Modeling and optimization with Gaussian processes in reduced eigenbases

David Gaudrie1,2 · Rodolphe Le Riche1,3 · Victor Picheny4 · Benoît Enaux2 · Vincent Herbert2

Received: 13 September 2019 / Revised: 18 November 2019 / Accepted: 20 November 2019 / Published online: 29 February 2020
© Springer-Verlag GmbH Germany, part of Springer Nature 2020

Abstract
Parametric shape optimization aims at minimizing an objective function \( f(x) \) where \( x \) are CAD parameters. This task is difficult when \( f(·) \) is the output of an expensive-to-evaluate numerical simulator and the number of CAD parameters is large. Most often, the set of all considered CAD shapes resides in a manifold of lower effective dimension in which it is preferable to build the surrogate model and perform the optimization. In this work, we uncover the manifold through a high-dimensional shape mapping and build a new coordinate system made of eigenshapes. The surrogate model is learned in the space of eigenshapes: a regularized likelihood maximization provides the most relevant dimensions for the output. The final surrogate model is detailed (anisotropic) with respect to the most sensitive eigenshapes and rough (isotropic) in the remaining dimensions. Last, the optimization is carried out with a focus on the critical dimensions, the remaining ones being coarsely optimized through a random embedding and the manifold being accounted for through a replication strategy. At low budgets, the methodology leads to a more accurate model and a faster optimization than the classical approach of directly working with the CAD parameters.

Keywords
Dimension reduction · Principal component analysis · Parametric shape optimization · Gaussian processes · Surrogate-based optimization

1 Introduction

The most frequent approach to shape optimization is to describe the shape by a vector of \( d \) computer-aided design (CAD) parameters, \( x \in X \subset \mathbb{R}^d \) and to search for the parameters that minimize an objective function, \( x^* = \arg \min_{x \in X} f(x) \). In the CAD modeling process, the set of all possible shapes has been reduced to a space of parameterized shapes, \( \Omega := \{\Omega_x, x \in X\} \).

It is common for \( d \) to be large, \( d \gtrsim 50 \). Optimization in such a high-dimensional design space is difficult, especially when \( f(·) \) is the output of a high fidelity numerical simulator that can only be run a restricted number of times (Shan and Wang 2010). In computational fluid dynamics for example, simulations easily take 12 to 24 h and evaluation budgets range between 100 and 200 calls. Surrogate-based approaches (Sacks et al. 1989; Forrester and Keane 2009) have proven their effectiveness to tackle optimization problems in a few calls to \( f(·) \). They rely on a surrogate model (or metamodel, e.g., Gaussian processes Stein 1999; Cressie 1992; Rasmussen and Williams 2006) built upon \( n \) past observations of \( y_i = f(x^{(i)}) \). For a Gaussian process (GP, Stein 1999; Cressie 1992; Rasmussen and Williams 2006), given \( D_n = \{(x^{(1)}, y_1), \ldots, (x^{(n)}, y_n)\} = \{x^{(1:n)}, y_{1:n}\} \), \( f(·) \) can be predicted in closed form at any untested point \( x^{\text{new}} \in X \) via the kriging mean predictor, \( m(x^{\text{new}}) \). The probabilistic framework of GPs additionally provides the uncertainty associated with the prediction, known as the kriging variance, \( s^2(x^{\text{new}}) \), also computable in closed form (Rasmussen and Williams 2006). For the optimization, the metamodel’s prediction and uncertainty are mixed by an acquisition function such as the expected improvement (Mockus 1975) to decide which design \( x^{(n+1)} \) should be evaluated next. However, such techniques suffer
from the curse of dimensionality (Bellman 1961) when \( d \) is large. The budget is also typically too narrow to perform sensitivity analysis (Saltelli et al. 2004) and select variables prior to optimizing. A further issue is that the CAD parameters \( x \) commonly have heterogeneous impacts on the shapes \( \Omega \): many of them are intended to refine the shape locally whereas others have a global influence so that shapes of practical interest involve interactions between all the parameters.

Most often, the set of all CAD generated shapes, \( \Omega \), can be approximated in a \( \delta \)-dimensional manifold, \( \delta < d \). In Raghavan et al. (2013, 2014), this manifold is accessed through an auxiliary description of the shape, \( \phi(\Omega) \), \( \phi \) being either its characteristic function or the signed distance to its contour. The authors aim at minimizing an objective function using diffuse approximation and gradient-based techniques, while staying on the manifold of admissible shapes. Active Shape Models (Cootes et al. 1995) provide another way to handle shapes in which the contour is discretized (Stegmann and Gomez 2002; Wang 2012).

Building a surrogate model in reduced dimension can be performed in different ways. The simplest is to restrict the metamodel to the most influential variables. But typical evaluation budgets are too narrow to find these variables before the optimization. Moreover, correlations might exist among the original dimensions (here CAD parameters) so that a selection of few variables may not constitute a valid reduced order description and meta-variables may be more appropriate. In Wu et al. (2019), the high-dimensional input space is circumvented by decomposing the model into a series of low-dimensional models after an ANOVA procedure. In Bouhlel et al. (2016), a kriging model is built in the space of the first Partial Least Squares axes for emphasizing the most relevant directions. Related approaches for dimensionality reduction inside GPs consist in a projection of the input \( x \) on a lower dimensional hyperplane spanned by orthogonal vectors. These vectors are determined in different manners, e.g., by searching the active space in Constantine et al. (2014) and Li et al. (2019), or during the hyperparameters estimation in Tripathy et al. (2016). A more detailed bibliography of dimension reduction in GPs is conducted in Section 3.

For optimization purposes, the modes of discretized shapes (Stegmann and Gomez 2002) are integrated in a surrogate model in Li et al. (2018). In Cinquegrana and Iuliano (2018), the optimization is carried out on the most relevant modes using evolutionary algorithms combined with an adaptive adjustment of the bounds of the design space, also employed in Shan and Wang (2004).

Following the same route, in Section 2, we retrieve a shape manifold with dimension \( \delta < d \). Our approach is based on a principal component analysis (PCA, Wall et al. 2003) of shapes described in an ad hoc manner in the same vein as Cinquegrana and Iuliano (2018) and Li et al. (2018) but it provides a new investigation of the best way to characterize shapes. Section 3 is devoted to
the construction of a kriging surrogate model in reduced dimension. Contrarily to Li et al. (2018) and Li et al. (2019), the least important dimensions are still accounted for. A regularized likelihood approach is employed for dimension selection, instead of the linear PLS method (Bouhlel et al. 2016). In Section 4, we employ the metamodel to perform global optimization (Jones et al. 1998) via the maximization of the expected improvement (Mockus 1975). A reduction of the space dimension is achieved through a random embedding technique (Wang et al. 2013) and a pre-image problem is solved to keep the correspondence between the eigenshapes and the CAD parameters. The proposed method is summarized in Fig. 1.

2 From CAD description to shape eigenbasis

CAD parameters are usually set up by engineers to automate shape generation. These parameters may be Bézier or Spline control points which locally readjust the shape. Other CAD parameters, such as the overall width or the length of a component, have a more global impact on the shape. While these parameters are intuitive to a designer, they are not chosen to achieve any specific mathematical property and in particular do not let themselves interpret to reduce dimensionality.

In order to define a better behaved description of the shapes that will help in reducing dimensionality, we exploit the fact that the time to generate a shape $\Omega_x$ is negligible in comparison with the evaluation time of $f(x)$.

In the spirit of kernel methods (Vapnik 1995; Schölkopf et al. 1997), we analyze the designs $x$ in a high-dimensional feature space $\Phi \subset \mathbb{R}^D$, $D \gg d$ (potentially infinite dimensional) that is defined via a mapping $\phi(x), \phi : X \rightarrow \Phi$. With an appropriate $\phi(\cdot)$, it is possible to distinguish a lower dimensional manifold embedded in $\Phi$. As we deal with shapes, natural candidates for $\phi(\cdot)$ are shape representations.

This paper is motivated by parametric shape optimization problems. However, the approaches developed for metamodelling and optimization are generic and extend to any situation where a pre-existing collection of designs $\{x^{(1)}, \ldots, x^{(N)}\}$ and a fast auxiliary mapping $\phi(x)$ exist. $\phi(x) = x$ is a possible case. If $x$ are parameters that generate a signal, another example would be $\phi(x)$, the discretized times series.

2.1 PCA of shape representations

In the literature, shapes have been described in different ways. Common mappings $\phi(\cdot)$ are the characteristic function of a shape $\Omega_x$ (Raghavan et al. 2013) or the signed distance to its contour $\partial \Omega_x$ (Raghavan et al. 2014). In the Point Distribution Model (Cootes et al. 1995; Stegmann and Gomez 2002), $\partial \Omega_x$ is discretized at $D/k$ locations $s^{(i)} \in \partial \Omega_x \subset \mathbb{R}^k$ ($k = 2$ or 3), and $\phi(x) = (s^{(1)} \top, \ldots, s^{(D/k)} \top) \top \in \mathbb{R}^D$.

