Multi-objective design space exploration using explainable surrogate models

Pramudita Satria Palar, Yohanes Bimo Dwianto, Lavi Rizki Zuhal, Joseph Morlier, Koji Shimoyama, Shigeru Obayashi

Multi-objective design space exploration using explainable surrogate models

Pramudita Satria Palar[1] · Yohanes Bimo Dwianto[1] · Lavi Rizki Zuhal[1] · Joseph Morlier[2] · Koji Shimoyama[3] · Shigeru Obayashi[4]

Abstract

The surrogate model is an essential part of modern design optimization and exploration. In some cases, exploration of design space in multi-objective problems is important to reveal useful design insight and guidelines that will be useful for engineers. However, most surrogate models are black boxes, making interpretation difficult. This paper investigates the framework of explainable surrogate models using Shapley Additive Explanations (SHAP) to gain important design insight that helps users better understand the relationship between objective functions and design variables. We applied the explainable surrogate model framework to multi-objective design problems and performed a comparison with active subspaces and Sobol indices. Several techniques to extract design insight based on SHAP values are discussed: the averaged SHAP, the SHAP summary plot, the single- and bi-objective SHAP dependence plot, and the SHAP correlation matrix. Two aerodynamic design cases are selected to demonstrate the capability of explainable surrogate models: nine-variable inviscid and twenty-variable viscous transonic airfoil design. The findings indicate that SHAP provides more valuable insights than active subspaces and Sobol indices, particularly regarding the impact of individual design variables on the objectives. Consequently, SHAP can be employed in conjunction with active subspaces and Sobol indices to explore the input–output relationship in multi-objective design exploration comprehensively.

Keywords Design space exploration · Multi-objective · Surrogate model · Interpretability · Shapley additive explanations

1 Introduction

Optimization plays an essential role in engineering design, which aims to improve the performance of solutions by finding better alternatives that enhance performance (e.g., efficiency). Complementary to optimization, Design Space Exploration (DSE) plays a crucial role in uncovering crucial trends and physical insights that benefit engineers. DSE may include various tasks such as global sensitivity analysis, parameter sweep, and complexity analysis of the design space. Although DSE sometimes refers to electronic and embedded systems (Pimentel 2016), the term can refer to any systematic design analysis in virtually any field. Surrogate models, which are fast analytical approximations of black-box models, are enabling technologies that greatly assist DSE. There is a plethora of surrogate models available in the literature. Some popular examples include support vector regression (SVR) (Smola and Schölkopf 2004), Gaussian Process Regression (GPR)/Kriging (Rasmussen 2003), neural network/deep learning (LeCun et al. 2015), random forest (Ho 1998) and polynomial regression/polynomial chaos expansion (PCE) (Blatman and Sudret 2011; Xiu and Karniadakis 2003).

There are several design frameworks similar to DSE. For example, the framework of multi-objective design exploration (MODE) uncovers the structure of design space in multi-objective optimization through data mining,

Responsible Editor: Xiaoping Du

✉ Pramudita Satria Palar
pramsp@itb.ac.id

[1] Faculty of Mechanical and Aerospace Engineering, Institut Teknologi Bandung, Bandung, Indonesia

[2] ICA, Université de Toulouse, ISAE–SUPAERO, INSA, CNRS, MINES ALBI, UPS, Toulouse, France

[3] Department of Mechanical Engineering, Kyushu University, Fukuoka 819-0395, Japan

