* (prior to LA standardization). Then, we sparsely resample the CST representation with the same sampling scheme for $n_c \leq n$ and provide the sparse landmarks to a cubic spline interpolation routine parametrized over sparse n_c using (7). The spline approximation is then used to generate LA standardized refinements \tilde{X} up to the corresponding $n = 10000$ according to $\xi(s_i^*)$ —e.g., ξ is the identity for the planar spline built from landmark entries over $(s_i)_{i=1}^{n_c}$, while alternative splines may utilize ξ as a piecewise cubic Hermite interpolating polynomial (PCHIP) such that $\hat{s}_i = \xi(s_i)$ correspond to (10) for the same n_c -mesh. The nature of the spline interpolation over cumulative Euclidean lengths as an approximation induces an error that manifests as a Euclidean shape distance, $\|X - X^*\|_{\infty,2} = \max_{k \in \{1, \dots, n\}} \{\|\mathbf{x}_k - \mathbf{x}_k^*\|_2\}$, or as the sum of principal angles between n -refinement $[\tilde{X}]$ and $[\tilde{X}^*]$ as a distance over the Grassmannian, $d_{G(n,2)}([\tilde{X}], [\tilde{X}^*])$. Evidence is depicted in Fig. 4.

Increasing n_c landmarks with reduced landmark gauge provided as input data improve accuracy in the separable tensor representations. The observed order of convergence—specifically for the chosen random CST airfoils as a particularly relevant class of shapes—to known representative elements on the Grassmannian is reduced from the result of the Lemma 1. However, we observe

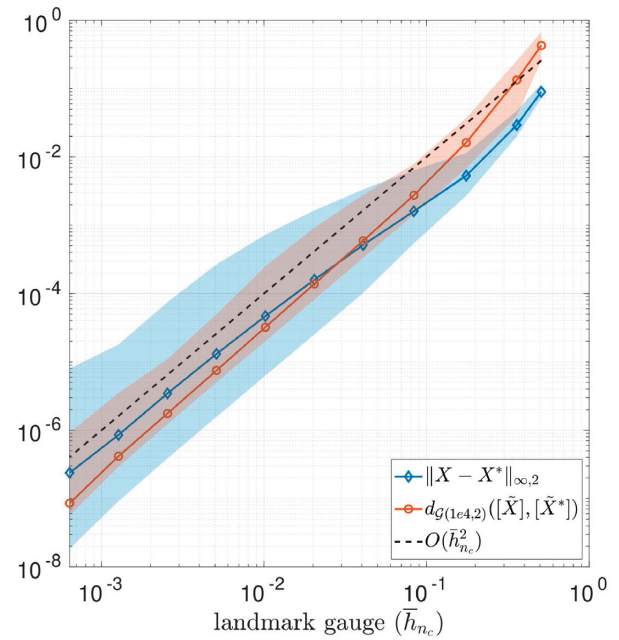


Figure 4: Convergence of $n = 10000$ refinements X and \tilde{X} , generated by planar splines reparametrized over (7), given n_c landmarks as data with corresponding landmark gauge h_{n_c} . Shaded regions represent a max-min envelope over all 100 random CST airfoils. Colored curves represent the average Euclidean shape distance and Grassmannian distance (sum of principal angles) between interpolated refinements and true refinements over all random shapes. The dashed black curve depicts the observed quadratic convergence.

that the preprocessing procedure still offers a relatively fast order of convergence over a class of relevant shapes.

2.2.5. Shortcomings

The proposed separable representations (5) and (6) are not without drawbacks. In particular, physically relevant shapes \mathbf{c} are non-intersecting, with the possible exception of closed curves, which necessarily coincide at the endpoints, i.e., embeddings (Welker, 2021). Our current formalism could violate this. Specifically, the discrete separable representations $(\pi^{-1} \circ [\tilde{X}])(\mathbf{t}, \cdot)$ can generate self-intersections in continuous reconstructions for large changes in parameters \mathbf{t} . We mitigate these concerns by preventing extrapolation beyond embeddings (implicit to the data) using a numerical routine to check for a piecewise linear intersection condition (accurate for sufficiently large n). We hypothesize that constraining against significant extrapolation, with an improved notion of distance, beyond a set of discrete non-intersecting shapes as data is sufficient to protect against generating self-intersections—this is supplemented by empirical evidence. Future work will focus on improved constraints or continuous analogues avoiding self-intersection to explicitly constrain to a set of embedded curves.

2.3. Continuous analogues and comparisons

We unpack an interpretation of the LA standardized shapes as they relate to orthogonal polynomials. Thus extending discrete representations to continuous forms. We conclude with comparisons to alternative AI/ML frameworks for generative modeling of discrete representatives. These discussions set the stage for future work and possible numerical comparisons.

We explore building continuous analogues from the discrete shapes by approximating a so-called *quasimatrix*—i.e., a 2D array that is discrete along one dimension and continuous along the

other (Townsend & Trefethen, 2015). The basic idea is to explore a procedure for computing orthogonal functions given two discretizations (one per column) encoded in the rows of V . Specifically, we seek interpolations for the columns of the LA standardized shape $V = \tilde{X}$ (see proof of Proposition 4) such that

$$(X - B(X))(X - B(X))^T \mathbf{v}_j = \sigma_j^2 \mathbf{v}_j, \quad j = 1, 2, \quad (8)$$

where σ_1, σ_2 are the diagonal entries of Σ in (2). In this interpretation, (8) is a discrete analogue of the Fredholm integral equation

$$\int_{\mathcal{I}} k_{(c,c)}(s', s) v_j(s') d\mu(s') = \sigma_j^2 v_j(s) \quad (9)$$

for some measure μ .

We present (8) to motivate an interpretation of the columns of V as they relate to (9). In this interpretation, the columns of $V - \mathbf{v}_j$ for $j = 1, 2$ —represent discretizations of two $L^2(\mathcal{I})$ orthonormal eigenfunctions $v_j(s_i)$ for $s_i \in \mathcal{I}$ with ordering $s_1 < s_2 < \dots < s_n$. These eigenfunctions constitute coordinate functions for an LA-standardized planar curve $\tilde{\mathbf{c}}(s) = (v_1(s), v_2(s))$. The Mercer kernel $k_{(c,c)}$ is the Euclidean inner product of the centered planar curve with itself, akin to entries of a Gram matrix but distinct from landmark sample covariance. This choice is consistent with the “ $A = 0$ ” metric discussed in supporting work (Joshi et al., 2007; Michor & Mumford, 2006; Schulz, 2014). Although this choice of metric admits a pathology in the continuous framework (Michor & Mumford, 2006), it may still be worthwhile to consider in a discrete setting (Joshi et al., 2007; Schulz, 2014). This interpretation establishes the utility of examining and parametrizing the range of V . Additionally, this interpretation motivates how we may modify our implicit choice of kernel and/or shape metric for future applications.

Next, we inform continuous reconstructions using interpolation of discrete data while retaining the desired separability in deformations. To begin, we compute V via the SVD of centered X . Then, we consider the normalized cumulative Euclidean length along the discrete curve $V = \tilde{X} = (\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_n)^T$, over which we will construct our interpolation,

$$\tilde{s}_i = \begin{cases} 0, & i = 1 \\ \sum_{k=1}^{i-1} \|\tilde{\mathbf{x}}_{k+1} - \tilde{\mathbf{x}}_k\|_2 / \sum_{k=1}^{n-1} \|\tilde{\mathbf{x}}_{k+1} - \tilde{\mathbf{x}}_k\|_2, & i = 2, \dots, n. \end{cases} \quad (10)$$

This results in pairs $\{(\tilde{s}_i, v_{ij})\}$ for $j = 1, 2$, where v_{ij} is the ij entry of V . We can construct Barycentric Lagrange interpolation over these data pairs, defined by

$$v_j(\tilde{s}) = \sum_{i=1}^n \frac{w_i}{\tilde{s} - \tilde{s}_i} v_{ij} / \sum_{i=1}^n \frac{w_i}{\tilde{s} - \tilde{s}_i}, \quad (11)$$

where weights are given by $w_i = 1 / \prod_{k \neq i} (\tilde{s}_i - \tilde{s}_k)$ for all $i = 1, \dots, n$ (Berrut & Trefethen, 2004; Higham, 2004). Alternatively, we could employ piecewise interpolating splines with conditions designed for improved “fairness” (Sapidis, 1994) to build the two curves. Alternative interpolations include regularized cubic splines (clamped, natural, or periodic), PCHIP splines, B-splines, non-uniform rational basis splines, Hicks–Henne bump functions, or radial basis functions. The results, in any case, are two functions $v_1(\tilde{s})$ and $v_2(\tilde{s})$, that interpolate pairs $\{(\tilde{s}_i, v_{ij})\}$ for all $i = 1, \dots, n$ with $j = 1$ or $j = 2$, respectively. Concatenating $(v_1(\cdot), v_2(\cdot))$ into the columns of $V(\cdot)$, the interpolations defining $V(\cdot)$ are no longer necessarily orthogonal but retain some nice (often subjective, yet prescriptive) design characteristics. However, despite the choices defining the designed continuous reconstruction $V(\cdot)$, the two interpolations can be evaluated uniformly to in-

duce a fixed reparametrized integral measure $d\tilde{\mu}(\tilde{s}) = d\tilde{s}$ along the curve, which can then be projected onto a space of orthonormal Legendre polynomials (in this case, the arbitrary closed interval \mathcal{I} is reparametrized to $[-1, 1]$ in contrast to the choice of $[0, 1]$ in 10) via a QR-decomposition of the prescribed $\infty \times 2$ quasimatrix (Trefethen, 2010)—i.e., $V(\cdot) = Q(\cdot)R$ for Q an $\infty \times 2$ quasimatrix and R a 2×2 upper triangular. The quasimatrix $Q(\cdot)$ from the QR-decomposition becomes the continuous analogue that satisfies the orthonormal constraint of the representative discrete shape \tilde{X} , which is defined by a relevant choice of LA standardized landmark data interpolation, $V(\cdot)$.

If desired, the Legendre polynomials as columns of $Q(\cdot)$ —with sufficient differentiability over \tilde{s} —can be used to compute unit tangent and normal vectors of the airfoil shapes as well as curvature after right multiplication with corresponding scale variations. The continuous representation can then be expressed as

$$C(\tilde{s}; \mathbf{a}, \boldsymbol{\ell}) = Q(\tilde{s}; \mathbf{a})M(\boldsymbol{\ell}) \quad (12)$$

or

$$C(\tilde{s}; \mathbf{a}, \boldsymbol{\ell}) = Q(\tilde{s}; \mathbf{a})P(\boldsymbol{\ell}) \quad (13)$$

parametrized by a vector of polynomial coefficients \mathbf{a} and scale (length) variations $\boldsymbol{\ell}$ over the respective choice of manifold, generally for $M(\boldsymbol{\ell}) \in GL_2$ or specifically for $P(\boldsymbol{\ell}) \in S_{++}^2$. In other words, an interpolation approximating the continuous reconstruction of the curve is $\tilde{\mathbf{c}}(\tilde{s}; \mathbf{a}, \boldsymbol{\ell}) = (q_1(\tilde{s}; \mathbf{a}_1, \boldsymbol{\ell}), q_2(\tilde{s}; \mathbf{a}_2, \boldsymbol{\ell}))$, where q_i are parametrized linear combinations (over variations in $\boldsymbol{\ell}$) of two orthogonal Legendre polynomials constituting the columns of $Q(\cdot)$ with coefficients $\mathbf{a} = (\mathbf{a}_1, \mathbf{a}_2)$ over the continuous dimension of $C(\cdot; \mathbf{a}, \boldsymbol{\ell})$. In this context, $(\mathbf{a}_1, \mathbf{a}_2)$ represents a partition of the full set of coefficients \mathbf{a} into corresponding component functions (q_1, q_2) of the curve.