During the step 1 of our method (see Fig. 1), a large number $(N)$ of plausible designs $x^{(i)} \in X$ is mapped to $\Phi \subset \mathbb{R}^D$ and build the matrix $\Phi \in \mathbb{R}^{N \times D}$ which contains the $\phi(x^{(i)}) \in \mathbb{R}^D$ in rows and whose column-wise mean is $\overline{\phi} \in \mathbb{R}^D$. In the absence of a set of relevant $x^{(i)}$s, these designs can be sampled from an a priori distribution, typically a uniform distribution. Next (step 2 in Fig. 1), we perform a principal component analysis (PCA) on $\Phi$: correlations are sought between the $\phi(x)^{(i)}$’s, $j = 1, \ldots, D$. The eigenvectors of the empirical covariance matrix $C_\Phi := \frac{1}{N}(\Phi - 1_N \overline{\phi}) \top (\Phi - 1_N \overline{\phi})$, written $v^j \in \mathbb{R}^D$, form an ordered orthonormal basis of $\Phi$ with decreasing importance as measured by the PCA eigenvalues $\lambda_j, j = 1, \ldots, D$. They correspond to orthonormal directions in $\Phi$ that explain the most the dispersion of the high-dimensional representations of the shapes, $\phi(x^{(i)})$. Any design $x$ can now be expressed in the eigenbasis $V := \{v^1, \ldots, v^D\}$ since

$$\phi(x) = \overline{\phi} + \sum_{j=1}^{D} \alpha_j v^j \tag{1}$$

where $(\alpha_1, \ldots, \alpha_D) \top =: \alpha = V \top (\phi(x) - \overline{\phi})$ are the coordinates in $V$ (principal components), and $V := (v^1, \ldots, v^D) \in \mathbb{R}^{D \times D}$ is the matrix of eigenvectors (principal axes). $\alpha_j$ is the deviation from the mean shape $\overline{\phi}$, in the direction of the eigenvector $v^j$. The $\alpha^{(i)}$s form a manifold $A_N := \{\alpha^{(1)}, \ldots, \alpha^{(N)}\}$ which approximates the true $\alpha$ manifold, $A := \{\alpha \in \mathbb{R}^D : \exists x \in X, \alpha = V \top (\phi(x) - \overline{\phi})\}$. Even though $A_N \subset \mathbb{R}^D$, it is often a manifold of lower dimension, $\delta \ll D$. In experiments reported in Gaudrie et al. (2019), we have generated shapes of known true dimensionality and applied the previous methodology. We have then investigated slightly more complex and realistic shapes described with many parameters but that mainly depend on few intrinsic dimensions. The analysis of the eigenvalues $\lambda_j$ and of the eigenvectors $v^j$ has revealed that the discretization of the contour was the only mapping for which $E(d) := \sum_{j=1}^{D} \lambda_j / \sum_{j=1}^{D} \lambda_j = 1$: the true dimension is retrieved with this $\phi(\cdot)$ while more components are necessary with the characteristic function or the signed distance. Additionally, with the discretization, the first eigenvectors look like shapes on their own, that we will call eigenshapes in the following, and ease interpretations.

2.1.1 Link with kernel PCA

$N$ designs $x^{(i)} \in \mathbb{R}^d$ have been mapped to a high-dimensional feature space $\Phi \subset \mathbb{R}^D$ in which PCA was carried out. This is precisely the task that is performed in
Kernel PCA (Schölkopf et al. 1997), a nonlinear dimension reduction technique (contrarily to PCA which seeks linear directions in \( \mathbb{R}^d \)). KPCA aims at finding a linear description of the data in a feature space \( \Phi \), by applying a PCA to nonlinearly mapped \( \phi(x^{(i)}) \in \Phi \). The difference with our approach is that the mapping \( \phi(\cdot) \) and the feature space \( \Phi \) are usually unknown in KPCA, since \( \phi(x) \) may live in a very high-dimensional or even infinite dimensional space in which dot products cannot be computed efficiently. Instead, dot products are computed using designs in the original space \( X \) via a kernel which should not be mistaken with the kernel of GPs, \( k_\phi : X \times X \to \mathbb{R}, k_\phi(x, x') = \langle \phi(x), \phi(x') \rangle_\phi \) (this is called the “kernel-trick” Vapnik 1995; Schölkopf et al. 1997). The eigencomponents of the points after mapping, \( \alpha^{(i)}_j = v_j^\top (\phi(x^{(i)}) - \overline{\phi}) \), can be recovered from the eigenanalysis of the \( N \times N \) Gram matrix \( K \) with \( K_{ij} = k_\phi(x^{(i)}, x^{(j)}) \) (see Schölkopf et al. 1997; Wang 2012 for algebraic details). Finding which original variables in \( x \) correspond to a given \( v^j \) is not straightforward and requires the resolution of a pre-image problem (Mika et al. 1999; Wang 2012).

Having a shape-related and computable \( \phi(\cdot) \) avoids these ruses and makes the principal axes \( v^j \) directly meaningful. It is further possible to give the expression of the equivalent kernel in our approach, in terms of the mapping \( \phi(\cdot) \), from the polarization identity. By definition of the (centered) high dimensional mapping to \( \Phi, x \mapsto \phi(x) - \overline{\phi} \),

\[
\|\phi(x) - \overline{\phi}\|_{\mathbb{R}^D}^2 = \langle (\phi(x) - \overline{\phi}) - (\phi(x') - \overline{\phi}), (\phi(x) - \overline{\phi}) - (\phi(x') - \overline{\phi}) \rangle_{\mathbb{R}^D} \\
= \|\phi(x) - \overline{\phi}\|_{\mathbb{R}^D}^2 + \|\phi(x') - \overline{\phi}\|_{\mathbb{R}^D}^2 - 2\langle (\phi(x) - \overline{\phi}), (\phi(x') - \overline{\phi}) \rangle_{\mathbb{R}^D}
\]

hence,

\[
k_\phi(x, x') = \frac{1}{2}(\|\phi(x) - \overline{\phi}\|_{\mathbb{R}^D}^2 + \|\phi(x') - \overline{\phi}\|_{\mathbb{R}^D}^2 - 2\langle (\phi(x) - \overline{\phi}), (\phi(x') - \overline{\phi}) \rangle_{\mathbb{R}^D}).
\]

Logically, \( k_\phi(\cdot, \cdot) \), a similarity measure between designs, is negatively proportional to the distance between the shape representations. Because of the size of the eigenanalyses to be performed, kernel PCA is advantageous over a mapping followed by a PCA when \( D > N \), i.e., when the shapes have a very high resolution, and vice versa. In the current work where \( \phi(\cdot) \) is known and \( D \) is smaller than 1000, we will follow the mapping plus PCA approach.

### 2.2 Experiments

Three examples out of Gaudrie et al. (2019) are detailed here and will be investigated in Sections 3 and 4, when turning to metamodelling and optimization. Unless stated otherwise, the database \( \Phi \) is made of \( N = 5000 \) designs sampled uniformly in \( X \).

**Example** A rectangle ABCD with \( x \in \mathbb{R}^{40} \) whose parameters \( x_1 \) and \( x_2 \) are the location of A, \( x_3 \) and \( x_4 \) are the width and the height of ABCD, and \( x_{5:13}, x_{14:22}, x_{23:31} \) and \( x_{32:40} \) are small evenly distributed perturbations, on the AB, BC, CD, and DA segments, respectively.

\( x_1, \ldots, x_4 \) are of a magnitude larger than the other parameters to ensure a close-to-rectangular shape, as shown in the left part of Fig. 2.

In this example where 4 parameters (position and sizes) mainly explain the differences among shapes, a reconstruction quality of \( E(4) = 0.9983 \) is attained with the 4 first eigenvectors \( v^1, v^2 \) and \( v^3 \), the most influencing eigenshapes plotted in black and blue act as translations, while \( v^3 \) and \( v^4 \) (in red and green) correspond to widening and heightening of the rectangle. The reconstruction accuracy varies according to the number of selected modes, e.g., \( E(2) = 0.9742 \) and \( E(16) = 0.9991 \). The fluctuations along the segments appear from the 5th eigenshape on. Any shape is retrieved with the \( d = 40 \) first eigenshapes which corresponds to the total number of parameters as \( E(40) = 1 \).

**Example** A straight line joining two fixed points A and B, modified by smooth perturbations \( r \in \mathbb{R}^{29} \), evenly distributed along [AB] to approximate a smooth curve.

This example is inspired by the catenoid problem (Colding and Minicozzi 2006). The perturbations \( r \) are generated by a Gaussian process with squared exponential kernel and with length-scale 6 times smaller than [AB]. Therefore, in this example, the \( N = 5000 \) \( r^{(i)} \)'s used for building \( \Phi \) are not uniformly distributed in \( X \). Some curves are shown in the left part of Fig. 3.

The initial dimension \( (d = 29) \) is recovered by looking at the strictly positive eigenvalues \( E(29) = 1 \). Furthermore,
Fig. 2  Left: rectangle with varying position, size, and deformation of its sides. Right: 6 first eigenshapes (in the order black, blue, red, green, yellow, purple) of the rectangles

the manifold is found to mainly lie in a lower dimensional space: $\mathcal{A}_N$ can be approximated in $\delta = 7$ dimensions because $E(7) = 0.99975$. Four eigenvectors lead to a slightly degraded but still good reconstruction, $E(4) = 0.97973$. The right part of Fig. 3 shows the corresponding eigenshapes. These are similar to the ordered modes of the harmonic series with the associated eigenvalues ordered as the inverse of the frequencies.

Example  A modified NACA airfoil which is parameterized by $d = 22$ parameters: $\mathbf{x} = (M, P, T, L_1, \ldots, L_{19})^T \in \mathbb{R}^{22}$ where $M$, $P$, and $T$ are the standard NACA parameters, and the $L_i$s are small bumps along the airfoil. Nine such shapes are shown at the top of Fig. 4. More details can be found in Gaudrie et al. (2019).