[4] Institute of Fluid Science, Tohoku University, Sendai, Miyagi 980-8577, Japan

visualization, and trade-off analysis (Obayashi et al. 2005, 2007, 2010). Examples of tools used in MODE include variance-based sensitivity analysis (Sobol 2001), self-organizing map (SOM) (Kohonen 1990), decision theory, and rough sets (Pawlak 1998). The main emphasis of MODE is on revealing trade-off information between multiple objectives. Recent applications of such design thinking include intake design optimization (Brahmachary et al. 2020) and blended-wing-body type flyback booster (Sumimoto et al. 2019). An integral part of MODE encompasses a data-efficient optimization algorithm, where surrogate-based optimization plays a crucial role in this context. Bayesian optimization emerges as an important surrogate-based optimization approach that has proven effective in addressing complex engineering scenarios, including optimization tasks within the aerospace design domain (Grapin et al. 2022; Bartoli et al. 2023). A more general term for such a research endeavour is *knowledge discovery* from multi-objective optimization, in which Bandaru et al. (2017) emphasize the importance of visual data mining methods and machine learning (ML) for knowledge discovery. In recent decades, the field of ML has experienced significant growth, including the subfield of interpretable ML. The primary focus of interpretable ML is to provide accurate predictions and extract valuable insights. Within the context of ML models, interpretability is defined as the capacity of an ML model to offer easily comprehensible explanations to humans (Doshi-Velez and Kim 2017). Although intrinsically interpretable models (e.g., linear regression) are easy to understand, they often lack accuracy when compared to more complex and less interpretable models. To address this issue, model-agnostic techniques have been developed to facilitate the interpretation of black-box models. These techniques help to dissect the black-box models and enhance their explainability. Although interpretability and explainability are sometimes used interchangeably, we use the former to denote inherently interpretable models, while the latter is used when explaining hard-to-interpret models (i.e., to make the models explainable). Some explainability techniques include partial dependence plot (PDP) (Greenwell et al. 2018), accumulated local effects (Apley and Zhu 2020), individual conditional expectations (ICE) (Goldstein et al. 2015), local surrogate (LIME), and, most recently, Shapley Additive Explanations (SHAP) (Lundberg and Lee 2017). To take examples outside engineering design, explainable models are important to allow reasonable data-driven decision models in healthcare (Stiglic et al. 2020), predictive maintenance (Bukhsh et al. 2019; Vollert et al. 2021), and genetics (Azodi et al. 2020).

Applying explainable ML in engineering, particularly in engineering design optimization, is still relatively uncommon. However, the explainable framework has tremendous potential for assisting engineering design optimization and exploration. Understanding the trade-off between objectives and the individual impact of design variables is crucial for engineers. Additionally, engineers can use explainable ML models to better analyze the input–output relationship, including nonlinearity. Surrogate models are particularly beneficial in engineering as they allow rapid exploration of the design space using high-fidelity simulations. With the growing interest in ML for engineering design and the increasing use of surrogate models, there is enormous potential for deploying explainable ML techniques to support engineering design.

A surrogate model is often a black-box model that is difficult to interpret. While accuracy is critical in DSE, exploring the surrogate model can provide better design insights to users. SHAP is one of the most recent and useful explainable ML techniques available. SHAP connects Shapley values and LIME, making it highly effective for explaining surrogate models. However, despite its usefulness, SHAP is still not commonly applied in engineering design, and most existing applications are unrelated to engineering optimization and DSE. For example, SHAP has been used in the failure mode and effect analysis in civil engineering (Mangalathu et al. 2020) and diagnostic for nuclear power plants (Park et al. 2022). Recent papers on SHAP for knowledge discovery include Palar et al. (2023) and Takanashi et al. (2023). Palar et al. (2023) emphasizes the use SHAP for single-criteria analysis and compares its effectiveness with the Morris' elementary effect. On the other hand, Takanashi et al. (2023) applies SHAP for multi-criteria analysis of aerodynamic design. The current work shares a similar spirit with both papers. The main theme of this paper is to systematically investigate how SHAP can be used for effective multi-objective engineering design exploration.