2.3.1. AI/ML comparisons and interpretations

We consider alternative approaches that leverage AI-based tools for dimension reduction and generative modeling, such as autoencoders or variational autoencoders (VAEs; Kingma & Welling, 2013; Kramer, 1991) and generative adversarial networks (GANs; Goodfellow et al., 2020). Autoencoder models learn non-linear reduced representations by mapping input data through a so-called information bottleneck (i.e., the latent space) before reconstructing it. The basic architecture of these models is the composition of an encoder π_{enc} with a decoder π_{dec} , with each component part comprised of multiple neural processing layers. Given a training data set $\{X_k\}$, we fit the model parameters $\mathbf{w} \in \mathbb{R}^D$ by minimizing some reconstruction loss

$$\underset{\mathbf{w} \in \mathbb{R}^D}{\text{minimize}} \sum_k \|X_k - (\pi_{\text{dec}} \circ \pi_{\text{enc}})(X_k; \mathbf{w})\|. \quad (14)$$

VAEs expand on traditional autoencoders by encouraging desirable distributions on the latent variables by adding a term such as the Kullback–Leibler divergence $KL(\pi_{\text{enc}}(X_k; \mathbf{w}) \parallel \rho(\mathbf{z}))$, where $\rho(\mathbf{z})$ is the target latent space distribution.

GANs are similarly comprised of two neural network models; however, unlike with autoencoders, these models are trained against each other. The generator network π_{gen} maps random latent vectors $\mathbf{z} \sim \rho$ to outputs in the space of the training data. The discriminator network π_{disc} maps data to $[0, 1]$ corresponding to a probabilistic prediction that the input data came from the generator or the training data. The network parameters are fit according

to a minimax optimization of the training data:

$$\begin{aligned} & \underset{\mathbf{w}_{\text{gen}} \in \mathbb{R}^{\text{p}_{\text{gen}}}}{\text{minimize}} \underset{\mathbf{w}_{\text{disc}} \in \mathbb{R}^{\text{p}_{\text{disc}}}}{\text{maximize}} \sum_k \log \pi_{\text{disc}}(X_k; \mathbf{w}_{\text{disc}}) \\ & + \sum_k \log(1 - \pi_{\text{disc}}(\pi_{\text{gen}}(X_k; \mathbf{w}_{\text{gen}}); \mathbf{w}_{\text{disc}})). \end{aligned} \quad (15)$$

This optimization ultimately encourages the generator to draw plausible samples from the training data distribution. The generator from GANs and the decoder from VAEs can both be used to map low-dimensional, latent space parametrizations to new realizations. Both methods have been applied to the design of airfoil shapes (Achour et al., 2020; Chen et al., 2019; Wang et al., 2022; Yang et al., 2022; Yonekura & Suzuki, 2021).

In practice, we seek a robust and interpretable parametrization for the decoder/generator model to act upon. For VAEs, this corresponds to π_{enc} being surjective onto the latent space. However, this property can be difficult to guarantee in general. In contrast, the proposed tensor representations are supported by Propositions 1–4 and Lemma 1. The implications of ambiguous properties on the VAE latent space are ill-posed constructions and parametrizations that are highly dependent on the stochastic training process for the model. Consequently, the resulting parametrizations are often overfit to the specific data types and sets used during training. Although the targeted loss terms can encourage desirable behavior, these results are not guaranteed. Despite this, geometric interpretations have led to a set of boundary-value problems defined by the pullback metric (geodesic equations) to offer improved notions of distances between points in latent space (Arvanitidis et al., 2017). Such innovations and interpretations are crucial in the continued development of VAEs.

GANs benefit from a well-defined parameter space, as the distribution over the latent variables is decided prior to network training. However, it can be difficult to control the relationship between the latent variables and their resulting shape generations, leading to poor interpretability and little insight into the intrinsic dimension of the shape parametrization. Similar to the VAEs, the resulting parametrization is heavily influenced by the randomized network initialization and training procedure. Furthermore, although the adversarial training encourages the generation of quality shapes, issues such as mode collapse and overparametrization of the networks may cause the generator to miss key novel shape designs that drive innovation. Lastly, the computational burden—and the associated energy costs—required to train such sophisticated models are considerable (Strubell et al., 2019).

In comparing these approaches to this work, the decoder π_{dec} from VAEs and the generator π_{gen} from GANs may be considered as analogues of the right inverse π^{-1} , which constitute a parametrization over a *local section of the fiber bundle* (Lee, 2006). Exploring this connection further, the analogous network latent spaces are akin to *normal coordinates*, defined in Lee (2006), over “parent” matrix manifolds that generate separable shape deformations in our context. These normal coordinates constitute a set of naturally defined parameters describing general manifold topologies, as opposed to the obscure latent space emulating a target distribution.

In Section 2.4, we describe how our principled separable representations (5) and (6) offer improved geometric interpretations with significantly reduced computational cost. In particular, our principled approach to shape representation takes advantage of rigorously studied matrix manifolds and linear algebra to avoid the need for general non-convex optimization and numerical in-

tegration of boundary-value problems when computing geodesics and distances over latent spaces. We also assert additional geometric interpretations beyond geodesics and distances that enable novel deformations of 3D shapes—a means of interpolating and applying consistent deformations to distinct 2D shapes. The result is an analytic generative model from a “learned” (data-driven) manifold of shapes.

2.4. Riemannian interpretations

We formally develop a data-driven framework for parametrizing elements over topologies of separable shape tensors, which leverage an extension of principal component analysis (PCA) to Riemannian manifolds. This requires a pair of fundamental intrinsic maps for mapping between a given manifold and a tangent space at a central element. We also discuss an improved notion of distance as lengths of geodesic curves over the manifold. Lastly, we present parallel transport as a method for applying consistent deformations to different shapes—motivating a novel approach to deform 3D blades. These interpretations are backed by sections detailing algorithms to compute all necessary maps over the presented manifolds.

In Section 2.1, we demonstrated how to define airfoil representations that separate important aerodynamic scale variations from higher order undulations in the shape. The methods for LA standardization, namely polar decomposition, map discrete airfoil shapes X to representative elements of the Grassmannian \bar{X} . Here, we provide a data-driven approach to parametrizing these separable shapes that leverages rigorously designed data sets of airfoil coordinates (UIUC Applied Aerodynamics Group, 2022) and/or systematically engineered airfoil expansion representations (Kulfan, 2008). The goal is to use ensembles of existing, well-designed shapes $\{X_k\}$ to construct the separable forms in (5) or (6). To do so, we perform a statistical analysis of a given ensemble of discrete shapes factored into the separable forms. This constitutes a data-driven perspective for reparametrizing shapes via separable tensors.

By introducing separability, we must consider the Riemannian geometry of $\mathcal{G}(n, 2)$ and S_{++}^2 , or possibly GL_2 , to parametrize our discrete shape space. In general, we consider any as a smooth manifold \mathcal{M} such that the Riemannian manifold is defined as (\mathcal{M}, g) for some choice of metric g that induces an inner product $g_p : T_p\mathcal{M} \times T_p\mathcal{M} \rightarrow \mathbb{R}$ over the tangent space $T_p\mathcal{M}$ for some $p \in \mathcal{M}$. Note that in this abstracted sense, p is a representative matrix element of the smooth manifold, whereas $\mathbf{v} \in T_p\mathcal{M}$ are generalized notions of “directions” in a tangent space at p but are also ultimately represented as matrices.

Building submanifolds from parent matrix manifolds offers several advantages over alternative approaches, including modern AI/ML methods discussed previously. This perspective supplements a novel treatment for learning a non-Euclidean manifold of discrete shapes from data, namely by Propositions 1 and 4. This supplements a more prudent notion of distance between discrete shapes and their separable deformations. Shapes in this framework are defined by “explainable” and “interpretable” representations induced by parent matrix manifolds. That is, there is no obscurity about our explicit parametrizations defining submanifolds of discrete shapes—an argument which is very difficult to develop in general for alternative frameworks. Moreover, the geometry of the parent manifolds is largely understood and supported by robust theoretical foundations. The advantage is that each computation involved in defining parameters of the submanifold can be explained using the interpretations of linear al-

gebra and Riemannian geometry. These explanations and interpretations are transparent and supplement a more precise critique of potential flaws in previous methods and applications.

The goal is to use a given ensemble of discrete airfoils to infer submanifolds of either $\mathcal{G}(n, 2) \times GL_2 \subset \mathbb{R}_*^{n \times 2}$ or $\mathcal{G}(n, 2) \times S_{++}^2 \subset \mathbb{R}_*^{n \times 2}$. Over the former, parametrizations through GL_2 are often prescribed by the design problem under consideration—e.g., using (1) to enforce structural/regulatory constraints dictating that a specified airfoil thickness at a given location along a wind turbine blade. In this case, scale variations and rotations are prescribed such that a fixed or constrained $M(\boldsymbol{\ell})$ is paired with unknown \mathbf{t} . Hence, the geometry of GL_2 is largely inconsequential for aerodynamic design given explicit definitions of $M(\boldsymbol{\ell})$, and only inferences involving $\mathcal{G}(n, 2)$ are required. For $\mathcal{G}(n, 2) \times S_{++}^2$, a more flexible representation of airfoil shapes is offered, independent of rotations/reflections. This is most useful for more general representations of airfoil shapes if scale variations independent of rotations are not prescribed—i.e., unknown $(\mathbf{t}, \boldsymbol{\ell})$. This insight provides a blueprint for leveraging different representations of airfoil shapes. That is, if you understand scale variations (e.g., traditional blade design), then compute $M(\boldsymbol{\ell})$ explicitly and pair it with inferred statistical properties of $[\tilde{X}](\mathbf{t})$ from data projected onto $\mathcal{G}(n, 2)$ to parametrize shape deformations using (5). Otherwise, attempt to infer statistical properties for pairs of $[\tilde{X}](\mathbf{t})$ and $P(\boldsymbol{\ell})$ over $\mathcal{G}(n, 2) \times S_{++}^2$, and parametrize shape deformations using (6) for more generalized airfoil design.

2.4.1. Intrinsic maps

There are two important intrinsic maps of Riemannian manifolds that are used to parametrize data-driven submanifolds. First, we require the exponential map $\text{Exp}_p : T_p\mathcal{M} \rightarrow \mathcal{M}$ that parametrizes an initial value problem for geodesic trajectories along the respective manifold \mathcal{M} beginning at $p \in \mathcal{M}$. Specifically, a geodesic curve beginning at p is parametrized over a direction $\mathbf{v} \in T_p\mathcal{M}$ as $\boldsymbol{\gamma}(t; p, \mathbf{v}) = \text{Exp}_p(t\mathbf{v})$ such that $\boldsymbol{\gamma}(0; p, \mathbf{v}) = p$. Second, we require the corresponding inverse $\text{Log}_p : \mathcal{M} \rightarrow T_p\mathcal{M}$ which parametrizes a boundary-value problem given two points connected by a geodesic trajectory over the respective \mathcal{M} . Figure 5a shows an example visualization of the Exp and Log maps over a 2-sphere. Compositions involving these maps with a corresponding basis in a fixed tangent space $T_p\mathcal{M}$ define “normal coordinates” of the manifold (Lee, 2006). We leverage reduced-dimension subspaces of these normal coordinates to parametrize discrete shape submanifolds from data.