Fig. 3  Left: a straight line joining two points, modified by the perturbations $r_j$ to approximate a curve. Gray: the line joining A and B. Blue, red, yellow, and green curve: examples of lines with regular $r_j$ perturbations. Red envelope: boundaries for the $r_j$’s. Right: 7 first eigenshapes

The 6 first eigenshapes (bottom left) as well as the $\mathcal{A}_N$ manifold (bottom right) are depicted in Fig. 4. The eigenvectors can be interpreted as a reformulation of the CAD parameters. The first eigenshape (blue) is a symmetric airfoil. Multiplying it by a coefficient (after adding it to the black mean shape) will increase or decrease the thickness of the airfoil; hence, it plays a similar role to the $T$ parameter. The second eigenshape is a cambered airfoil, whose role is similar to $M$ (maximum camber). Last, the third airfoil, which has a much smaller eigenvalue $\lambda_3$, is very thin, positive in the first part of the airfoil, and negative in its second part. It balances the camber of the airfoil towards the leading edge or towards the rear and plays a role similar to $P$, the position of the maximum camber. Small details that act on the airfoil such as the bumps only appear from the
4th eigenshape on. Not taking them into account leads to a weaker reconstruction, as shown in the right part of Fig. 5.

The analysis of $\mathcal{A}_N$ (Fig. 4, bottom right) is physically meaningful: even though $\mathbf{x}(i)$ are sampled uniformly in $X$, $\mathcal{A}_N$ resembles a pyramid in the $(v_1, v_2, v_3)$ basis. Designs with minimal $\alpha_2$ share the same $\alpha_3$ value. Since negative $\alpha_2$s correspond to wings with little camber, the position of this maximum camber has very little impact, hence the almost null $\alpha_3$ value. By looking at $\mathcal{A}_N$, it is learned that the parameter $P$ does not matter when $M$ is small, which is intuitive but is not expressed by the $(M, P, T)$ coordinates. Distances in $\mathcal{A}_N$ are therefore more representative of shape differences. An additional advantage of analyzing shapes is that correlations in the space of parameters (such as the...
one between \( M \) and \( P \) in this example) are discovered and removed, since \( \mathcal{V} \) is an orthonormal basis.

The reconstruction quality increases with the number of modes. Indeed, when looking at specimen of NACA 22 airfoils (top of Fig. 4), less than 22 dimensions are expected to be necessary to retrieve an approximation of sufficient quality. Small details that act on the airfoil such as the bumps only appear from the 4th eigenshape on. Not taking them into account leads to a weaker reconstruction, as shown in the right part of Fig. 5.

According to these experiments and to the supplementary ones in Gaudrie et al. (2019), the eigenvectors \( \psi_j, j \in \{d + 1, \ldots, D\} \), can already be discarded without even considering the values of the associated objective functions since the \( d \) first shape modes explain the whole variability of the discretized shapes. In practice, to filter numerical noise and to remove non-informative modes in shapes that are truly over-parameterized, we only consider the \( d' \) first eigenshapes, \( d' := \min(d, \tilde{d}) \) where \( \tilde{d} \) corresponds to the smallest number of axes that explain more than a given level of diversity in \( \Phi \) (e.g., 99.9, 99.95, or 99.99%), measured by \( E(\tilde{d}) \). Another alternative is to define \( \tilde{d} \) according to the dimensions for which \( \lambda_j/\lambda_1 \) is smaller than a prescribed threshold (e.g., 1/1000). Even though the notation \( D \) is kept, the eigenvectors \( \psi_j \) and the principal components \( \alpha_j \) are considered to be null \( \forall j > d' \) so that in fact \( D = d' \) in the following.

### 3 GP models for reduced eigenspaces

Building a surrogate model in the space of principal components has already been investigated in the context of reduced order models (Berkooz et al. 1993). In most applications, the dimension reduction is carried out in the output space, which has large dimension when it corresponds to values on a finite element mesh. The response is approximated by a linear combination of a small number of modes, and the metamodel is a function of the modes coefficients. The construction of surrogates with inherent dimensionality reduction has also been considered. In the active subspace method (Constantine et al. 2014), the dimension reduction comes from a linear combination of the inputs which is carried out by projecting \( x \) onto the hyperplane spanned by the directions of largest \( \nabla f(x) \) variation. The reduced-dimension GP is then \( Y(W^T \mathbf{x}) \) with \( W \in \mathbb{R}^{d \times \delta} \) containing these directions in columns. In Palar and Shimoyama (2018), cross-validation is employed for choosing the number of such axes. An application to airfoils is given in Li et al. (2019) where the authors take the directions of largest drag and lift gradients as columns of \( W \), even though this basis is no longer orthogonal. Another related technique with a \( Y(W^T x) \) GP which does not require the knowledge of \( \nabla f(x) \) is the kriging and partial least squares (KPLS) method (Bouhlel et al. 2016), where \( x \) is projected onto the hyperplane spanned by the first \( \delta \) axes of a PLS regression (Frank and Friedman 1993). The dimension reduction is output-driven but \( W \) is no longer orthogonal, and information may be lost when \( n < d' \) because any shape (of effective dimension \( d' \)) cannot be exactly reconstructed (1) with these \( n \) vectors. Coordinates in the PLS space are therefore incomplete and metamodelling loses precision when \( n \) is too small. In the same spirit, a double maximum-likelihood procedure is developed in Tripathy et al. (2016) to build an output-related and orthogonal matrix \( W \) for the construction of a Gaussian process with built-in dimensionality reduction. Rotating the design space through hyperparameters determined by maximum likelihood is also performed in (Namura et al. 2017). Table 1 summarizes the existing literature for building such GPs as well as the approach introduced in Section 3.1.2 (last column).

Instead of the space of CAD parameters \( x \), we reduce the dimension of the input space by building the surrogate with information from the space of shape representations, \( \Phi \), as in Li et al. (2018). To circumvent the high dimensionality of \( \Phi \subset \mathbb{R}^D \), a linear dimension reduction of \( \phi(x) \) is achieved by building the model in the space spanned by \( W^T \phi(x) \). A natural candidate for \( W \) is a restriction to few columns (eigenshapes) of the matrix \( \mathbf{V} \). Notice that contrarily to the other dimension reduction techniques which operate a linear dimension reduction of \( x \), this approach is nonlinear in \( x \).

| Model               | \( Y(W^T \mathbf{x}) \)          | \( Y(W^T \phi(x)) \)          | \( Y^\mathcal{Q}(W_a^T \phi(x)) + Y^\mathcal{T}(W_d^T \phi(x)) \) |
|---------------------|---------------------------------|--------------------------------|-------------------------------------------------|
| Dimension reduction | Linear in \( x \)               | Nonlinear in \( x \)           | Nonlinear in \( x \); group-additive model       |
| Construction of \( W \) | Active subspaces (Constantine et al. 2014; Palar and Shimoyama 2018; Li et al. 2019) PLS (Bouhlel et al. 2016) GP hyperparameters (Tripathy et al. 2016) and (Namura et al. 2017) Sensitivity analysis (Ben Salem et al. 2019) | PLS (Li et al. 2018) | Selection of mapped variables through penalized likelihood (Section 3.1.1) |
since it operates linearly on the nonlinear transformation $\phi(x)$. Also, it operates on a better suited representation of the designs, their shapes, instead of their parameters.

A first idea to reduce the dimension of the problem is to conserve the $\delta$ first eigenvectors $v^j$ according to some reconstruction quality criterion measured by the eigenvalues. The surrogate model is implemented in the space of the $\delta$ first principal components as

$$Y(\alpha_{1,\delta}) = Y(V_{1,\delta}^T \phi(x) - \phi_\delta).$$

Using a stationary kernel for the $Y(\alpha_{1,\delta})$ GP, i.e.,

$$k(\alpha_{1,\delta}, \alpha'_{1,\delta}) = \tilde{k}(||\alpha_{1,\delta} - \alpha'_{1,\delta}||_2),$$

the correlation between designs is $k(\alpha_{1,\delta}, \alpha'_{1,\delta}) = \tilde{k}(||V_{1,\delta}^T (\phi(x) - \phi(x'))||_2) = \tilde{k}(r)$ with $r^2 = (\phi(x) - \phi(x'))^TM(\phi(x) - \phi(x'))$ where $M = V_{1,\delta}V_{1,\delta}^T$ is a $D \times D$ matrix with low rank $(\delta)$.

Hence, this model implements a Gaussian process in the $\Phi$ space with an integrated linear dimensionality reduction step (Rasmussen and Williams 2006). Note that the kernel is non-stationary in the original $X$ space.

The approaches (Constantine et al. 2014; Bouhlel et al. 2016; Tripathy et al. 2016) mainly differ from that proposed in (3) in the construction of the reduced basis: in (3), dimension reduction is carried out without the need to call the expensive $f(x)$ (or its gradient): the directions of largest variation of an easy to compute mapping $f(\cdot)$ are used instead. This also prevents from a spurious or incomplete projection when $n$ is smaller than $D$ and avoids recomputing the basis at each iteration.

This is nonetheless a limitation since the $Y(\alpha_{1,\delta})$ approach relies only on considerations about the shape geometry. The output $y$ is not taken into account for the dimension reduction even though some $v^j$, $j \in \{1, \ldots, \delta\}$ may influence $y$ or not. Two shapes which differ in the $v^j$ components with $j \leq \delta$ may behave similarly in terms of output $y$, so that further dimension reduction is possible. Vice versa, eigencomponents that have a small geometrical effect and were neglected may be reintroduced because they matter for $y$.