The main objective of this paper is to introduce the framework of the explainable surrogate model so that the user can gain better design insight from the analysis of the multiple input-multiple output relationship. This paper also investigates the capability of SHAP for trade-off analysis in multi-objective design. We present several approaches to visualize SHAP in the context of single- and bi-criteria design exploration, making it possible to depict important information regarding the relationship between the design variables and the two objectives. Several methods to aid multi-objective design based on SHAP are presented, including (1) the averaged SHAP values, (2) the SHAP summary plot, (3) the single-objective SHAP dependence plot, (4) the bi-objective SHAP dependence plot, and (5) the SHAP correlation matrix. Specifically, we introduce the utilization of SHAP correlation values to deepen comprehension. Additionally, we suggest enhancing the SHAP dependence plot by incorporating the relative impact of input variables and the SHAP correlation values, aiming to broaden knowledge and improve clarity. Using GPR and polynomial chaos (PC)-Kriging (Kersaudy et al. 2015) as the models of

choice, the multi-objective explainable surrogate model framework is then demonstrated on two engineering design problems: a nine-variable inviscid airfoil and 20-variable viscous airfoil design problem. Comparison with related techniques, namely active subspace method (ASM) (Constantine 2015) and Sobol indices (Sobol 2001), is performed to understand better the advantages and possible disadvantages of SHAP compared to the two techniques.

The primary contributions of this paper can be outlined in three aspects: (1) Introducing a unified framework for multi-objective design exploration using SHAP, enhancing analysis and providing insights that may be challenging or impossible to obtain through other methods, (2) introducing the SHAP correlation matrix to facilitate the analysis of multi-objective design exploration, revealing variables that contribute to the trade-offs between objectives alongside their input importance, and (3) comparing SHAP with established methods, namely Sobol indices and ASM, specifically in the context of multi-objective design exploration.

The rest of this paper is structured as follows. Section 2 explains the surrogate model types utilized in this paper. Section 3 details the Shapley values, SHAP, and various methods to explore information from SHAP for multi-objective design exploration. Section 4 presents numerical results on aerodynamic problems. Finally, Sect. 5 concludes the paper with suggestions for future works.

2 Surrogate modeling

Let us denote an input variable vector $\boldsymbol{x} = \{x_1, x_2, \dots, x_m\}^{\mathrm{T}}$, where $\boldsymbol{x} \in \mathbb{R}^m$, and $y = f(\boldsymbol{x})$ as the output. A surrogate model $\hat{f}(\boldsymbol{x})$ replicates $f(\boldsymbol{x})$ based on a finite set of experimental design (ED), $\mathcal{X} = \{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \dots, \boldsymbol{x}^{(n)}\}^{\mathrm{T}}$, where n is the size of the ED. To build a surrogate model, the observations at the ED are required, i.e., $\boldsymbol{y} = \{y^{(1)}, y^{(2)}, \dots, y^{(n)}\}^{\mathrm{T}}$ where $\boldsymbol{y} = f(\mathcal{X})$. There are some advancements in the field of surrogate modeling for mixed variables (Saves et al. 2023), but such a topic is beyond the scope of this paper. This paper uses GPR and PC-Kriging as surrogate models, as explained below.

2.1 Gaussian process regression

GPR treats a black-box function as a realization of jointly normally distributed random variables (Sacks et al. 1989; Rasmussen 2003), reads as

$$Y(\boldsymbol{x}) = \mu(\boldsymbol{x}) + Z(\boldsymbol{x}), \tag{1}$$

where $\mu(\boldsymbol{x})$ is the mean function and $Z(\boldsymbol{x})$ is a zero-mean stochastic process. The most common form is to set $\mu(\boldsymbol{x})$ as a constant, i.e., $\mu(\boldsymbol{x}) = \mu_{GP}$, which is obtained from maximum likelihood estimation.

GPR assumes that the outputs are correlated, which is modeled by the kernel function. Consider two different inputs, $\boldsymbol{x}$ and $\boldsymbol{x}'$, in which $\boldsymbol{x}, \boldsymbol{x}' \in \mathbb{R}^m$. A kernel function $k(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta})$, where $\boldsymbol{\theta} = \{\theta_1, \theta_2, \dots, \theta_m\}$ is the vector of length-scales, models the correlation between the output of $\boldsymbol{x}$ and $\boldsymbol{x}'$. More formally,

$$k(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta}) = \mathrm{corr}(y, y'; \boldsymbol{\theta}). \tag{2}$$