2.4.2. Distances

Distance over a Riemannian manifold (\mathcal{M}, g) is defined as the infimum over a set of lengths—i.e., line integrals of the inner product of curve velocities induced by g —connecting two points in the manifold. Per the developments of Lee (2006), this definition of distance is consistent with that of zero acceleration curves (or geodesics) over the manifold such that these curves have a length-minimizing property. Consequently, geodesics induce an intuitive notion of shortest-distance curves whose lengths inform a metric over geodesic balls. These geodesic balls are defined as the image of Exp at a point over an open or closed ball in normal coordinates, as visualized in Fig. 5c. When restricted to a geodesic ball over the manifold, these geodesic distances are equivalently represented by the norm over the image of the Log map (Lee, 2006),

$$d_{\mathcal{M}} : \mathcal{M} \times \mathcal{M} \rightarrow \mathbb{R}_+ : (p, z) \mapsto \|\text{Log}_p(z)\|_g, \quad (16)$$

with the norm induced by the metric g at the corresponding point. In our case, the norm in (16) is the Frobenius norm, inducing d_G and $d_{S_{++}^2}$ with corresponding Log for the respective manifolds. In our data-driven setting, we assume data is concentrated within geodesic balls to leverage (16) for computing distances, given the desire to compute Log and Exp maps that define normal coordinate charts. However, the assumed restriction to geodesic balls is more generally extensible by constructing an atlas of normal coordinate charts using a collection of disjoint tangent spaces.

2.4.3. Riemannian statistics

We next introduce two algorithms that extend the fundamental data analysis technique of PCA to Riemannian manifolds as a so-called PGA (Fletcher et al., 2003; Pennec, 1999). The output of PGA will inform a parametrization over the Grassmannian $\mathcal{G}(n, 2)$, which defines the higher order perturbations to airfoil shapes. The basic premise of classical PCA is that, given a set of points as data, we can determine directions that maximize sample covariance G of the point set,

$$\underset{\mathbf{v} \in T, \mathbf{v}^T \mathbf{v} = 1}{\text{argmax}} \mathbf{v}^T G \mathbf{v}. \quad (17)$$

Writing the Lagrangian and solving for the stationarity condition in the optimization problem implies $(G - \lambda I)\mathbf{v} = 0$ for strictly non-negative λ . Thus, examining the decreasing ordered pairs $(\lambda_i, \mathbf{v}_i)$ of eigenvalues and eigenvectors offers a set of directions defining a basis for a reduced-dimension subspace over which the covariance-weighted inner product changes the most, on average.

We require two functionalities to extend PCA to Riemannian manifolds: (i) we must “center on the data” by an appropriate notion of an *intrinsic mean* over the manifold; then, (ii) we must “identify important directions” over the manifold that form an orthogonal frame for a *submanifold*. Figure 5b offers a useful visual analogue for the case of a sphere. For completeness, we present the two algorithms in detail as Algorithms 1 and 2. Algorithm 1 computes the intrinsic (Karcher or Fréchet) mean by computing the mean of the data in the tangent space of an iterative approximation restricted over the manifold. Algorithm 2 computes the corresponding PGA directions in the central tangent space using the SVD of the data lifted to the tangent space at the intrinsic mean computed from the previous algorithm. Together, these algorithms provide a framework for parametrizing separable shape tensors. Additional motivation and development for these algorithms can be found in Pennec (1999), Fletcher et al. (2003), and Fletcher and Joshi (2004).

In Algorithm 1, the choice of norm $\|\cdot\|$ over the tangent space in the case of matrix manifolds is taken as the Frobenius norm—induced by the choice of ambient inner product, $\text{tr}(A^T B)$. For Algorithm 2 applied to matrix manifolds, Step 2 requires $\text{vec} : \mathbb{R}^{n \times 2} \rightarrow \mathbb{R}^{2n}$, which stacks the columns of the matrices such that the k th columns of Δ_N are replaced with $\text{vec}(\text{Log}_{p_0}(p_k))$. This does not modify the induced metric implicit to Algorithm 2—which maximizes an approximated notion of a covariance-weighted inner product (Fletcher et al., 2003)—since $\text{tr}(A^T B) = \text{vec}(A)^T \text{vec}(B)$. Consequently, the left singular vectors of Δ_N still correspond to principal directions in $T_{p_0}\mathcal{M}$ that maximize sample covariance in normal coordinates proportional to $\Delta_N^T \Delta_N$. When mapping forward through these important directions, which are defined by the left singular vectors as columns of U_T to parametrize a submanifold, we must reshape once more prior to composition with Exp_{p_0} . Defining $\text{vec}^{-1} : \mathbb{R}^{2n} \rightarrow \mathbb{R}^{n \times 2}$ such that $\text{vec}^{-1}(\text{vec}(A)) = A$, the r -dimensional matrix submanifold of

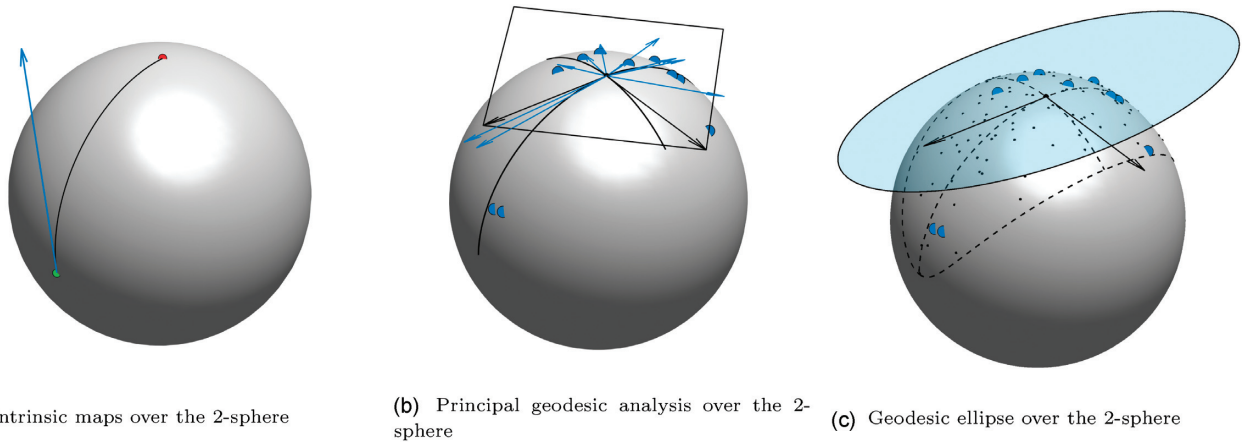


Figure 5: Visualizations of data-driven methods over the 2-sphere. (a) A visualization of the exponential $\text{Exp}_A(\Delta) = B$ and inverse exponential $\text{Log}_A(B) = \Delta$ maps over the sphere for points $A, B \in \mathcal{M}$ and corresponding tangent vector $\Delta \in T_A\mathcal{M}$. (b) Principal geodesic analysis (PGA) over the two-sphere embedded in \mathbb{R}^3 . The large black dot represents the intrinsic mean of the blue dots, which are provided as data. The black arrows represent PGA directions in the central tangent space at the intrinsic mean. The black lines represent geodesics along the PGA directions. (c) A geodesic ellipse and grid (dashed boundary) lifted to the normal coordinate neighborhood (light blue region with solid boundary). Smaller black dots are new samples generated uniformly over the submanifold spanned by both directions—constituting a generative model.

Algorithm 1. Intrinsic mean over Riemannian manifold.

Require: Data points $\{p_k\}_{k=1}^N \in \mathcal{M}$ and chosen convergence threshold $\epsilon > 0$

- 1: Set $p' = p_1$ and initialize \mathbf{v} such that $\|\mathbf{v}\| > \epsilon$
 - 2: **while** $\|\mathbf{v}\| \geq \epsilon$ **do**
 - 3: $\mathbf{v} = \frac{1}{N} \sum_{k=1}^N \text{Log}_{p'}(p_k)$
 - 4: $p' = \text{Exp}_{p'}(\mathbf{v})$
 - 5: **end while**
 - return** $p_0 = p' \in \mathcal{M}$
-

Algorithm 2. PGA directions.

Require: Data points $\{p_k\}_{k=1}^N \in \mathcal{M}$, chosen convergence threshold $\epsilon > 0$, and dimensionality r

- 1: Compute p_0 according to Algorithm 1 with parameter ϵ
 - 2: Compute matrix

$$\Delta_N = \frac{1}{\sqrt{N-1}} (\text{Log}_{p_0}(p_1), \dots, \text{Log}_{p_0}(p_N))$$
 - 3: Compute reduced SVD, up to $r \leq \dim(\mathcal{M})$,

$$\Delta_N = U_r \Sigma_r V_r^T$$
 - 4: Compute normal coordinates in the new basis,

$$T = U_r^T \Delta_N = \Sigma_r V_r^T$$
 - return** U_r with columns as the r directions in $T_{p_0}\mathcal{M}$ and T as the matrix of principal normal coordinates
-

interest becomes

$$\{p \in \mathcal{M} : p = \text{Exp}_{p_0}(\text{vec}^{-1}(U_r \mathbf{t}))\} \quad (18)$$

for a vector of PGA coefficients $\mathbf{t} \in \mathbb{R}^r$ and basis $U_r \in \mathbb{R}^{2n \times r}$. Once again, despite the composition with reshaping, the choice of ambient inner product remains consistent for $\mathbf{v} = U_r \mathbf{t}$, such that $\text{tr}(\text{vec}^{-1}(\mathbf{v})^T \text{vec}^{-1}(\mathbf{v})) = \mathbf{v}^T \mathbf{v}$.

2.4.4. Consistent deformations

To supplement 3D design for blade or wing shapes, we will require parallel translation to facilitate the mapping of consistent perturbations to different elements of the manifold. Given $p, z \in \mathcal{M}$, parallel translation acts as an isometry parametrizing zero-acceleration transport of tangent vectors over geodesic curves in the manifold, $\tau_{p\|z} : T_p\mathcal{M} \rightarrow T_z\mathcal{M}$. Note that parallel translation is unique and exists over any curve in \mathcal{M} . However, if defined over unique geodesic curves within a normal neighborhood, parallel translation has the advantage of not requiring any memory of the curve. That is, we can always reconstruct the original tangent vectors by “back-tracing” parallel translation over the choice of unique geodesics that are intrinsic to the manifold. This alleviates the computational burden by taking advantage of tensor parametrizations that are consistent with geodesic trajectories but do not require the otherwise burdensome numerical integration of the dynamics (Arvanitidis et al., 2017) for the two special cases of Grassmannian and SPD manifolds.

The mapping $\tau_{p\|z}$ preserves the notion of direction within a tangent space such that

$$g_p(\mathbf{u}, \mathbf{v}) = g_z(\tau_{p\|z}(\mathbf{u}), \tau_{p\|z}(\mathbf{v})) \quad (19)$$

for $\mathbf{u}, \mathbf{v} \in T_p\mathcal{M}$ and $p, z \in \mathcal{M}$. Consequently, given a basis defined at a particular point, we can map directions in the span of this basis to new tangent spaces along geodesics parametrized by endpoints, i.e., $\tau_{p\|z}(\cdot) = \tau_{\parallel}(\cdot; p, z)$. These connected directions have equivalent inner products taken in the central tangent space, constituting the closest analogue of a consistent parameter direction over distinct tangent spaces.