As an illustration, consider the red and black shapes of Fig. 6. Both are associated with parameters $x$ and $x'$ and their discretizations $\phi(x)$ and $\phi(x')$ are quite different. Depending on the objective function, $f(x)$ and $f(x')$ might differ widely. However, when considering the $\tilde{\phi} + \sum_{j=1}^\delta \alpha_j v^j$ reconstruction with $\delta = 3$, they look very similar because $\alpha_{1,3} \approx \alpha'_{1,3}$. Even though $V_{1,3} := \{v^1, v^2, v^3\}$ is a tempting basis because it explains 98.5% of the discretizations variance, it is not a good choice if $f(x)$ and $f(x')$ are different: because of continuity assumptions, a surrogate model would typically suffer from inputs $\alpha \approx \alpha'$ with $y \neq y'$.

For this reason, instead of building the surrogate in the space spanned by the most relevant shape modes, we would prefer to build it in the $V_d \subset V$ basis of the most output-influencing eigenshapes $v^j$. Additionally, since the remaining “inactive” components $\tilde{\alpha}^2$ refine the shape and might explain small fluctuations of $y$, instead of omitting them (which is equivalent to stating $\tilde{\alpha}^2 = 0$), we would like to keep them in the surrogate model while prioritizing $\tilde{\alpha}^2$: a GP $Y^2(W_d \phi(x)) + Y^2(W_d \phi(x))$ is detailed in Section 3.1.2.

3.1 Supervised dimension reduction

3.1.1 Selection of active eigenshapes

To select the eigencomponents that impact $y$ the most, the penalized log-likelihood (Yi et al. 2011) of a regular, anisotropic GP in the high-dimensional space of $\alpha$s is considered,

$$\max_{\theta} pl(\alpha^{(1,\eta)}, y_{1,\eta}; \theta) \quad \text{where} \quad pl(\alpha^{(1,\eta)}, y_{1,\eta}; \theta) := l(\alpha^{(1,\eta)}, y_{1,\eta}; \theta) - \lambda \|\theta^{-1}\|_1.$$  

The $\theta$ are the GP’s hyperparameters made of the length-scales $\theta_j$, a constant mean term $\beta$, and the variance of the GP $\sigma^2$. $\alpha^{(1,\eta)}$ are the eigencomponents of the evaluated designs $x^{(1)}, \ldots, x^{(n)}$, and $y_{1,\eta}$ the associated outputs,

$$pl(\alpha^{(1,\eta)}, y_{1,\eta}; \theta) = -\frac{n}{2} \log(2\pi) - \frac{1}{2} \log(|K_\theta|) - \frac{1}{2} (y_{1,\eta} - 1\hat{\beta})^T K_\theta^{-1} (y_{1,\eta} - 1\hat{\beta}) - \lambda \|\theta^{-1}\|_1$$

where $K_\theta$ is the covariance matrix with entries $K_{\theta ij} = \sigma^2 K_{ij}(x^{(i)}, x^{(j)})$. The penalization is applied to $\theta^{-1} := (1/\theta_1, \ldots, 1/\theta_D)^T$, the vector containing the inverse length-scales of the GP. It is indeed clear (Ben Salem et al. 2019) that if $\theta_j \to +\infty$, the direction $v^j$ has no influence on $y$ as all the points are perfectly correlated together, making the GP flat in this dimension. The $L^1$ penalty term applied to the $\theta_j$s performs variable selection: this Lasso-like procedure promotes zeros in the vector of inverse length-scales, hence sets many $\theta_j$s to $+\infty$. 

\[ Springer \]
Few directions with small $\theta_j$ are selected and make the active dimensions, $\alpha^a$ (step 3 in Fig. 1). Even if the maximization of $p(l_a)$ is carried out in a $D$-dimensional space, the problem is tractable since the gradients of $p(l_a)$ are analytically known (Roustant et al. 2012), and because the $L^1$ penalty convexifies the problem. We solve it using standard gradient-based techniques such as BFGS (Liu and Nocedal 1989) with multistart.

Numerical experiments not reported here for reasons of brevity have shown that most local optima to this problem solely differ in $\theta_j$ s that are already too large to be relevant and consistently yield the same set of active variables $\alpha^a$. Notice that in Yi et al. (2011), a similar approach is undertaken but the penalization was applied on the reciprocal variables $w = (w_1, \ldots, w_D)^T$ with $w_j = 1/\theta_j$. In our work, the inverse length-scales are penalized, the gradient of the penalty is proportional to $1/\theta_j^2$. This might help the optimizer since directions with $\theta_j$ s that are not large yet are given more emphasis. In comparison, the $w$ penalty function’s gradient is isotropic. Since we can restrict the number of variables to $d' \ll D$ with no loss of information (cf. discussion at the end of Section 2.2), the dimension of (4) is substantially reduced which leads to a more efficient resolution.

### 3.1.2 Additive GP between active and inactive eigenshapes

Completely omitting the non-active dimensions, $\alpha^\pi \in \mathbb{R}^{D-d}$, and building the surrogate model $Y(\cdot)$ in the sole $\alpha^a$ space may amount to erasing some geometric patterns of the shapes which contribute to small variations of $y$. For this reason, an additive GP (Durrande et al. 2012; Duvenaud et al. 2011) with zonal anisotropy (Allard et al. 2016) between the active and inactive eigenshapes is considered (step 4 in Fig. 1):

$$Y(\alpha) = \beta + Y^a(\alpha^a) + Y^\pi(\alpha^\pi).$$

$Y^a(\alpha^a)$ is the anisotropic main-effect GP which works in the reduced space of active variables. It requires the estimation of $d + 1$ hyperparameters (the length-scales $\theta_j$ and a GP variance $\sigma_a^2$) and aims at capturing most of $y$’s variation, related to $\alpha^a$’s effect. $Y^\pi(\alpha^\pi)$ is a GP over the large space of inactive components. It is a GP which just takes residual effects into account. To keep $Y^\pi(\alpha^\pi)$ tractable, it is considered isotropic, i.e., it only has 2 hyper-parameters, a unique length-scale $\theta_\pi$ and a variance $\sigma^2_\pi$. In the end, even though $Y(\alpha)$ operates with $\alpha$’s $\in \mathbb{R}^D$ and there are fewer observations than dimensions, $n \ll D$, it remains tractable since only a total of $\delta + 3 \ll n$ hyperparameters have to be learned, which guarantees the identifiability, i.e., the unicity of the hyperparameters solution even when the number of observations is small. Although the $\alpha_j$’s have different ranges, they are homogeneous in that they all multiply normalized eigenshapes. Thus, the distances inside the shape manifold, $A$, should be relevant and an isotropic model is a possible assumption, which again, tends to emphasize eigenshapes that appear the most within the designs. This additive model can be interpreted as a GP in the $\alpha^a$ space, with an inhomogeneous noise fitted by the $Y^\pi(\cdot)$ GP (Durrande 2011). It aims at modeling a function that varies primarily along the active dimensions and fluctuates only marginally along the inactive ones.

Denoting $k_a$ and $k_\pi$ the kernels of the GPs, the hyper-parameters $\theta_a = (\theta_{a_1}, \ldots, \theta_{a_D}, \sigma_a^2)$ and $\theta_\pi = (\theta_\pi, \sigma^2_\pi)$ are estimated by maximizing the log-likelihood of (6) given the observed data $y_{1:n}$.

$$l_Y(\alpha^{(1:n)}, y_{1:n}; \theta_a, \theta_\pi) = -\frac{n}{2} \log 2\pi - \frac{1}{2} \log(|K|) - \frac{1}{2}(y_{1:n} - \hat{\beta})^T K^{-1}(y_{1:n} - \hat{\beta}),$$

\footnote{Even if pruning the $\alpha_j$ components for $j > d'$ (see comments at the end of Section 2.2), $n < d'$ may hold.}
using the kergp package (Deville et al. 2015). $K = K_0 + K_\pi$, with $K_{aij} = \sigma^2_\alpha k_\alpha(a^{(i)}, a^{(j)})$, and $K_{\pi ij} = \sigma^2_\pi k_\pi(\alpha^{(i)}, \pi^{(j)})$. The correlation between $\alpha$ and $\alpha'$ being $k(\alpha, \alpha') = \sigma^2_\alpha k_\alpha(\alpha^{(i)}, \alpha^{(j)}) + \sigma^2_\pi k_\pi(\alpha^{(i)}, \pi^{(j)})$, the kriging predictor and variance of this additive GP are (Rasmussen and Williams 2006)

\[
m(\alpha) = 1_n\hat{\beta} + k(\alpha, \alpha^{(1:n)})^T K^{-1}(y_1 - 1_n\hat{\beta})
\]

\[
\delta^2(\alpha) = \sigma^2_\alpha + \sigma^2_\pi - k(\alpha, \alpha^{(1:n)})^T K^{-1}k(\alpha, \alpha^{(1:n)})
\]

(7)

3.2 Experiments: metamodeling in the eigenshape basis

We now study the performance of the variable selection and of the additive GP described in the previous section. The GP in the original space of parameters $X$, $GP(X)$ is compared with the additive GP (Section 3.1.2), $\text{AddGP}(\alpha^{(i)} + \alpha^{(j)})$.

We equip the examples of Section 2.2 with objective functions $f(x)$ that are to be modeled by the GPs. For each function, the predictive capability of both models is compared on a distinct test set using the R2 coefficient of determination. Later, in Section 4, the objective functions will be optimized.