The most widely used kernel function is the squared-exponential, which reads as

$$k(\boldsymbol{x}, \boldsymbol{x}'; \theta) = \exp\left(-0.5 \left(\frac{\boldsymbol{x} - \boldsymbol{x}'}{\theta} \right)^2 \right). \tag{3}$$

A correlation matrix $\boldsymbol{R}$ of size $n \times n$, where $R_{ij} = k(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}; \boldsymbol{\theta})$, can be constructed after defining the kernel function. Let us also define $\boldsymbol{r}(\boldsymbol{x}) = \{k(\boldsymbol{x}, \boldsymbol{x}^{(1)}; \boldsymbol{\theta}), \dots, k(\boldsymbol{x}, \boldsymbol{x}^{(n)}; \boldsymbol{\theta})\}^{\mathrm{T}}$. The prediction of a GPR model reads as

$$\hat{y}(\boldsymbol{x}) = \mu_{GP} + \boldsymbol{r}^{\mathrm{T}} \boldsymbol{R}^{-1} (\boldsymbol{y} - \boldsymbol{1}\mu_{GP}), \tag{4}$$

where $\boldsymbol{1}$ is a vector of ones with size $n \times 1$. Due to its probabilistic treatment, GPR also outputs the uncertainty estimate of the prediction, which reads as

$$\begin{aligned} \hat{s}^2(\boldsymbol{x}) =& \sigma_{GP}^2 \big(1 - (\boldsymbol{r}(\boldsymbol{x})^{\mathrm{T}} \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x})) \\ &+ \left(1 - \boldsymbol{1}^{\mathrm{T}} \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x}) \right)^2 \left(\boldsymbol{1}^{\mathrm{T}} \boldsymbol{R}^{-1} \boldsymbol{1} \right)^{-1} \big), \end{aligned} \tag{5}$$

where σ_{GP}^2 is the signal variance. Adding a small regression factor λ to the diagonal elements of $\boldsymbol{R}$ is usual to enhance numerical stability.

Calibration of a GPR model is usually done via maximum likelihood estimation. Let us denote a vector of hyperparameters $\boldsymbol{\gamma} = \{\boldsymbol{\theta}, \mu_{GP}, \sigma_{GP}^2, \lambda\}$. The hyperparameters are calibrated to maximize the following log-likelihood function

$$\begin{aligned} \ln \mathcal{L}(\boldsymbol{\theta}) =& -\frac{n}{2} \ln(2\pi) - \frac{n}{2} \ln(\sigma_{GP}^2) - \frac{1}{2} \ln(|\boldsymbol{R}|) \\ &- \frac{(\boldsymbol{y} - \boldsymbol{1}\mu_{GP})^{\mathrm{T}} \boldsymbol{R}^{-1} (\boldsymbol{y} - \boldsymbol{1}\mu_{GP})}{2\sigma_{GP}^2}. \end{aligned} \tag{6}$$

The mean term is calculated as follows:

$$\hat{\mu}_{GP} = (\boldsymbol{1}^{\mathrm{T}} \boldsymbol{R}^{-1} \boldsymbol{1})^{-1} \boldsymbol{1}^{\mathrm{T}} \boldsymbol{R}^{-1} \boldsymbol{y}. \tag{7}$$

We employ a hybrid approach that combines the covariance-matrix adaptation evolution strategy (CMA-ES) (Hansen and Ostermeier 2001) with the Broyden–Fletcher–Goldfarb–Shanno (BFGS) (Liu and Nocedal 1989) algorithm as a local optimizer to optimize the log-likelihood function by tuning $\boldsymbol{\gamma}$. The CMA-ES implementation employs a population size of 100, a maximum iteration limit of 5000, and sets the number of stall generations to 100. Subsequently, the

BFGS algorithm utilizes the solution obtained by CMA-ES for local refinement of the log-likelihood.