Applying this process to airfoil shapes, we can use Algorithm 1 to identify the intrinsic mean shape p_0 and its central tangent space $T_{p_0}\mathcal{M}$. Next, using Algorithm 2, we compute a basis in this tangent space describing dominant perturbations about this mean shape. Given a particular deformation to the mean shape $U_r \mathbf{t} \in T_{p_0}\mathcal{M}$ with coefficients \mathbf{t} , we can then consistently map these deformations to new shapes via $\tau_{\parallel}(U_r \mathbf{t}; p_0, p)$. Such a procedure preserves the original notion of direction in the parametrization at $T_{p_0}\mathcal{M}$ but assigns the deformation to the shape p . The result

is our definition of “consistent deformations” of distinct shapes—specifically, consistency in the inner product at the intrinsic mean.

2.4.5. Geometry of Riemannian product manifolds

Motivated by the separability of our shape parametrizations from (6), we explore the concept of a product manifold. Given (\mathcal{M}_1, g_1) and (\mathcal{M}_2, g_2) , we consider the topology of a Riemannian product manifold $(\mathcal{M}_1 \times \mathcal{M}_2, g_1 + g_2)$. Under this construction, we take advantage of the ability to parametrize geodesics in a componentwise manner. Letting Exp_i denote the exponential map for the manifold \mathcal{M}_i , $(\text{Exp}_{1,p}(\mathbf{t}\mathbf{u}), \text{Exp}_{2,z}(\mathbf{t}\mathbf{u}))$ is the corresponding geodesic over $(\mathcal{M}_1 \times \mathcal{M}_2, g_1 + g_2)$ for $t \in \mathbb{R}$. This allows us to independently formulate the necessary intrinsic maps over distinct manifolds and combine these computations in a componentwise manner to build shapes using our separable representations. Given this convenience, the general submanifold of discrete shapes assumes the defined product manifold topology over $\mathcal{G}(n, 2) \times S_{++}^2$ with the canonical ambient metric and affine-invariant metric, respectively.

2.4.6. Geometry of orthogonal matrices: $\mathcal{G}(n, 2)$

Following the developments of Edelman et al. (1998), Gallivan et al. (2003), and Bendokat et al. (2020), we present algorithmic routines for the computation of the Exp and Log maps over $\mathcal{G}(n, 2)$ for completeness (we assume the Riemannian metric $\text{tr}(A^T B)$ inherited from embedding space; Absil et al., 2008).

Algorithm 3. Thm. 2.3 (Edelman et al., 1998), Grassmann exponential with Stiefel representatives.

Require: Representative matrix $\tilde{X} \in \mathbb{R}_*^{n \times 2}$ with orthonormal columns and direction $\Delta \in T_{[\tilde{X}]} \mathcal{G}(n, 2) \subset \mathbb{R}_*^{n \times 2}$

- 1: Compute thin SVD,
 $\Delta = U_\Delta \Sigma_\Delta V_\Delta^T$
 - 2: Compute, with $\cos(\cdot)$ and $\sin(\cdot)$ acting only on diagonal entries,
 $\tilde{Y} = \tilde{X} V_\Delta \cos(\Sigma_\Delta) V_\Delta^T + U_\Delta \sin(\Sigma_\Delta) V_\Delta^T$
- return** $[\tilde{Y}] = \text{Exp}_{[\tilde{X}]}(\Delta) \in \mathcal{G}(n, 2)$
-

First, we present the exponential map as Algorithm 3. Leveraging this algorithm, we can compute $\text{Exp}_{[\tilde{X}]}(\Delta)$ to take a unit step in the direction Δ from the equivalence class $[\tilde{X}]$. As a reparametrization, we can arbitrarily scale distances along this geodesic via the one-parameter subgroup $\gamma(t; [\tilde{X}], \Delta) = \text{Exp}_{[\tilde{X}]}(t\Delta)$ for all $t \in \mathbb{R}$, identifying the base point as $\gamma(0; [\tilde{X}], \Delta) = [\tilde{X}]$. For general $\mathcal{G}(n, q)$, this algorithm has computational complexity $O(nq^2)$ by virtue of a generalized SVD as the only iterative procedure (Gallivan et al., 2003). In our fixed ambient spatial dimension $q = 2$, we anticipate linear scaling of the computational burden with increasing n (Bendokat et al., 2020; Gallivan et al., 2003; Zimmermann, 2019).

Next, we present Algorithm 4 for computing the inverse exponential map (Absil et al., 2004; Bendokat et al., 2020; Gallivan et al., 2003; Zimmermann, 2019). In general, this algorithm has computational complexity $O(nq^2)$, which again implies linear complexity with landmark refinements for fixed ambient spatial dimension $q = 2$. However, treating this map simply as a matrix transformation is subject to $\text{Exp}_{[\tilde{X}]}(\text{Log}_{[\tilde{X}]}(\tilde{Y})) \neq \tilde{Y}$ with mismatch up to rotation (Bendokat et al., 2020; Zimmermann, 2019). The mismatched matrix still corresponds to the same $[\tilde{Y}] = [\tilde{Y}']$ such that $\tilde{Y} \sim \tilde{Y}'$; however, it lacks the desirable property that $\text{Exp}_{[\tilde{X}]} \circ \text{Log}_{[\tilde{X}]}$ returns the same element of the equivalence class up to re-

flections. This computational inconvenience is corrected by Procrustes matching (Bendokat et al., 2020; Zimmermann, 2019). This concept comes into play for the purposes of 3D blade and wing cross-section interpolation, where a sequence of representative matrices of the Grassmannian intended for interpolation can be mismatched up to rotation, requiring Procrustes analysis to align. This is discussed further in Section 2.5. Despite this correction, there remains a possibility that the shape is reflected in the composition $\text{Exp}_{[\tilde{X}]} \circ \text{Log}_{[\tilde{X}]}$. Additional checks or constraints may be required for detecting such a condition, depending on the data.

Algorithm 4. Alg. 5.3 (Bendokat et al., 2020), Alg. 10 (Zimmermann, 2019), and Grassmann logarithm with Stiefel representatives.

Require: Representative matrices $\tilde{X}, \tilde{Y} \in \mathbb{R}_*^{n \times 2}$ with orthonormal columns

- 1: Take the matrix product, $\tilde{Q} = \tilde{X}^T \tilde{Y}$
 - 2: Define orthogonal projection (to normal space),
 $\pi_n^\perp = (I_n - \tilde{X} \tilde{X}^T)$
 - 3: Compute thin SVD,
 $\pi_n^\perp (\tilde{Y} \tilde{Q}^{-1}) = U_\Delta \Sigma_\Delta V_\Delta^T$
 - 4: Compute, with $\arctan(\cdot)$ acting only on diagonal entries,
 $\Delta = U_\Delta \arctan(\Sigma_\Delta) V_\Delta^T$
- return** $\Delta = \text{Log}_{[\tilde{X}]}([\tilde{Y}]) \in T_{[\tilde{X}]} \mathcal{G}(n, 2)$
-

A modified version of Algorithm 4 such that $\text{Exp}_{[\tilde{X}]}(\text{Log}_{[\tilde{X}]}(\tilde{Y})) = \tilde{Y}$ up to reflections can be found in Zimmermann (2019) and Bendokat et al. (2020) but is omitted from this presentation for brevity. The motivation here is to highlight the relatively inexpensive computations of these intrinsic mappings—linear growth in computational complexity for refined shapes in fixed ambient spatial dimension—for the purposes of informing a statistical analysis and separable representation of shapes. The modified version of Algorithm 4 requires two SVD computations but remedies the representative rotational mismatch and, as an added benefit, avoids the calculation of \tilde{Q}^{-1} (Zimmermann, 2019).

Algorithm 5. Thm. 2.4 (Edelman et al., 1998) and Grassmann parallel transport along geodesic with Stiefel representatives.

Require: Representative matrix $\tilde{X} \in \mathbb{R}_*^{n \times 2}$ with orthonormal columns and directions $\Delta, \Gamma \in T_{[\tilde{X}]} \mathcal{G}(n, 2) \subset \mathbb{R}_*^{n \times 2}$, and scalar $t \in \mathbb{R}$

- 1: Compute thin SVD,
 $\Delta = U_\Delta \Sigma_\Delta V_\Delta^T$
- 2: Compute parallel transport operator, with $\cos(\cdot)$ and $\sin(\cdot)$ acting only on diagonal entries,

$$\tau_{\parallel}(t; [\tilde{X}], \Delta) = (\tilde{X} V_\Delta U_\Delta) \begin{pmatrix} -\sin(t \Sigma_\Delta) \\ \cos(t \Sigma_\Delta) \end{pmatrix} U_\Delta^T + I_n - U_\Delta U_\Delta^T$$

return $\Gamma(t; [\tilde{X}], \Delta) = \tau_{\parallel}(t; [\tilde{X}], \Delta) \Gamma$

Lastly, we present the algorithm for parallel translation (Edelman et al., 1998). Using Algorithm 5, $\Gamma(t; [\tilde{X}], \Delta) \in T_{\gamma(t; [\tilde{X}], \Delta)} \mathcal{G}(n, 2)$ is the parallel translation of Γ along the geodesic a distance scaled by t emanating from $[\tilde{X}]$ in the direction Δ . Again, a comparable computational complexity is achieved by only requiring the thin SVD of the direction Δ —defining the geodesic $\gamma(t; [\tilde{X}], \Delta)$ —over which Γ is parallel translated by the $n \times n$ matrix $\tau_{\parallel}(t; [\tilde{X}], \Delta)$.

Notice that Algorithm 5 is parametrized by a “point” and a “direction” as opposed to two endpoints, per the development of consistent deformations, $\tau_{p\parallel z}(\cdot) = \tau_{\parallel}(\cdot; p, z)$. Both interpretations are

valid (within a geodesic ball) because the direction and magnitude parametrizing the geodesic from p to z (generally) is implied by $\text{Log}_p(z) \in T_p\mathcal{M}$, while the endpoints in Algorithm 5 are implied by $\gamma(0; [\tilde{X}], \Delta)$ and $\gamma(t; [\tilde{X}], \Delta)$ using Algorithm 3. In our implementations, the endpoint parametrization is most appropriate, thus modifying the Grassmannian parallel translation of $\Gamma \in T_{[\tilde{X}]}G(n, 2)$ to $T_{[\tilde{Y}]}G(n, 2)$, as $\tau_{\parallel}(\Gamma; [\tilde{X}], [\tilde{Y}]) = \tau_{\parallel}(1; [\tilde{X}], \text{Log}_{[\tilde{X}]}([\tilde{Y}]))\Gamma$ by composing Algorithm 5 with Algorithm 4.

2.4.7. Geometry of the symmetric space: S_{++}^2

Next, we review the necessary computations over the convex cone of 2×2 SPD matrices, S_{++}^2 . We follow the developments of Fletcher and Joshi (2004), Pennec et al. (2006), and Bonnabel and Sepulchre (2010) for algorithms to compute Exp and Log maps, while Sra and Hosseini (2015) additionally expound on the computation of parallel transport. More recent implementations and interpretations are facilitated by Yair et al. (2019) and Zimmermann (2019). Consistent with the referenced developments, we assume the *affine-invariant metric* for all computations over S_{++}^2 . For brevity, we reiterate the presentation of Zimmermann (2019) and note that a consistent yet more systematic implementation is discussed in Fletcher and Joshi (2004) for the purposes of computing $p^{1/2}$ and $p^{-1/2}$.