Table 2 Average R2 over 10 runs when metamodeling the functions with $GP(X)$ or with $\text{AddGP}(\alpha + \alpha^{(j)})$

| $n$  | $f_{\text{heart}}$ | $f_{\text{caten}}$ | $f_{\text{NACA-L}}$ | $f_{\text{NACA-D}}$ |
|------|---------------------|---------------------|---------------------|---------------------|
|      | $GP(X)$ | AddGP($\alpha + \alpha^{(j)}$) | $GP(X)$ | AddGP($\alpha + \alpha^{(j)}$) | $GP(X)$ | AddGP($\alpha + \alpha^{(j)}$) | $GP(X)$ | AddGP($\alpha + \alpha^{(j)}$) |
| 20   | 0.967        | 0.992               | 0.956               | 0.995               | 0.956               | 0.995               | 0.771               | 0.986               |
| 50   | 0.991        | 0.997               | 0.976               | 0.997               | 0.976               | 0.997               | 0.861               | 0.945               |
| 100  | 0.997        | 0.999               | 0.992               | 0.999               | 0.992               | 0.999               | 0.915               | 0.969               |
| 200  | 0.999        | 0.999               | 0.997               | 0.996               | 0.997               | 0.996               | 0.915               | 0.969               |
4 Optimization in reduced dimension

We now turn to the problem of finding the shape that minimizes an expensive objective function $f(\cdot)$. To this aim, we employ the previous additive GP, which works in the space of eigenvectors $\alpha$, in an Efficient Global Optimization procedure (Jones et al. 1998): at each iteration, a new shape is determined given the previous observations $\{(\alpha^{(1)}, y_1), \ldots, (\alpha^{(n)}, y_n)\}$ by maximizing the expected improvement (EI, Mockus 1975; Jones et al. 1998) as calculated with the GP $Y(\alpha)$:

$$\alpha^{(n+1)*} = \arg \max_{\alpha \in \mathbb{R}^D} \text{EI}(\alpha; Y(\alpha)) \quad (8)$$

where the EI is defined as

$$\text{EI}(x; Y(x)) = (a - m(x))\phi_N \left( \frac{a - m(x)}{s(x)} \right) + s(x)\psi_N \left( \frac{a - m(x)}{s(x)} \right). \quad (9)$$

$m(\cdot)$ and $s(\cdot)$ are the conditional mean and standard deviation of $Y(\cdot)$ (7), respectively, while $\phi_N$ and $\psi_N$ stand for the normal cumulative distribution function and probability density function. The threshold $a$ is usually set as the current minimum, $f_{\text{min}} := \min_{i=1,\ldots,n} y_i$, while other values have also been investigated (Jones 2001; Gaudrie et al. 2018, 2020).

4.1 Alternative expected improvement maximizations

4.1.1 Maximization in the entire $\alpha$ space

The most straightforward way to maximize the EI is to consider its maximization in $\mathbb{R}^D$ as in (8). However, this optimization is typically difficult as the EI is a multi-modal and high $(D)$ dimensional function.\(^2\)

\(^2\)As explained at the end of Section 2, we can restrict all calculations to $\alpha$’s’ $d'$ first coordinates. Even though $d' \ll D$, it has approximately the same dimension as $d$, hence the optimization is still carried out in a high-dimensional space.

4.1.2 Maximization in the $\alpha^d$ space

We can however take advantage of the dimension reduction beyond the construction of $Y(\cdot): \mathbb{R}^d \rightarrow \mathbb{R}$ are the variables that affect $y$ the most and should be prioritized for the optimization of $f(\cdot)$. A second option is therefore to maximize the EI solely with respect to $Y^d(\alpha^d)$ in dimension $\delta$. This option is nonetheless incomplete as the full GP $Y(\cdot)$ requires the knowledge of $\alpha = [\alpha^d, \alpha^\perp]$.

A first simple idea to augment $\alpha^d$ is to set $\alpha^\perp$ equal to its mean, $\theta$. The inactive part of the covariance matrix $\mathbf{K}_\alpha$ would be filled with the same scalar and the full covariance matrix $\mathbf{K} = \mathbf{K}_\alpha + \mathbf{K}_\mathbf{\perp}$ would have a degraded conditioning. A second simple idea is to sample $\alpha^\perp \sim \mathcal{N}(\theta, \lambda_\mathbf{\perp})$. However, $\alpha^\perp$ act as local refinements to the shape that contribute a little to $y$, and should also be optimized. In Li et al. (2019), the authors observed that despite the gain in accuracy of surrogate models in a reduced basis (directions of largest variation of the gradient of the lift and drag in their application), a restriction to too few directions led to poorer optimizations since small effects could not be accounted for.

4.1.3 Optimization in $\alpha^d$ space complemented with a random embedding in $\alpha^\perp$

This leads to the third proposed EI maximization, which makes step 3 in Fig. 1: a maximization of the EI with respect to $\alpha^d$ and the use of a random embedding (Wang et al. 2013) to coarsely optimize the components $\alpha^\perp$: $\text{EI}([\alpha^d, \overline{\alpha}]\mathbf{1})$ is maximized, where $\overline{\alpha} \in \mathbb{R}$ is the coordinate along a random line in the $\alpha^\perp$ space, $\overline{\alpha} = (\overline{\alpha}_1, \ldots, \overline{\alpha}_{D-d})^T$. Since $\alpha^\perp$ have been classified as inactive, it is not necessary to make a large effort for their optimization. This approach can be viewed as an extension of REMBO (Wang et al. 2013). In REMBO, a lower dimensional vector $y \in \mathbb{R}^d$ is embedded in $X$ through a linear random embedding, $y \mapsto A_Ry$, where $A_R \in \mathbb{R}^{D\times\delta}$ is a random matrix. Instead of choosing a completely random and linear embedding with user-chosen (investigated in Binois et al. 2017) effective dimension $\delta$, our embedding is nonlinear (effect of the mapping $\phi(\cdot)$), supervised and semi-random (choice of the active/inactive directions). The dimension is no longer arbitrarily chosen since it is determined by the number of selected active components (Section 3.1.1), and the random part of the embedding is only associated with the inactive parts of $\alpha$. In this variant, the EI maximization is carried out in a much more tractable $\delta + 1$ -dimensional space. From its optimum, $\alpha^* = [\alpha^d*, \overline{\alpha}] \in \mathbb{R}^{\delta+1}$ arises a $D$-dimensional vector, $\alpha^{(n+1)*} = [\alpha^d*, \overline{\alpha}]\mathbf{1}$ to be evaluated by the true function (this is the pre-image problem discussed in Section 4.2).
4.1.4 EI gradient in $\alpha$ space

The expected improvement (9) is differentiable and its derivative is known in closed form (Roustant et al. 2012) in all maximization alternatives, including the the random embedding of $\mathcal{T}$ (more details in Gaudrie et al. 2019). This helps its maximization which is carried out by genoud (Mebane et al. 2011), an evolutionary algorithm using derivatives.

4.1.5 Setting bounds on $\alpha$ for the EI maximization

As seen in the examples of Section 2.2, neither the manifold of $\alpha$s nor its restriction to $\alpha^a$ needs to be hyper-rectangular domains, which is a common assumption made by most optimizers. Two strategies were imagined to control the space in which the EI is maximized (8): the first one is to restrict the EI maximization to $\mathcal{A}$ by setting it to zero for $\alpha$s that are outside of the manifold. The benefit of this approach is that only realistic $\alpha$s are proposed. But it might suffer from an incomplete description of the entire manifold of $\alpha$s, $\mathcal{A}$, which is approximated by $\mathcal{A}_N$. Additionally, given $\mathcal{A}_N$, the statement “being inside/outside the manifold” has to be clarified. We rely on a nearest neighbor strategy in which the statement “being inside/outside the manifold” has to be clarified. We rely on a nearest neighbor strategy in which the EI is maximized in $\mathcal{A}_N$’s covering hyper-rectangle, is also investigated.

4.1.6 EI maximization via the CAD parameters

A last option consists in carrying the maximization in the $X$ space through the mapping $\phi(\cdot)$ by max $\text{EI}(X; Y(V^\top(\phi(X) - \overline{\phi})) = \text{EI}(X; Y(\alpha(x))))$. This avoids both the aforementioned optimization domain handling and the pre-image search described in the following section. However, this optimization might be less efficient since it is a maximization in $d > \delta$ dimensions, and since $V\phi(x)$ is unknown, the EI loses the closed-form expression of its gradient.

4.2 From the eigencomponents to the original parameters: the pre-image problem

The (often expensive) numerical simulator underlying the objective function can only take the original (e.g., CAD) parameters as inputs. When the EI maximization is carried out in the eigencomponents space, the $\alpha$s need to be translated into $x$s. To this aim, the pre-image problem consists in finding the CAD parameter vector $x$ whose description in the shape representation space $\Phi$ equals $V\alpha^{(n+1)^*} + \overline{\phi}$. Because there are more $\alpha$s than $x$s, $D \gg d$, a strict equality may not hold and the pre-image problem is relaxed into:

$$x^{(n+1)} = \arg\min_{x \in X} \| (\phi(x) - \overline{\phi}) - V\alpha^{(n+1)^*} \|^2_{\mathbb{R}^D}. \quad (10)$$

To complete an iteration, the pre-image problem (10) is solved and its solution $x^{(n+1)}$, the parametric shape that resembles $\alpha^{(n+1)^*}$ the most, is evaluated by the simulator, which returns $y_{n+1} = f(x^{(n+1)})$. Solving the pre-image problem does not involve calls to the simulator so that it is relatively not costly. The surrogate model is then updated with $y_{n+1}$ and $\alpha^{(n+1)} := V^\top(\phi(x^{(n+1)}) - \overline{\phi})$, the $x^{(n+1)}$ description in the $V$ basis (step 6 in Fig. 1).