Algorithm 6. SPD exponential (Fletcher & Joshi, 2004; Zimmermann, 2019).

Require: Matrix $P \in S_{++}^2$ and direction $S \in T_p S_{++}^2 \cong \text{Sym}(2)$
 1: $D = p^{1/2} \exp(p^{-1/2} S p^{-1/2}) p^{1/2}$
return $D = \text{Exp}_p(S) \in S_{++}^2$

The exponential map over S_{++}^2 is stated as Algorithm 6. This algorithm induces the one-parameter subgroup for moving along geodesics emanating from P in the direction S as a 2×2 symmetric matrix, $\gamma(t; P, S) = \text{Exp}_P(tS)$. Note that “exp” represents the matrix exponential which is distinct from “Exp,” given by the algorithm for a particular choice of Riemannian metric.

Algorithm 7. SPD logarithm (Fletcher & Joshi, 2004; Zimmermann, 2019).

Require: Matrices $P, D \in S_{++}^2$
 1: $S = p^{1/2} \log(p^{-1/2} D p^{-1/2}) p^{1/2}$
return $S = \text{Log}_p(D) \in T_p S_{++}^2 \cong \text{Sym}(2)$

The inverse exponential map is stated as Algorithm 7. Note that in this algorithm, “log” represents the matrix logarithm (distinct from “Log”).

Finally, parallel translation is given by Algorithm 8. In this algorithm, parallel translation is parametrized by the two endpoints of the geodesic curve, $\tau_{\parallel}(S; P, D) = \tau_{P \parallel D}(S)$, consistent with the generalizations described in Section 2.4.4.

These routines, combined with the Grassmannian routines, offer a complete picture of the necessary computations for learning a manifold of discrete shapes for 2D and 3D blade design and interpolation. The improved notion of distances between shapes informs more favorable interpolations and distributions for numerical studies and supplements improved shape representations by

Algorithm 8. SPD parallel translation along geodesic (Sra & Hosseini, 2015; Yair et al., 2019).

Require: Matrices $P, D \in S_{++}^2$ and direction $S \in T_p S_{++}^2 \cong \text{Sym}(2)$
 1: Compute $E = (DP^{-1})^{1/2}$
return $ESE^T = \tau_{\parallel}(S; P, D) \in T_D S_{++}^2$

regularizing deformations—i.e., constraining to data-driven submanifolds. Moreover, this approach constitutes a more principled perspective for learning a submanifold of discrete shapes from data with reduced computational costs compared to alternative ML-based methods.

2.5. Grassmannian blade interpolation

We discuss the procedure for applying the framework of separable shape tensors to interpolate a sequence of 2D shapes into a 3D blade/wing. We also discuss the implications of a “Procrustes clustering” approach to select best representative matrices from equivalence classes for interpolation.

The separable shape tensor framework for airfoil representation has the added benefit of enabling the design of 3D wings and blades. In the context of wind energy, blade shapes are defined by a limited number of landmark airfoils located at different blade-span positions, as well as by profiles of twist, chordal scaling, and bending. Defining the full blade shape given the relatively small number of defining airfoil shapes along the blade is a nontrivial problem. Simple interpolation techniques result in undesirable blade features, such as kinks or dimples, in the regions between airfoils. Currently, airfoils must be designed as collections or families that will interpolate smoothly to construct the blade. The goal here is to define an interpolation of these shapes—independent of the prescribed affine deformations—that results in physically relevant blade definitions. In addition to interpolating between designed airfoil shapes, we may seek a separable representation from measured blades and subsequently infer a smoothly varying set of affine deformations over discrete blade-span positions corresponding to twist, scaling, and bending profiles of the blade.

2.5.1. Interpolation procedure

Given a sequence of matrices $(X_k) \in \mathbb{R}_*^{n \times 2}$ for $k = 1, \dots, N$ that represent landmark airfoil shapes in a wing or turbine blade, we induce the corresponding sequence of equivalence classes $([\tilde{X}_k]) \in G(n, 2)$ located at blade-span positions $\eta_k \in \mathcal{S} \subset \mathbb{R}$ from root to tip. An example of such landmark shapes is depicted by the red-yellow curves in Fig. 6. We define a piecewise geodesic path over the Grassmannian to interpolate representative airfoil shapes from $([\tilde{X}_k])$. This results in a continuous representation of the 3D blade shape using piecewise geodesic paths over ordered blade-span positions η_k along a non-linear representative manifold of shapes. In practice, we use piecewise geodesic interpolation $\gamma(\tilde{t}; [\tilde{X}_k], \text{Log}_{[\tilde{X}_k]}([\tilde{X}_{k+1}]))$ via Algorithms 3 and 4 for all $\tilde{t} \in [0, 1]$ and $k = 1, \dots, N - 1$ enumerating the sequence of geodesics interpolating between representative shapes over the Grassmannian. This procedure is also described as Algorithm 2 of Zimmermann (2019) and is summarized below.

To implement this approach, we must first reconcile differences in length scales over the piecewise geodesic Grassmannian curve and the spanwise physical distances between the airfoils within the blade. We define a monotonic reparametrization to consider a mapping from the physically relevant blade-span position $\eta \in \mathcal{S}$ to

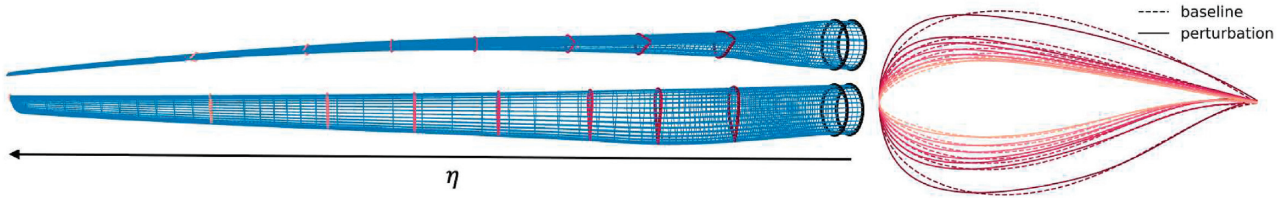


Figure 6: Example of an IEA 15-MW blade wire-frame (left-hand panel) obtained from interpolation of the solid-color cross-sections (right-hand panel) omitting the unperturbed circles. Note that consistent perturbations to the shape (right-hand panel) are applied to all of the baseline airfoils in the blade. Visually, we observe each of the distinct airfoils sampled from distinct classes deformed in a markedly similar manner. Also, note the curved span axis (bending along η), which is easily accounted for by including planar shapes in \mathbb{R}^3 , then rotating the shapes into a plane with normal tangent to the span axis.

the corresponding cumulative distance over the Grassmannian, $\varphi: \eta \rightarrow t$. In practice, this mapping can be built from a PCHIP of $\{(\eta_k, t_k)\}$ as a monotonically increasing function such that

$$t_k = \begin{cases} 0, & k = 1 \\ \sum_{i=1}^{k-1} d_G([\tilde{X}_{i+1}], [\tilde{X}_i]), & k = 2, \dots, N. \end{cases} \quad (20)$$

Then, within any piecewise interval $[t_k, t_{k+1}] = [\varphi(\eta_k), \varphi(\eta_{k+1})]$, we scale $[t_k, t_{k+1}]$ to $[0, 1]$ to build interpolated shapes over the subinterval, informing $[\tilde{X}](\varphi(\eta))$ for all $\eta \in [\eta_k, \eta_{k+1}]$. As a three-step procedure, given any $\eta \in \mathcal{S}$: (i) convert to the appropriate cumulative Grassmannian distance $t = \varphi(\eta)$ and identify the corresponding subinterval, $t \in [t_k, t_{k+1}]$; (ii) scale to a normalized coordinate $\tilde{t} = (\varphi(\eta) - t_k) / (t_{k+1} - t_k)$; and (iii) compute $\text{Exp}_{[\tilde{X}_k]}(\tilde{t} \text{Log}_{[\tilde{X}_k]}([\tilde{X}_{k+1}]))$ using the composition of Algorithms 3 and 4. Finally, to map the interpolated shapes over the Grassmannian back to physically relevant scales, we apply appropriate affine deformations using six regularized splines of data or explicit parametrizations $M(\eta)$ and $\mathbf{b}(\eta)$:

$$X(\eta) = (\tilde{X} \circ \varphi)(\eta)M(\eta) + \mathbf{1}_{n,2} \text{diag}(\mathbf{b}(\eta)). \quad (21)$$

In Section 2.5.2, we discuss the implications of inferring $M(\eta)$ and $\mathbf{b}(\eta)$ from measured data (X_k) using splines.

As an example computation, on a laptop (2.4 GHz 8-Core Intel Core i9 macOS Catalina Memory: 32 GB 2667 MHz DDR4), the interpolation routine with reparametrized shapes according to $n = 401$, with $N = 10$ nominal cross-sections as input and a refinement of 100 cross-sections defining the wire-frame, took approximately 0.04 seconds, on average. The corresponding blade is shown in Fig. 6, with a resulting structured surface mesh shown in Fig. 7. Varying refinements up to 1000 new cross-sections defining the wire-frame took 0.12 seconds, and refinements up to 100 000 new cross-sections took 14.9 seconds, on average (all else fixed). For comparison, it often takes longer to read the 10 nominal cross-sections into memory than it does to run the interpolation routine with refinements between 100 and 1000 cross-sections. Code and examples are available at Doronina et al. (2022).

2.5.2. Procrustes clustering

If the affine scale variations are implicitly encoded in the original data (X_k)—i.e., the sequence of discrete airfoils has already been appropriately scaled to size and orientation and do not constitute shapes with fixed orientation and unit-chord—then computations of the discrete centers of mass $\mathbf{b}_k = 1/n X_k^T \mathbf{1}_{n,1}$ and (\tilde{M}_k) , using (2) for (X_k), can be utilized. With data $\{(\eta_k, \tilde{M}_k, \mathbf{b}_k)\}$, we construct six entrywise splines over strictly increasing (η_k) , defining $M(\eta) = \tilde{M}(\eta)$ and $\mathbf{b}(\eta)$ in (21) with appropriate endpoint conditions. However, large uncontrolled variations in (\tilde{M}_k) , namely from rotation, may be problematic in the construction of the four corresponding entrywise splines.

Recalling the concerns with Algorithm 4 about a rotational mismatch between representative shapes from equivalence classes, we perform a “Procrustes” clustering along the reversed order of representative shapes—i.e., for $k = N, \dots, 2$, we solve

$$R_{k-1} = \underset{R \in \mathcal{O}(2)}{\text{argmin}} \|\tilde{X}_k - \tilde{X}_{k-1}R\|_F \quad (22)$$

then apply the computed rotation to the LA-standardized shape $\tilde{X}_{k-1}R_{k-1}$ and reassign scale variations to $\tilde{M}_{k-1}R_{k-1}$ so that the reversed sequence of shapes are best matched (sequentially) for interpolation. Our intuition is that wind turbine shapes are typically more similar from tip to hub, and the hub shape is often circular and thus invariant under rotation; thus, reversing the order may be desirable but is seemingly unnecessary.

This is an important caveat when inverting the shapes in (21) back to the physically relevant scales for subsequent affine deformations inferred from data. As a procedural interpretation, from the blade tip shape \tilde{X}_N to the blade hub shape \tilde{X}_1 , we sequentially match the representative LA-standardized shapes via Procrustes analysis (Gower, 1975) using (22). This offers rotations that can be applied to representative LA-standardized airfoils for matching—but does not modify the underlying piecewise geodesic interpolation along the Grassmannian, which is independent of these rotations. Consequently, we cluster the sequence of representative shapes \tilde{X}_k by optimal rotations in each $[\tilde{X}_k]$ to ensure they are best oriented from tip to hub and to mitigate concerns about large variations in entrywise splines of $\tilde{M}(\eta)$.

3. Data-Driven Representations

We apply the detailed developments of the Riemannian interpretations to learn the manifold coordinates $(\mathbf{t}, \boldsymbol{\ell})$ that define a generative airfoil model from data in seconds. We then demonstrate the generation of novel 3D blade shapes by consistently deforming landmark airfoils with chosen parameter dimensionality independent from the number of interpolated 2D cross-sections.

To explore separable shape representations, we use a data set containing 1000 perturbations to 16 baseline airfoils for a total of 16 000 shapes from the NREL 5-MW, Technical University of Denmark 10-MW, and IEA 15-MW reference wind turbines (Bak et al., 2013; Gaertner et al., 2020; Jonkman et al., 2009). The baseline airfoils are defined by 18 nominal CST coefficients with the trailing edge thickness coefficients set to zero. We then perturb these 18 coefficients by up to $\pm 20\%$ of their nominal value to create an ensemble of random airfoils.

The left-hand panel of Fig. 8 shows a marginal 2D slice through the 18D space of CST coefficients that defines the collection of shapes under consideration. Note that across the 16 baseline shapes, the groups of perturbations to nominal CST coefficients

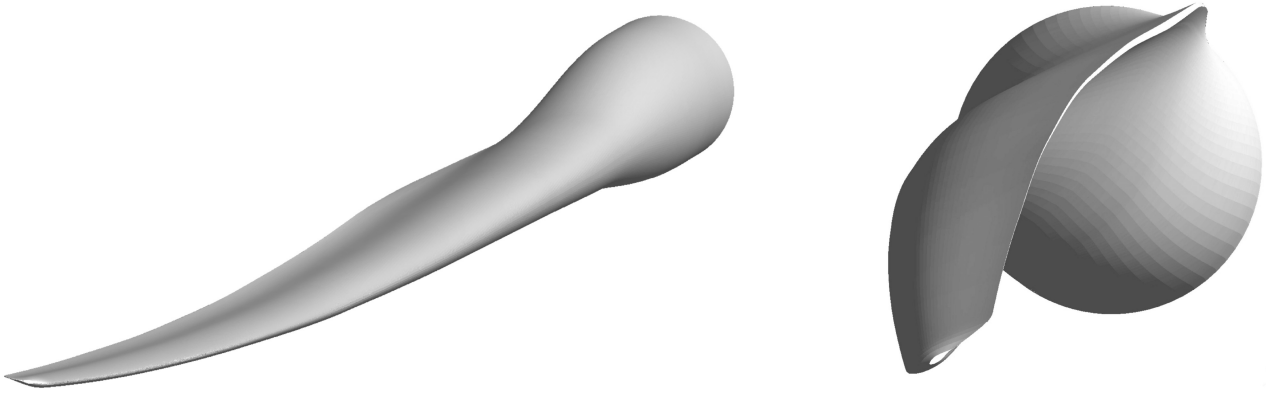


Figure 7: Structured surface mesh of 10 nominal cross-sections interpolated to 100 refined cross-sections, defining a “through curves” surface representation from B-splines of discrete shapes with $n = 401$. The mesh was generated using Gmsh (Geuzaine & Remacle, 2009) with the Grassmannian interpolation wire-frame provided as input. The input wire-frame to the surface meshing routine is consistent with the wire-frame depicted in Fig. 6.

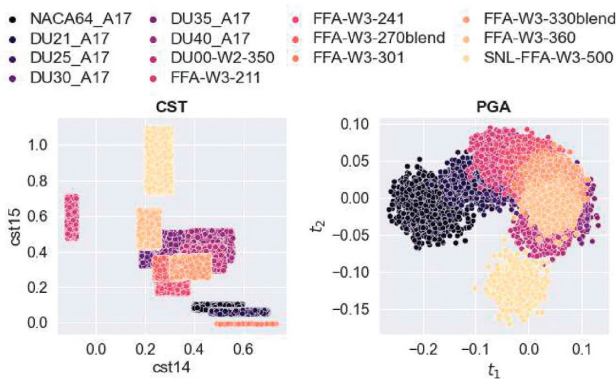


Figure 8: Comparison of the airfoil data over two of the 18 total CST parameters (left-hand panel) and two of the four total normal coordinates (right-hand panel), with colors indicating different named classes of designed airfoils. Notice that the transformed normal coordinates resemble a Gaussian mixture.

create a complex, highly disjoint design domain given the variety of airfoil classes. This can significantly impact the performance of various AI/ML algorithms in analyzing airfoils across this domain. We demonstrate how the proposed separable representation addresses these issues with the CST representation.

3.1. Principal geodesic deformations

We describe the process for applying the framework of separable shape tensors to learning coordinates $(\mathbf{t}, \boldsymbol{\ell}) \in \mathcal{T} \times \mathcal{P}$ for aerodynamic design and generation of new shapes. We explicitly parametrize two submanifolds used to define the shape product manifold. Additionally, interpolation is revisited in this context for the construction of 3D wings/blades from sequences of 2D shapes. Numerical examples demonstrate an improvement in shape generation along random continuous sweeps over the inferred Grassmannian parameter domain.

We infer nonparametric data-driven airfoil deformations by applying PGA (Algorithm 2), using Algorithms 3 and 4 and/or Algorithms 6 and 7 to compute the required intrinsic maps for the respective matrix manifolds. This informs an approximated *central tangent space* at a Karcher mean, denoted $T_{[\bar{x}_0]} \mathcal{G}(n, 2)$ or $T_{[P_0]} S_{++}^2$, computed with Algorithm 1 using the corresponding intrinsic maps for either manifold of interest. The subsequent image of Exp , given at respective Karcher means over subspaces $\mathcal{T} \subseteq T_{[\bar{x}_0]} \mathcal{G}(n, 2)$

or $\mathcal{P} \subseteq T_{[P_0]} S_{++}^2$, define *planar sections* (Lee, 2006) of the manifolds as submanifolds, $\text{Exp}_{[\bar{x}_0]}(\mathcal{T}) \subseteq \mathcal{G}(n, 2)$ or $\text{Exp}_{[P_0]}(\mathcal{P}) \subseteq S_{++}^2$. In this way, PGA and the product manifold construction constitute a manifold learning procedure for computing important submanifolds that represent a design space of physically relevant airfoil shapes inferred from provided data.

Based on the strength of the decay in eigenvalues over $T_{[\bar{x}_0]} \mathcal{G}(n, 2)$, we take the first r eigenvectors as a reduced basis for Grassmannian PGA deformations. For PGA over the SPD matrix manifold, we retain all three dimensions of S_{++}^2 . To compute coordinates $\mathbf{t}_k \in \mathcal{T}$ or $\boldsymbol{\ell}_k \in \mathcal{P}$ corresponding to each X_k in a new basis returned by Algorithm 2, we map LA-standardized airfoils to normal coordinates of $T_{[\bar{x}_0]} \mathcal{G}(n, 2)$ or $T_{[P_0]} S_{++}^2$ via inner products with the computed basis. In particular, in Algorithm 2, we compute

$$\mathbf{t}_k = T_r^T \text{vec}(\text{Log}_{[\bar{x}_0]}([\bar{X}_k])) \quad (23)$$

or

$$\boldsymbol{\ell}_k = L_3^T \text{vec}(\text{Log}_{[P_0]}(P_k)) \quad (24)$$

with T_r and L_3 as the approximated bases returned by Algorithm 2 applied to either matrix manifold of interest. Note that the “vec” operation applied to symmetric matrices in $T_{[P_0]} S_{++}^2$ only “vectorizes” the three unique entries. In a similar manner, vec^{-1} in this sense would duplicate the corresponding symmetric entry when mapping to the representative matrix, such that $\text{vec}^{-1}(\text{vec}(P)) = P$. Also note that the coordinates \mathbf{t}_k and $\boldsymbol{\ell}_k$ correspond to the columns of the principal normal coordinate matrix T returned by Algorithm 2. This implies definitions of parameter spaces as $\mathcal{T} \subseteq T_{[\bar{x}_0]} \mathcal{A}_r$ and $\mathcal{P} \subseteq T_{[P_0]} S_{++}^2$.

As an example, on a laptop (2.4 GHz 8-Core Intel Core i9 macOS Catalina Memory: 32 GB 2667 MHz DDR4), we randomly subsampled $N = 10\,000$ shapes from a database of 35 035 total shapes. Then, 10 sets of random draws with $N = 10\,000$ were fed to Algorithms 1 and 2 which were applied over the Grassmannian for $n = 401$ and $\epsilon = 1e - 8$. Algorithm 1 ran in 14.3 seconds, on average, over the 10 draws, while Algorithm 2—including the computational time of Algorithm 1—ran in 18.2 seconds (an additional 3.9 seconds) on average. Again, code and examples are available at Doronina et al. (2022).

Having defined reduced-dimension normal coordinates $\mathbf{t}_k \in \mathcal{T}$ with paired $\boldsymbol{\ell}_k \in \mathcal{P}$, we restrict \mathcal{T} and \mathcal{P} as compact sets over the respective tangent spaces, which contain PGA coordinates described by an appropriate distribution—e.g., uniform over a ball

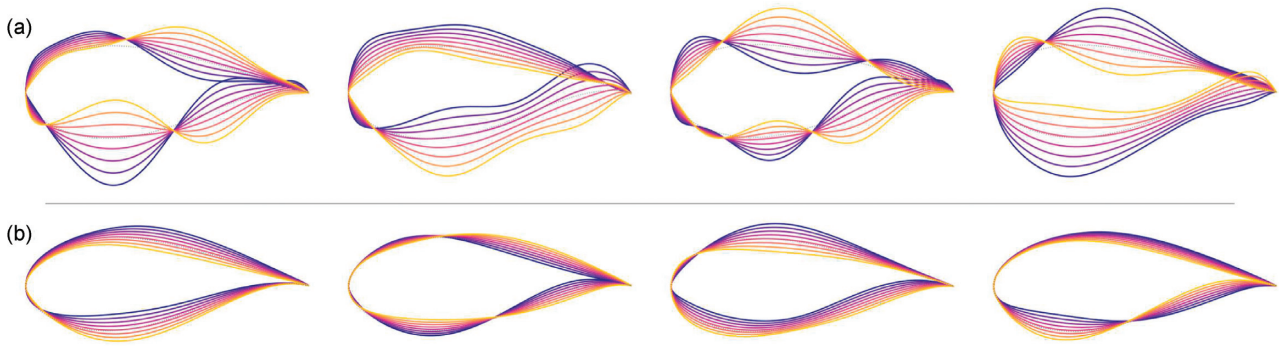


Figure 9: A series of random corner-to-corner sweeps across a hypercube containing all points partially visualized in Fig. 8 through (a) CST parameter space and (b) the Grassmannian submanifold \mathcal{A}_4 . The sweeps represent geodesic trajectories from yellow to blue shapes passing through the origin of the respective coordinate systems—CST parameter space is assumed to be flat.