Depending on the $\alpha^{(n+1)^*}$ yielded by the EI maximization (remember it may not stay on the manifold $\mathcal{A}$), $\phi^{(n+1)^*} := V\alpha^{(n+1)^*} + \overline{\phi}$ and $\phi^{(n+1)} := V\alpha^{(n+1)} + \overline{\phi}$, the shape representation of the $\alpha$ promoted by the EI and the shape representation of $x^{(n+1)}$, respectively, may substantially differ. While it is mandatory to update the GP (6) with the pair $(\alpha^{(n+1)}, y_{n+1})$, it may at first seem unclear what should be done with $\alpha^{(n+1)^*}$. When $\alpha^{(n+1)^*}$ does not belong to $\mathcal{A}$ and does not have a pre-image, it might seem straightforward to ignore it. However, if $\alpha^{(n+1)^*}$ was yielded by the EI, it is very likely to be promoted in the following iterations, since its uncertainty, $s^2(\phi^{(n+1)}^*)$, has not vanished. Therefore, if $\phi^{(n+1)^*}$ and $\phi^{(n+1)}$ are substantially different, the virtual pair $(\alpha^{(n+1)}, y_{n+1})$ is included in the GP (6) too in a strategy called replication. We define replication in general terms.

**Definition 1 (Replication)** In Bayesian optimization, when the GP is built over coordinates $\alpha$ that are a mapping of the original coordinates $x$, $\alpha = T(x)$, at the end of each iteration a pre-image problem such as (10) must be solved to translate the new acquisition criterion maximizer $\alpha^{(n+1)^*}$ into the next point to evaluate $x^{(n+1)}$ and the associated iterate $\alpha^{(n+1)} = T(x^{(n+1)})$. The replication strategy consists in updating the GP with both $(\alpha^{(n+1)}, f(x^{(n+1)}))$ and $(\alpha^{(n+1)^*}, f(x^{(n+1)})$ provided $\alpha^{(n+1)^*}$ and $\alpha^{(n+1)}$ are sufficiently different.

Here, the difference between $\alpha^{(n+1)^*}$ and $\alpha^{(n+1)}$ is calculated as the distance between the associated shapes

---

3In this article, the mapping $T(\cdot)$ is the composition of $\phi(\cdot)$ with the projection onto a subspace of $(v^1, \ldots, v^D)$. 
When \( \alpha^{(n+1)*} \notin A \), the solution of the pre-image problem (in the \( \alpha \) space), \( \alpha^{(n+1)} \) is its projection on \( A \).

Since the database \( \Phi \) contains the shape representation of \( N \) distinct designs, \( d_0 := \min_{i,j=1,...,N} \| \Phi_i - \Phi_j \|_{R^D} \), the minimal distance between two different designs in \( \Phi \) is used as a threshold beyond which \( \phi^{(n+1)} \) and \( \phi^{(n+1)*} \) are considered to be different. The replication strategy is further motivated by the fact that since \( x^{(n+1)} = \text{arg min}_{x \in X} \| \phi(x) - \bar{\phi} \|_{R^D} \)

\[
\text{Va}(\alpha^{(n+1)*})^2_{R^D} = \text{arg min}_{x \in X} \| V^T (\phi(x) - \bar{\phi}) - \alpha^{(n+1)*} \|^2_{R^D},
\]

where the last equality expresses just a change of basis since \( V \) is orthogonal, \( \alpha^{(n+1)} \) is an orthogonal projection of \( \alpha^{(n+1)*} \) on \( A \), see Fig. 8.

This is somehow similar to Raghavan et al. (2013, 2014) where the authors project non-realistic shapes on a smooth surface built via diffuse approximation or a local polynomial fitting, using the points of \( A_N \), to retrieve a realistic design. In our approach, unrealistic shape representations are directly projected onto \( A \) through the resolution of (10). Incorporating the non-physical \( (\alpha^{(n+1)*}, y_{n+1}) \) in the surrogate model can be viewed as an extension of the surrogate model outside its domain (Shahriari et al. 2016) (outside the manifold \( A \) in our case) by constant prolongation.

4Since we do not know the convexity of \( A \), the projection might not be unique.

5 Experiments

The ideas developed in Sections 2, 3, and 4 when put together make the method already sketched in Fig. 1 and more detailed in the following pseudo-code:

\begin{algorithm}
\caption{Pseudo-code of the Bayesian optimization in reduced eigencomponents, AddGP \((\alpha^a, \alpha^r)\) -EI embed with replication.}
\begin{algorithmic}
\State Sample \( N \) designs \( x^{(i)} \) and discretize them to form the matrix \( \Phi \); /* see Section 2.1 */
\State Eigendecomposition of \( \Phi \): \( \Phi = V \Sigma V^T \) with \( \Sigma \) diagonal and \( V \) orthonormal, \( \alpha = \text{proj}_{A} \alpha^{(n+1)} \) and \( \alpha^{(n+1)*} \).
\State Evaluate \( n \) designs \( \Phi^{(1:n)} \Rightarrow y_{1:n} \), and compute their eigencomponents \( \alpha^{(1:n)} \).
\While {\( n < \text{computational budget} \)}
\State Maximize \( \text{EI} (\alpha^{(1:n)}, y_{1:n}, \theta) \Rightarrow \text{active} \) and \( \text{inactive} \) eigencomponents, \( \alpha = [\alpha^a, \alpha^r] \); /* see Section 3.1.1 */
\State Build the additive GP \( Y(\alpha) = \beta + Y^a(\alpha^a) + Y^r(\alpha^r) \); /* see Section 3.1.2 */
\State Randomly draw a vector of \( D - 8 \) components, \( \bar{\alpha} \).
\State Maximize the EI with respect to \( [\alpha^a, \bar{\alpha}] \) \Rightarrow \( \alpha^{(n+1)*} \) next shape to be evaluated; /* sharp maximization w.r.t. \( \alpha^a \) \) and coarse maximization w.r.t. \( \alpha^r \), see Section 4.1 * /
\State Solve pre-image problem \( \Rightarrow x^{(n+1)} \) to be evaluated \( \Rightarrow y_{n+1} = f(x^{(n+1)}) \), and \( \alpha^{(n+1)} = V^T (\phi(x^{(n+1)}) - \bar{\phi}) \), associated eigencomponents; /* see Section 4.2 * /
\If {\( \alpha^{(n+1)} \text{ and } \alpha^{(n+1)*} \text{ too different} \)}
\State Update the GP with \( (\alpha^{(n+1)}, y_{n+1}) \) and \( (\alpha^{(n+1)*}, y_{n+1}) \); /* Replication, see Definition 1 */
\Else
\State Update the GP with \( (\alpha^{(n+1)}, y_{n+1}) \);
\EndIf
\State \( n \leftarrow n + 1 \);
\EndWhile
\end{algorithmic}
\end{algorithm}

We now compare the versions of the algorithm made of the additive GP and the different EI maximizations with the standard technique of performing the optimization in the original space, \( X \). A more complete investigation of models (including those of Section 3) and EI maximizations can be found in Gaudrie et al. (2019). We introduce a shorthand...
The initial Design of Experiments (DoE) is made of computational effort is put on the internal EI maximization. The initial Designs of Experiments (DoE) are made of two parts separated by a dash, AddGP \((\alpha^a + \alpha^\tau) - EI (\alpha^a)\) and \(1.18 (0.45)\), \(27.58 (0.53)\), \(166.9 [4]\), \(42.1 (26.5)\). As the designs have defined in Section 3.2. For fair comparison, the same introduced in Section 2.2 whose objective functions were \((\text{Def. 1})\) is employed.

The notation: \(\text{GP} (X)\) is the standard approach in which both the metamodel and the EI maximization are carried out in the design space \(X\); the names of the algorithms with additive \(\text{GP}\) are made of two parts separated by a dash, AddGP \((\alpha^a + \alpha^\tau) - EI\) embed no replication. The second part of the name specifies if the EI maximization is restricted to \(A_N\) (on manifold) and if not, if the replication strategy \((\text{Def. 1})\) is employed.

We now turn to the shape optimization of the designs introduced in Section 2.2 whose objective functions were defined in Section 3.2. For fair comparison, the same computational effort is put on the internal EI maximization. The initial Design of Experiments (DoE) is made of \(n = 20\) observations for the additive \(\text{GP}\). As the designs have \(d > 20\) variables, \(\text{GP} (X)\) initialized with \(n_X = 50\) individuals in the heart rectangle and \(n_X = 40\) in the catenoid problem. The algorithms are run for a total of \(2n_X\) functions evaluations, i.e., \(80\) or \(50\) additional calls to \(f(\cdot)\) are allowed on the rectangle problem for AddGP and \(\text{GP} (X)\), and \(60\) or \(40\) on the catenoid problem. Tables 3 and 4 report the best objective function value found at the end of the search, averaged over \(10\) runs, as well as the number of function evaluations required to reach a fixed target on the rectangle and catenoid problem. When at least one run has not reached the target, a rough estimator of the empirical runtime \((\text{Auger and Hansen 2005})\), \(\overline{t}_s/p_s\), is provided, in red, the number of runs achieving the target value being reported in brackets, where \(\overline{t}_s\) and \(p_s\) correspond to the average number of function evaluations of runs that reach the target and the proportion of runs attaining it. \(\times\) signifies that no run was able to attain the target within the limited budget. The computation of the NACA 22’s lift and drag being computationally more expensive, only one \(\text{GP} (X)\) and one AddGP \((\alpha^a + \alpha^\tau) - EI\) embed run are performed.

### 5.1 Heart rectangle

\(f_{\text{heart}}(\mathbf{x})\), that expresses the distance from a shape to a rectangle deformed as an heart, has to be minimized. For the additive \(\text{GP}\), the four EI maximization options of Section 4.1 are compared. Three EI maximization strategies are carried out in the covering hyper-rectangle of \(A_N\) (as opposed to restricted to the neighborhood of \(A_N\), on manifold), and EI maximizers which do not belong to \(A\) are nonetheless used for the \(\text{GP}\) update: the with replication strategy is employed because as will be shown next on the catenoid problem, this variant outperforms the no replication and the on manifold strategy.