containing the data $\{\mathbf{t}_k\}$ and/or $\{(\mathbf{t}_k, \ell_k)\}$. Then, the set of all linear combinations of the principal components, $T_r \mathbf{t}$ for all $\mathbf{t} \in \mathcal{T}$ and $L_3 \ell$ for all $\ell \in \mathcal{P}$, define an $(r + 3)$ -dimensional domain over $\mathcal{T} \times \mathcal{P} \subset T_{[\bar{X}_0]} \mathcal{G}(n, 2) \times T_{\mathbb{P}_0} S_{++}^2$. This parametrizes a section of the Grassmannian (r -submanifold) for all $\mathbf{t} \in \mathcal{T} \subset \mathbb{R}^r$ and $\Delta(\mathbf{t}) = \text{vec}^{-1}(T_r \mathbf{t})$:

$$\mathcal{A}_r = \{[\bar{X}] \in \mathcal{G}(n, 2) : [\bar{X}] = \text{Exp}_{[\bar{X}_0]}(\Delta(\mathbf{t}))\}, \quad (25)$$

and the SPD matrix manifold for all $\ell \in \mathcal{P}$:

$$\mathcal{S}_3 = \{P \in S_{++}^2 : P = \text{Exp}_{\mathbb{P}_0}(\text{vec}^{-1}(L_3 \ell))\}. \quad (26)$$

The resulting parametrized Riemannian submanifold $\mathcal{A}_r \times \mathcal{S}_3$ becomes the product manifold of interest for generating rotation/reflection-invariant airfoils $\bar{X}(\mathbf{t})P(\ell)$ according to (6). Moreover, when composed with subsequent rotations and span-wise reparametrizations, the data-driven representation (6) can also be used to interpolate any blade or wing over paired piecewise geodesics across respective manifolds. This procedure is consistent with the previous interpolation procedure, but is now applied independently over respective manifolds as

$$X(\eta) = (\bar{X} \circ \varphi)(\eta)(P \circ \psi)(\eta)R(\eta) + 1_{n,2} \text{diag}(\mathbf{b}(\eta)) \quad (27)$$

utilizing reparametrization over SPD distances as $\psi: \eta \rightarrow \ell$ and inferred (implicit) or parametrized (explicit) rotations $R(\eta)$ with paired translations $\mathbf{b}(\eta)$.

Truncating the Grassmannian principal basis to the first $r = 4$ components (based on the decay in PGA eigenvalues), we significantly reduce the number of parameters needed to define a rich set of airfoil deformations. Consequently, we have “learned” a 4D data-driven submanifold of airfoil undulations, \mathcal{A}_4 , which are independent of affine deformations. New parameters are now coordinates of this 4D subspace $\mathbf{t} \in T_{\mathbf{0}} \mathcal{A}_4 \cong \mathbb{R}^4$ over the tangent space at the Karcher mean (analogous origin for \mathcal{A}_r). This is subsequently composed with right group actions defined by normal coordinates over S_{++}^2 , fixed average length scales \bar{M} , explicit or inferred parametrizations $M(\ell)$, or some combination thereof to offer a complete representation of shapes.

The right-hand panel of Fig. 8 shows a 2D marginal slice of the airfoil data projected onto the 4D PGA basis \mathcal{T} —i.e., a discrete distribution of $\mathbf{t} \in T_{[\bar{X}_0]} \mathcal{A}_4$. Note that this design space roughly resembles a mixture of overlapping Gaussian distributions across the diverse family of airfoils. Compared to the CST representation, such a design space is significantly easier to infer or represent in the context of AI and ML algorithms. Further, extrapolation to shapes beyond the point cloud is significantly less volatile in this frame-

work, offering an improved notion of regularized shape deformations.

To demonstrate the improved regularization of deformations, Fig. 9 shows four random corner-to-corner sweeps (enveloping the data by bounding hyperrectangles) through CST and PGA spaces. The PGA perturbations are constructed with fixed average scales \bar{M} defining right inverse $\bar{X}(\mathbf{t})\bar{M}$ for $\mathbf{t} \in T_{[\bar{X}_0]} \mathcal{A}_4$. In CST space, it is difficult to define a single design space that covers the range of airfoils under consideration while allowing for smooth deformations between them—i.e., named classes correspond to largely distinct subsets of perturbed CST coefficients, and linear interpolation across these classes results in nonphysical shapes with large undulations. Conversely, all shapes generated using the proposed Grassmannian methodology result in valid airfoil designs while creating a rich design space worthy of investigation. Moreover, this data-driven approach and regularization over (25) mitigates undesirable oscillations and undulations in the shape, as depicted in Fig. 9.

3.2. Consistent blade deformations

We emphasize a powerful approach to dimension reduction for parametrizing 3D shapes from 2D sequences of discrete shapes. Namely, we introduce the concept of “consistent deformations” via parallel translation over the inferred PGA domain. The deformations retain the original notion of parameter direction over the PGA domain and result in a more intuitive deformation of blade shape—only requiring 7–10 parameters in total to define rich deformations to the entire 3D blade shape.

As previously noted, current approaches to 3D blade design require significant hand-tuning of airfoils to ensure the construction of valid blade geometries without dimples or kinks. Quantitatively, this can result from poor interpolation of shape characteristics, which deform shapes differently from one cross-section to the next. Our proposed approach enables the flexible design of new blades by applying consistent deformations to all airfoils in addition to affine-invariant interpolation of shapes along the span. The undesirable variations from one cross-section to the next are mitigated by parametrizing consistent perturbations enabled by parallel translation.

Blade perturbations are constructed from deformations to each of the given cross-sectional airfoils in consistent directions over $\mathbf{t} \in T_{[\bar{X}_0]} \mathcal{A}_4$. Having defined perturbation directions in the tangent space of the Karcher mean $[\bar{X}_0]$, we utilize parallel transport as an isometry to smoothly translate the perturbing vector field of the shape along separate geodesics—connecting the Karcher mean to

each of the LA standardized airfoils ($[\tilde{X}_k]$). More generally, this could be computed for $\mathcal{G}(n, 2)$ and S_{++}^2 using Algorithms 5 and 8, respectively. The result is a set of consistent directions—as equal inner products and consequently equivalent normal coordinates in the central tangent space—over ordered tangent spaces $T_{[\tilde{X}_k]} \mathcal{G}(n, 2)$, centered on each of the nominal $[\tilde{X}_k]$ defining the blade (similarly for corresponding SPD matrices). An example of a consistently perturbed sequence of cross-sectional airfoils is shown in Fig. 6 (right-hand panel), composed with average length scales \bar{M} and $\Delta(\mathbf{t}) = \text{vec}^{-1}(T; \mathbf{t})$ as

$$(\pi^{-1} \circ [\tilde{X}]) (\eta_k; \mathbf{t}, \bar{M}) = \text{Exp}_{[\tilde{X}_k]} (\tau_{\parallel} (\Delta(\mathbf{t}); [\tilde{X}_0], [\tilde{X}_k])) \bar{M}. \quad (28)$$

These deformed shapes are then interpolated over (η_k) to construct various blade deformations with any level of refinement as wire-frames. Consequently, undulations in the blade can be consistently parametrized and regularized by a shared set of four parameters $\mathbf{t} \in T_{[\tilde{X}_0]} \mathcal{A}_4$ in the tangent space of the Karcher mean using parallel transport with Algorithm 5 (and/or Algorithm 8 for a chosen separable representation).

Utilizing interpolation over consistent deformations with three to six independent affine parameters ℓ , chosen to be spanwise constant, as $\tilde{X}(\eta; \mathbf{t})M(\eta; \ell)$, constitutes a full set of 7–10 parameters that describe a rich feature space of 3D blade perturbations—a significant reduction compared to alternative explicitly designed expansion representations. Of course, the more general representation (6) requires distinct parallel transport over the product manifold $\mathcal{A}_4 \times \mathcal{S}_3$ to offer additional flexibility with similar implications using only seven, or generally $r + 3$, total parameters. Additional generalizations can be made at the expense of introducing additional parameters—perhaps incorporating controlled spanwise variations in \mathbf{t} . However, in this most simplified context, a rich set of deformations can still be defined with very few parameters. The result is a framework with a flexible means for designers to balance the total number of parameters to achieve sought blade deformations using a more explainable and interpretable representation—generated by an explainable and interpretable non-linear manifold of shapes.

4. Conclusions

The benefits of coherent shape deformations, coupled with a natural framework for interpolating 2D airfoil shapes into 3D blades and the decoupling of affine and undulation-type deformations, make Grassmannian-based shape representation a powerful tool for enabling aerodynamic design. Moreover, the proposed transformations also enable the representation of rotation- and reflection-invariant shapes over a product submanifold for the purposes of 2D design.

Transforming discrete shapes into separable tensors enables 2D deformations and designs with evidence of improved regularization against nonphysical deformations. Moreover, the transformed representation offers more visually compelling evidence that Gaussian mixture models (a common prior in AI-aided design) may be a more relevant choice of prior distribution over the Karcher (Fréchet) centered domain of normal coordinates. Additionally, the ability to construct consistent deformations to blade shapes via parallel transport offers a novel and intuitive regularization to dramatically reduce the total number of parameters for 3D blade design.

We have shown, through theoretical arguments and numerical demonstration, that samples drawn from a class of relevant discrete airfoil shapes converge to Grassmannian elements defined

as discrete refinements with fixed reparametrization. Moreover, we have motivated continuous analogues built from the corresponding discrete shapes for future extensions of this work. Lastly, in contrast to AI-based generative shape models, our methods offer a fast and extremely lightweight approach to shape-manifold learning.

Acknowledgments

We would like to thank our collaborators: Ryan King, Ganesh Vijayakumar, Bum Seok Lee, and James Baeder for discussions about the design requirements and constraints in wind turbine blade design. We would also like to thank Andrew Dienstfrey for supporting discussions and providing constructive technical critiques emphasizing the strengths and weaknesses of the proposed methods.

This work was authored in part by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under contract no. DE-AC36-08GO28308. Funding partially provided by the Advanced Research Projects Agency-Energy (ARPA-E) Design Intelligence Fostering Formidable Energy Reduction and Enabling Novel Totally Impactful Advanced Technology Enhancements (DIFFERENTIATE) program, award no. 19\CJ000\07\03. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government. This work is U.S. Government work and not protected by U.S. copyright. A portion of this research was performed using computational resources sponsored by the Department of Energy's Office of Energy Efficiency and Renewable Energy and located at the National Renewable Energy Laboratory. Certain commercial equipment, instruments, or materials are identified in this paper in order to specify hardware utilized in numerical experiments. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology and/or the National Renewable Energy Laboratory, nor is it intended to imply that the materials or equipment identified are necessarily the best available for the purpose.

References

- Absil, P.-A., Mahony, R., & Sepulchre, R. (2004). Riemannian geometry of Grassmann manifolds with a view on algorithmic computation. *Acta Applicandae Mathematica*, **80**(2), 199–220.
- Absil, P.-A., Mahony, R., & Sepulchre, R. (2008). *Optimization algorithms on matrix manifolds*. Princeton University Press.
- Achour, G., Sung, W.J., Pinon-Fischer, O.J., & Mavris, D.N. (2020). Development of a conditional generative adversarial network for airfoil shape optimization. In *Proceedings of the AIAA Scitech 2020 Forum* (p. 2261).
- Arvanitidis, G., Hansen, L. K., & Hauberg, S. (2018). Latent space oddity: On the curvature of deep generative models. *International Conference on Learning Representations*.
- Bak, C., Zahle, F., Bitsche, R., Kim, T., Yde, A., Henriksen, L. C., Nataraajan, A., & Hansen, M. H. (2013). The DTU 10-MW reference wind turbine. In *DTU Wind Energy Report-I-0092* (Vol. 5).
- Bendokat, T., Zimmermann, R., & Absil, P.-A. (2020). A Grassmann manifold handbook: Basic geometry and computational aspects. arXiv preprint arXiv:2011.13699.
- Benner, P., Gugercin, S., & Willcox, K. (2015). A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM Review*, **57**(4), 483–531.
- Berrut, J.-P., & Trefethen, L. N. (2004). Barycentric Lagrange interpolation. *SIAM Review*, **46**(3), 501–517.

- Birkhoff, G., & De Boor, C. (1964). Error bounds for spline interpolation. *Journal of Mathematics and Mechanics*, **13**, 827–835.
- Bonnabel, S., & Sepulchre, R. (2010). Riemannian metric and geometric mean for positive semidefinite matrices of fixed rank. *SIAM Journal on Matrix Analysis and Applications*, **31**(3), 1055–1070.
- Bryner, D., Klassen, E., Le, H., & Srivastava, A. (2014). 2D affine and projective shape analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **36**(5), 998–1011.
- Chen, W., Chiu, K., & Fuge, M. (2019). Aerodynamic design optimization and shape exploration using generative adversarial networks. In *Proceedings of AIAA Scitech 2019 Forum*(p. 2351).
- Dogan, G., Bernal, J., & Hagwood, C. R. (2015). A fast algorithm for elastic shape distances between closed planar curves. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*(pp. 4222–4230).
- Doronina, O., Glaws, A., King, R., Vijayakumar, G., & Grey, Z. (2022). G2Aero (G2Aero: Separable shape tensors for aerodynamic applications). USDOE Advanced Research Projects Agency - Energy (ARPA-E). <https://doi.org/10.11578/dc.20220422.1>.
- Edelman, A., Arias, T. A., & Smith, S. T. (1998). The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, **20**(2), 303–353.
- Fletcher, P. T., & Joshi, S. (2004). Principal geodesic analysis on symmetric spaces: Statistics of diffusion tensors. In *Computer vision and mathematical methods in medical and biomedical image analysis*(pp. 87–98). Springer.
- Fletcher, P. T., Lu, C., & Joshi, S. (2003). Statistics of shape via principal geodesic analysis on Lie groups. In *Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*(Vol. 1, pp. I–I). IEEE.
- Gaertner, E., Rinker, J., Sethuraman, L., Zahle, F., Anderson, B., Barter, G., Abbas, N., Meng, F., Bortolotti, P., Skrzypinski, W., Scott, G., Feil, R., Bredmose, H., Dykes, K., Sheilds, M., Allen, C., & Viselli, A. (2020). IEA wind TCP task 37: Definition of the IEA 15-megawatt offshore reference wind turbine. Technical report. International Energy Agency.
- Gallivan, K., Srivastava, A., Liu, X., & Van Dooren, P. (2003). Efficient algorithms for inferences on Grassmann manifolds. In *Proceedings of the 2003 IEEE Workshop on Statistical Signal Processing*(pp. 315–318). IEEE.
- Ge, M., Zhang, H., Wu, Y., & Li, Y. (2019). Effects of leading edge defects on aerodynamic performance of the s809 airfoil. *Energy Conversion and Management*, **195**, 466–479.
- Geuzaine, C., & Remacle, J.-F. (2009). Gmsh: A 3-D finite element mesh generator with built-in pre-and post-processing facilities. *International Journal for Numerical Methods in Engineering*, **79**(11), 1309–1331.
- Ghorbani, H., & Khameneifar, F. (2021). Airfoil profile reconstruction from unorganized noisy point cloud data. *Journal of Computational Design and Engineering*, **8**(2), 740–755. <https://doi.org/10.1093/jcde/qwab011>.
- Glaws, A., Hokanson, J., King, R., & Vijayakumar, G. (2022a). Regularizing invertible neural networks for airfoil design through dimension reduction. In *Proceedings of the AIAA SCITECH 2022 Forum*(p. 1098).
- Glaws, A., King, R. N., Vijayakumar, G., & Ananthan, S. (2022b). Invertible neural networks for airfoil design. *AIAA Journal*, **60**, 3035–3047.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, **63**(11), 139–144. <https://doi.org/10.1145/3422622>.
- Gower, J. C. (1975). Generalized procrustes analysis. *Psychometrika*, **40**(1), 33–51.
- Grey, Z. J. (2019). *Active manifold-geodesics: A Riemannian view on active subspaces with shape sensitivity applications*. Ph.D. Thesis. University of Colorado at Boulder.
- Grey, Z. J., & Constantine, P. G. (2018). Active subspaces of airfoil shape parameterizations. *AIAA Journal*, **56**(5), 2003–2017.
- Hagwood, C., Bernal, J., Halter, M., Elliott, J., & Brennan, T. (2013). Testing equality of cell populations based on shape and geodesic distance. *IEEE Transactions on Medical Imaging*, **32**(12), 2230–2237.
- Hall, C. A., & Meyer, W. W. (1976). Optimal error bounds for cubic spline interpolation. *Journal of Approximation Theory*, **16**(2), 105–122.
- Higham, N. J. (2004). The numerical stability of Barycentric Lagrange interpolation. *IMA Journal of Numerical Analysis*, **24**(4), 547–556.
- Hosseini, S. F., & Moetakef-Imani, B. (2016). Innovative approach to computer-aided design of horizontal axis wind turbine blades. *Journal of Computational Design and Engineering*, **4**(2), 98–105. <https://doi.org/10.1016/j.jcde.2016.11.001>.
- Hyvärinen, A., & Oja, E. (2000). Independent component analysis: Algorithms and applications. *Neural Networks*, **13**(4–5), 411–430.
- Jonkman, J., Butterfield, S., Musial, W., & Scott, G. (2009). *Definition of a 5-MW reference wind turbine for offshore system development*. Technical report (NREL/TP-500-38060). NREL.
- Joshi, S. H., Klassen, E., Srivastava, A., & Jermyn, I. (2007). A novel representation for Riemannian analysis of elastic curves in Rn. In *Proceedings of the 2007 IEEE Conference on Computer Vision and Pattern Recognition*(pp. 1–7). IEEE.
- Kendall, D. G. (1989). A survey of the statistical theory of shape. *Statistical Science*, **4**(2), 87–99. <https://doi.org/10.1214/ss/1177012582>.
- Kendall, D. G., Barden, D., Carne, T. K., & Le, H. (2009). *Shape and shape theory*. John Wiley & Sons.
- Kingma, D. P., & Welling, M. (2013). Auto-encoding variational Bayes. arXiv preprint arXiv:1312.6114.
- Klassen, E., Srivastava, A., Mio, M., & Joshi, S. H. (2004). Analysis of planar shapes using geodesic paths on shape spaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **26**(3), 372–383.
- Kramer, M. A. (1991). Nonlinear principal component analysis using autoassociative neural networks. *AIChE Journal*, **37**(2), 233–243.
- Kulfan, B. M. (2008). Universal parametric geometry representation method. *Journal of Aircraft*, **45**(1), 142–158.
- Lee, J. M. (2006). *Riemannian manifolds: An introduction to curvature* (Vol. 176). Springer Science & Business Media.
- Li, J., Bouhlel, M. A., & Martins, J. R. R. A. (2019). Data-based approach for fast airfoil analysis and optimization. *AIAA Journal*, **57**(2), 581–596.
- Michor, M., & Mumford, M. (2006). Riemannian geometries on spaces of plane curves. *Journal of the European Mathematical Society*, **8**, 1–48.
- Pennec, X. (1999). Probabilities and statistics on Riemannian manifolds: Basic tools for geometric measurements. In *Proceedings of Nonlinear Signal and Image Processing*(Vol. 3, pp. 194–198). IEEE-EURASIP.
- Pennec, X., Fillard, P., & Ayache, N. (2006). A Riemannian framework for tensor computing. *International Journal of Computer Vision*, **66**(1), 41–66.
- Sapidis, N. S. (1994). *Designing fair curves and surfaces: Shape quality in geometric modeling and computer-aided design*. SIAM.
- Schulz, V. H. (2014). A Riemannian view on shape optimization. *The Journal of the Society for the Foundations of Computational Mathematics*, **14**, 483–501.
- Seshadri, P., Shahpar, S., Constantine, P., Parks, G., & Adams, M. (2018). Turbomachinery active subspace performance maps. *Journal*

- of Turbomachinery, **140**(4), 041003. <https://doi.org/10.1115/1.4038839>.
- Slotnick, J. P., Khodadoust, A., Alonso, J., Darmofal, D., Gropp, W., Lurie, E., & Mavriplis, D. J. (2014). CFD vision 2030 study: A path to revolutionary computational aerosciences. Technical report. NASA.
- Sra, S., & Hosseini, R. (2015). Conic geometric optimization on the manifold of positive definite matrices. *SIAM Journal on Optimization*, **25**(1), 713–739.
- Strubell, E., Ganesh, A., & McCallum, A. (2019). Energy and policy considerations for deep learning in NLP. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 3645–3650. Florence, Italy: Association for Computational Linguistics. <https://doi.org/10.18653/v1/P19-1355>.
- Townsend, A., & Trefethen, L. N. (2015). Continuous analogues of matrix factorizations. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **471**(2173), 20140585.
- Trefethen, L. N. (2010). Householder triangularization of a quasimatrix. *IMA Journal of Numerical Analysis*, **30**(4), 887–897.
- UIUC Applied Aerodynamics Group (2022). UIUC Airfoil Coordinates Database. https://m-selig.ae.illinois.edu/ads/coord_database.html.
- Wang, J., Li, R., He, C., Chen, H., Cheng, R., Zhai, C., & Zhang, M. (2022). An inverse design method for supercritical airfoil based on conditional generative models. *Chinese Journal of Aeronautics*, **35**(3), 62–74. <https://doi.org/10.1016/j.cja.2021.03.006>.
- Welker, K. (2021). Suitable spaces for shape optimization. *Applied Mathematics & Optimization*, **84**(1), 869–902.
- Willcox, K., & Peraire, J. (2002). Balanced model reduction via the proper orthogonal decomposition. *AIAA Journal*, **40**(11), 2323–2330.
- Yair, O., Ben-Chen, M., & Talmon, R. (2019). Parallel transport on the cone manifold of SPD matrices for domain adaptation. *IEEE Transactions on Signal Processing*, **67**(7), 1797–1811.
- Yang, S., Lee, S., & Yee, K. (2022). Inverse design optimization framework via a two-step deep learning approach: Application to a wind turbine airfoil. *Engineering with Computers*, 1–17. <https://doi.org/10.1007/s00366-022-01617-6>.
- Yonekura, K., & Suzuki, K. (2021). Data-driven design exploration method using conditional variational autoencoder for airfoil design. *Structural and Multidisciplinary Optimization*, **64**, 613–624.
- Zhang, Y., Sung, W. J., & Mavris, D. N. (2018). Application of convolutional neural network to predict airfoil lift coefficient. In *Proceedings of the 2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*(p. 1903).
- Zimmermann, R. (2019). Manifold interpolation and model reduction. arXiv preprint arXiv:1902.06502.