In a first order approximation, \(v^3\) and \(v^4\) are the most influential eigenshapes with regard to \(f_{\text{heart}}\), which measures the nodal difference between \(\Omega_x\) and the target \(\Omega_t\), because they act on the rectangle’s size, see Fig. 2. These eigenshapes are always selected by the additive \(\text{GP}\) \((v^1, v^2, \text{and other eigenshapes are sometimes selected too})\). All additive \(\text{GPs}\) benefit from the reduced dimension and the prioritization of active eigenshapes and clearly outperform \(\text{GP} (X)\). In this example where two eigenshapes out of 40 mainly influence \(f_{\text{heart}}\), the EI maximization in the full space of \(\alpha\) \((\text{EI} (\alpha))\) is the weakest option. Better results are obtained with the embedding strategy or with the maximization of the active \(\alpha\) only. AddGP \((\alpha^a + \alpha^\tau) - EI\) embed needs more iterations to attain good values (smaller than 0.5) than AddGP \((\alpha^a + \alpha^\tau) - EI (\alpha^a)\) which is an early starter. This might be due to the additional though

| Method | Best value | Time to 0.5 | Time to 1 | Time to 3 |
|--------|------------|-------------|-----------|-----------|
| \(\text{GP} (X)\) | 1.18 (0.45) | × | 166.9 [4] | 42.1 (26.5) |
| AddGP \((\alpha^a + \alpha^\tau) - EI (\alpha^a)\) | 0.54 (0.19) | 199.4 [4] | 40.2 (12.3) | 30.2 (10.5) |
| AddGP \((\alpha^a + \alpha^\tau) - EI\) embed | 0.37 (0.08) | 49.0 (21.4) | 26.1 (5.6) | 22.2 (1.9) |
| AddGP \((\alpha^a + \alpha^\tau) - EI\) | 0.37 (0.09) | 33.3 (14.6) | 22.7 (2.6) | 21.4 (0.7) |
| AddGP \((\alpha^a + \alpha^\tau) - EI\) embed | 0.60 (0.26) | 106.7 [6] | 41.2 [9] | 21.5 (0.5) |

All algorithms with the additive \(\text{GP}\) perform the EI search in \(A_N\)’s covering hyper-rectangle and do it with replication.
Fig. 9 Optimization with EI maximization in the covering hyper-rectangle of \( \mathcal{A}_N \) without (left) or with (right) replication strategy.

less critical component (\( \overline{\alpha} \)) considered by this method. Passing through the \( X \) space for the EI maximization (\( \text{EI}(X) \)) is a weak option. Even though results are more stable and on average better than with \( \text{EI}(\alpha) \), \( \text{AddGP}(\alpha^d + \alpha^p) - \text{EI}(X) \) needs much more function evaluations to attain moderately good values of \( f_{\text{heart}} \).

5.2 Catenoid shape

We want to find a curve \( r(y) \) which minimizes the associated axisymmetric surface as expressed by the integral making \( f_{\text{cat}(x)} \) in the catenoid problem. With the additive GP, the EI maximization with embedding (\( \text{EI embed} \)) is

Fig. 10 Top row: drag optimization of the NACA 22 airfoil in the reduced eigenbasis with \( \text{AddGP}(\alpha^d + \alpha^p) - \text{EI embed} \) (left) or carried out in the CAD parameters space with \( \text{GP}(X) \) (center). Low drag airfoils are found with \( \text{AddGP}(\alpha^d + \alpha^p) - \text{EI embed} \) while the classical method still evaluates the airfoils of the initial design of experiments (right). Bottom row: lift optimization of the NACA 22 airfoil in the reduced eigenbasis with \( \text{AddGP}(\alpha^d + \alpha^p) - \text{EI embed} \) (left) or carried out in the CAD parameters space with \( \text{GP}(X) \) (center). High lift airfoils are found while the classical method still evaluates the airfoils of the initial design of experiments (right), i.e., lower objective functions are obtained faster.
employed. Here, we investigate the effect of restricting the search to \( \mathcal{A}_N \) (on manifold). If not, the maximization is carried out in \( \mathcal{A}_N \)'s covering hyper-rectangle, in which case the impact of the replication strategy is analyzed.

Again, the worth of the additive GPs is highlighted in this example: they clearly perform better than \( \text{GP}(X) \). Table 4 indicates that staying in the vicinity of \( \mathcal{A}_N \) worsens the convergence. Indeed, promising \( \alpha \)'s are difficult to attain or are even falsely considered as outside \( \mathcal{A} \). Staying both in the neighborhood of \( \mathcal{A}_N \) and on the random line further restricts the spectrum of valid \( \alpha \)'s. Better results are obtained when the EI maximization is carried out in the covering hyper-rectangle of \( \mathcal{A}_N \), in which case the replication strategy exhibits better performance than the strategy where only the projection, \( \alpha^{(n+1)} \), is used for updating the GP (in cases where the EI maximizer \( \alpha^{(n+1)*} \notin \mathcal{A} \)). Figure 9 shows the typical effect of the replication strategy. On the left, the inner EI maximization is carried out in the covering hyper-rectangle of \( \mathcal{A}_N \) but only the \( \alpha \in \mathcal{A} \) obtained through the pre-image problem solving are used to construct the surrogate model. On the right, all EI maximizers have been used for the GP, including \( \alpha \notin \mathcal{A} \). Without replication, since the variance of the GP at previous EI maximizers has not vanished, the EI continues promoting the same \( \alpha \)'s, which have approximately the same pre-image. The same part of the \( \alpha \) space is sampled, which not only leads to a premature convergence (the best observed value has already been attained after 6 iterations), but also increases the risk of getting a singular covariance matrix. With the replication, the GP variance vanishes for all EI maximizers, even those outside \( \mathcal{A} \), removing any further EI from these \( \alpha \)'s. The \( \alpha \) space is better explored with benefits on the objective value (26.26 against 27.13 here).

**NACA 22 optimization**

In this last test case, we compare the two aforementioned algorithms by optimizing the lift coefficient and the drag coefficient of a NACA 22 airfoil (\( f_{\text{NACA-L}} \) and \( f_{\text{NACA-D}} \)). The simulation is made with a computational fluids dynamic code that solves the Reynolds Averaged Navier-Stokes (RANS) equations with \( k-\varepsilon \) turbulence model. Since a single call to the simulator (one calculation of \( f_{\text{NACA}} \)) takes about 20 minutes on a standard personal computer, only two runs are compared for each objective. The first algorithm is the classical Bayesian optimizer where the GP is built in CAD parameter space, \( \text{GP}(X) \). In the second algorithm, \( \text{AddGP}(\alpha^a + \alpha^\overline{a}) - \text{EI embed} \), the GP is built...
in the $V$ basis of eigenshapes, while prioritizing the active dimensions, $\alpha^a$, via the additive GP and the EI random embedding method with the replication option, see Sect. 4.2. The optimization in the eigenshape basis starts with a DoE of $n = 10$ designs and is run for $p = 90$ additional iterations while, because there are 22 $x_i$’s, the optimization in the CAD parameters space starts using $n = 50$ designs and is run for $p = 50$ iterations.

Fig. 10 shows the optimization runs of both algorithms for the minimization of the NACA 22’s drag (top) and lift (bottom), and Fig. 11 the resulting airfoils.

In this application, the main advantage of the AddGP ($\alpha^d + \alpha^\Pi$) - EI embed (Fig. 10, top left and bottom left) over the standard Bayesian optimizer (top center and bottom center) is that it enables an early search for low drag, respectively high lift airfoils, at a time when the standard approach is still computing its initial DoE. Indeed, the classical method needs much more function evaluations for building the initial surrogate model (black dots) because the inputs live in a space of higher dimension. The approach introduced in this paper would further gain in relevance in problems with more than $d = 22$ CAD parameters, where it would almost be impossible to build a large enough initial design of experiments (whose size is typically of the order of $10^d$ dimension (Loeppky et al. 2009)).

It is observed in Fig. 11 that smoother airfoils are obtained with AddGP ($\alpha^d + \alpha^\Pi$) - EI embed (right column), because it uses a shape coordinate system instead of treating the $L_i$’s (i.e., $x_i$’s with local influences on the airfoil, see top left plot of Fig. 4) separately, as is done by GP ($X$) (left column). When the optimization aims at minimizing the drag, the AddGP ($\alpha^d + \alpha^\Pi$) - EI embed airfoil (top right) is smoother than the GP ($X$) one (top left). And when the objective is to maximize the lift, the camber of the AddGP ($\alpha^d + \alpha^\Pi$) - EI embed airfoil (bottom right) is increased in comparison with the design yielded by GP ($X$) (bottom left).

## 6 Conclusions

We have proposed a new methodology to apply Bayesian optimization techniques to parametric shapes and other problems where a pre-existing set of relevant points and a fast auxiliary mapping exist. Instead of working directly with the CAD parameters, which are too numerous for an efficient optimization and may not be the best representation of the underlying shape, we unveil the lower dimensional manifold of shapes through the auxiliary mapping and PCA. The dimensions of this manifold that contribute the most to the variation of the output are identified through an $L^1$ penalized likelihood and then used for building an additive Gaussian Process with a zonal anisotropy on the selected variables and isotropy on the other variables. This GP is then utilized for Bayesian optimization.

The construction of the reduced space of variables opens the way to several strategies for the maximization of the acquisition criterion, in particular the restriction or not to the manifold and the replication. The different variants for the construction of the surrogate model and for the EI maximization have been compared on 3 examples, 2 of them being analytical and easily reproducible, the last one being a realistic airfoil design.

Even though specific variants are more or less adapted to features of specific test problems, the supervised dimension reduction approach and the construction of an additive GP between active and inactive components have given the most reliable results.

Regarding the EI maximization, our experiments highlight the efficiency of the random embedding in the space of inactive variables in addition to the detailed optimization of the active variables. It is a trade-off between optimizing the active variables only, and optimizing all variables. Benefits have been observed for not restricting this inner maximization to the current approximation of $A$ as well as for the virtual replication of points outside $A$ when $\alpha \notin A$ is promoted by the EI.

Further research should consider shapes made of multiple elements. This is of practical importance and it brings a new theoretical feature, the presence of symmetries in $\Phi$. The knowledge about symmetries has to be propagated to the eigenshape space to enhance the surrogate model.

**Funding information** This research was partly funded by a CIFRE grant (convention #2016/0690) established between the ANRT and the Groupe PSA for the doctoral work of David Gaudrie.

**Compliance with ethical standards**

**Conflict of interest** The authors declare that they have no conflict of interest.

**Replication of results** Comparisons between variants of EGO algorithms working in the original $X$ space or in a reduced eigencoordinates space have been presented in Sections 3 and 4. Two examples out of three are analytical and easily reproducible. To facilitate a replication of results, the pseudo-code of the final approach we propose, AddGP ($\alpha^d + \alpha^\Pi$) - EI embed with replication, is given in Section 5. The penalized maximum likelihood was implemented in the R language extending the kergp package. The additive GP was built using the kergp package. The maximization of the EI was carried out by the genoud package.
References

Allard D, Senoussi R, Porcu E (2016) Anisotropy models for spatial data. Math Geosci 48(3):305–328
Auger A, Hansen N (2005) Performance evaluation of an advanced local search evolutionary algorithm. In: Congress on evolutionary computation, vol 2. IEEE, pp 1777–1784
Bellman RE (1961) Adaptive control processes: a guided tour. Princeton university press
Ben Salem M, Bachoc F, Roustant O, Gamboa F, Tomaso L (2019) Sequential dimension reduction for learning features of expensive black-box functions. hal-01688329v2. https://hal.archives-ouvertes.fr/hal-01688329v2
Berkooz G, Holmes P, Lumley JL (1993) The proper orthogonal decomposition in the analysis of turbulent flows. Ann Rev Fluid Mech 25(1):539–575
Binois M, Ginsbourger D, Roustant O (2017) On the choice of the low-dimensional domain for global optimization via random embeddings. arXiv:1704.05318
Bouhelé MA, Bartoli N, Otsmane A, Morlier J (2016) Improving kriging surrogates of high-dimensional model designs by partial least squares dimension reduction. Struct Multidiscip Optim 53(5):935–952
Cingolani D, Iuliano E (2018) Investigation of adaptive design variables bounds in dimensionality reduction for aerodynamic shape optimization. Comput Fluids 174:89–109
Colding TH, Miniciccozi WP (2006) Shapes of embedded minimal surfaces. Proc Nat Acad Sci 103(30):11,106–11,111
Constantine P, Dow E, Wang Q (2014) Active subspace methods in theory and practice: applications to kriging surfaces. SIAM J Sci Comput 36(4):A1500–A1524
Cootes TF, Taylor CJ, Cooper DH, Graham J (1995) Active shape models-their training and application. Comput Vis Image Understanding 61(1):38–59
Cressie N (1992) Statistics for spatial data. Terra Nova 4(5):613–617
Deville Y, Ginsbourger D, Durrande N, Roustant O, Roustant O (2015) Package`kergp`
Durrande N (2011) Étude de classes de noyaux adaptées à la simplification et à l’interprétation des modèles d’approximation. une approche fonctionnelle et probabiliste. Ph.D. thesis École Nationale Supérieure des Mines de Saint-Étienne
Durrande N, Ginsbourger D, Roustant O (2012) Additive covariance kernels for high-dimensional Gaussian process modeling. In: Annales de la Faculté des sciences de Toulouse: Mathématiques, vol 21, pp 481–499
Duvenaud D, Nickisch H, Rasmussen CE (2011) Additive Gaussian processes. In: Advances in neural information processing systems, pp 226–234
Forrester AI, Keane AJ (2009) Recent advances in surrogate-based optimization. Progress Aerosp Sci 45(1–3):50–79
Frank IE, Friedman J (1993) A statistical view of some chemometrics. Technometrics 35(2):109–135
Gaudrie D, Le Riche R, Picheny V, Enaux B, Herbert V (2018) Budgeted multi-objective optimization with a focus on the central part of the Pareto front-extended version. arXiv:1809.10482
Gaudrie D, Le Riche R, Picheny V, Enaux B, Herbert V (2019) Modeling and optimization with Gaussian processes in reduced eigenbases - extended version. arXiv:1908.11727
Gaudrie D, Le Riche R, Picheny V, Enaux B, Herbert V (2020) Targeting solutions in Bayesian multi-objective optimization: sequential and batch versions. Ann Math Artif Intell 88:187–212
Jones DR (2001) A taxonomy of global optimization methods based on response surfaces. J Global Optim 21(4):345–383
Jones DR, Schonlau M, Welch WJ (1998) Efficient global optimization of expensive black-box functions. J Global Optim 13(4):455–492
Li J, Bouhlel MA, Martins J (2018) A data-based approach for fast airfoil analysis and optimization. In: 2018 AIAA/ASCE/AHS/ASC structures, structural dynamics, and materials conference, p 1383
Li J, Cai J, Qu K (2019) Surrogate-based aerodynamic shape optimization with the active subspace method. Struct Multidiscip Optim 59(2):403–419
Liu D, Nocedal J (1989) On the limited memory BFGS method for large scale optimization. Math Program 45(1–3):503–528
Loopeky JL, Sacks J, Welch WJ (2009) Choosing the sample size of a computer experiment: a practical guide. Technometrics 51(4):366–376
Mebane WR Jr, Sekhon JS et al (2011) Genetic optimization using derivatives: the rngend package for R. J Stat Softw 42(11):1–26
Mika S, Schölkopf B, Smola A, Müller KR, Scholz M, Rätsch G (1999) Kernel PCA and de-noising in feature spaces. In: Advances in neural information processing systems, pp 536–542
Mockus J (1975) On Bayesian methods for seeking the extremum. In: Optimization techniques IFIP technical conference. Springer, pp 400–404
Namura N, Shiomoyama K, Obayashi S (2017) Kriging surrogate model with coordinate transformation based on likelihood and gradient. newblock Journal of Global Optimization 68(4):827–849
Palar PS, Shiomoyama K (2018) On the accuracy of kriging model in active subspaces. In: 2018 AIAA/ASCE/AHS/ASC structures, structural dynamics, and materials conference, p 9913
Raghaban B, Breitkopf P, Tourbier Y, Villon P (2013) Towards a space reduction approach for efficient structural shape optimization. Struct Multidiscip Optim 48(5):987–1000
Raghaban B, Le Quilliec G, Breitkopf P, Rassineux A, Roelandt JM, Villon P (2014) Numerical assessment of springback for the deep drawing process by level set interpolation using shape manifolds. Int J Mater Form 7(4):487–501
Rasmussen CE, Williams CK (2006) Gaussian processes for machine learning. The MIT Press
Roustant O, Ginsbourger D, Deville Y (2012) DiceKriging, DiceOptim: two R packages for the analysis of computer experiments by kriging-based metamodeling and optimization
Sacks J, Welch WJ, Mitchell TJ, Wynn HP (1989) Design and analysis of computer experiments. Stat Sci, 409–423
Saltelli A, Tarantola S, Campolongo F, Ratto M (2004) Sensitivity analysis in practice: a guide to assessing scientific models. Chichester
Schölkopf B, Smola A, Müller KR (1997) Kernel principal component analysis. In: International conference on artificial neural networks. Springer, pp 583–588
Shahriari B, Bouchard-Côté A, Freitas N (2016) Unbounded Bayesian optimization via regularization. In: Artificial intelligence and statistics, pp 1168–1176
Shan S, Wang GG (2004) Space exploration and global optimization for computationally intensive design problems: a rough set based approach. Struct Multidiscip Optim 28(6):427–441
Shan S, Wang GG (2010) Survey of modeling and optimization strategies to solve high-dimensional design problems with computationally-expensive black-box functions. Struct Multidiscip Optim 41(2):219–241
Stein M (1999) Interpolation of spatial data: some theory for kriging. Springer Science & Business Media
Tripathy R, Bilionis I, Gonzalez M (2016) Gaussian processes and optimization with built-in dimensionality reduction: applications to high-dimensional uncertainty propagation. J Comput Phys 321:191–223
Vapnik V (1995) The nature of statistical learning theory. Springer science & business media
Wall ME, Rechtsteiner A, Rocha LM (2003) Singular value decomposition and principal component analysis. In: A practical approach to microarray data analysis. Springer, pp 91–109
Wang Q (2012) Kernel principal component analysis and its applications in face recognition and active shape models. arXiv:1207.3538
Wang Z, Zoghi M, Hutter F, Matheson D, De Freitas N (2013) Bayesian optimization in high dimensions via random embeddings. In: Twenty-Third international joint conference on artificial intelligence

Wu X, Peng X, Chen W, Zhang W (2019) A developed surrogate-based optimization framework combining HDMR-based modeling technique and TLBO algorithm for high-dimensional engineering problems. Struct Multidiscip Optim 60:663–680
Yi G, Shi J, Choi T (2011) Penalized Gaussian process regression and classification for high-dimensional nonlinear data. Biometrics 67(4):1285–1294

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