2.2 Polynomial chaos-Kriging

The next surrogate model of interest is the PC-Kriging that combines PCE and GPR (Kersaudy et al. 2015), in which the former acts as the trend function for the latter (notice that we keep the name PC-Kriging instead of PC-GPR since the former is the name used in the original paper). PC-Kriging employs the PCE-based trend function, and standard likelihood-based maximization is performed at each combination of trend functions used within the least angle regression algorithm to find the Kriging hyperparameters. The role of least angle regression here is to find the best subset of polynomial bases to act as the trend function for GPR, eventually yielding a PC-Kriging model with the lowest cross-validation error. PC-Kriging holds the potential to discover a more accurate approximation model compared to PCE and GPR alone, albeit at the expense of increased computational costs. The main idea is to replace μ_{GP} with $\mu(\boldsymbol{x})$, in which the latter uses a PCE surrogate model.

The PCE trend function can then be written as (Blatman and Sudret 2011)

$$\mu(\boldsymbol{x}) = \sum_{\boldsymbol{\vartheta} \in \mathcal{I}_p} \alpha_{\boldsymbol{\vartheta}} \Psi_{\boldsymbol{\vartheta}}(\boldsymbol{x}), \tag{8}$$

where $\Psi_{\boldsymbol{\vartheta}}(\boldsymbol{x})$ is a multivariate orthogonal polynomial, $\alpha_{\boldsymbol{\vartheta}}$ is the PCE coefficient, and $\boldsymbol{\vartheta} = \{\vartheta_1, \vartheta_2, \ldots, \vartheta_m\}$, $\vartheta \geq 0$ is an index belongs to a set $\mathcal{I}_p$, i.e., $\boldsymbol{\vartheta} \in \mathcal{I}_p$. In this paper, the PCE model is constructed using Legendre polynomials, and the polynomial basis set $\mathcal{I}_p$. Consequently, it is necessary to normalize the design space to the range of $[-1, 1]^m$ prior to surrogate model construction. In this paper, the basis set $\mathcal{I}_p$ is formed by employing total order expansion (Blatman and Sudret 2011), which can be expressed as:

$$\mathcal{I}_p \equiv \{\boldsymbol{\vartheta} \in \mathbb{N}^m : \|\boldsymbol{\vartheta}\|_1 \leq p\}. \tag{9}$$

where $\|\boldsymbol{\vartheta}\|_1 = \sum_{i=1}^m \vartheta_i$. It is worth noting that we did not directly use the PCE surrogate model. Rather, it is incorporated as the trend function for GPR. This paper utilizes the least angle regression-based method to compute the vector of coefficients, given the experimental design and the maximum polynomial order in the expansion. Some changes are necessary to adjust to the non-constant trend function, especially in calculating the likelihood and prediction. Most importantly, the trend coefficients of the PCE are determined through generalized least squares, diverging from the stand-alone PCE which employs ordinary least squares (OLS). Further, the candidate polynomial basis set needs to be determined by the user, by providing the algorithm with the basis set $\mathcal{I}_p$ (in practice, by specifying the value of p). However, to avoid a lengthy paper, we refer interested readers to Kersaudy et al. (2015) for more details on PC-Kriging.

The accuracy of the model was measured by using the following leave-one-out cross-validation (LOOCV) error computed at the experimental design:

$$\varepsilon_{\text{LOO}} = \frac{1}{n}\left[\frac{\sum_{i=1}^n \left(f\left(\boldsymbol{x}^{(i)}\right) - \hat{f}_{-i}\left(\boldsymbol{x}^{(i)}\right)^2\right)}{\text{Var}[f(\boldsymbol{x})]}\right] \tag{10}$$

where $\hat{f}_{-i}(\boldsymbol{x}^{(i)})$ denotes the surrogate model constructed using all points in the experimental design excluding $\boldsymbol{x}^{(i)}$, and $\text{Var}[f(\boldsymbol{x})]$ represents the variance of the response, estimated from all available samples.

3 Explainable surrogate models for design exploration

3.1 Shapley values

First, let us discuss the concept of Shapley values before SHAP (Shapley 2016). In game theory, a coalition game is defined as a game consisting of m players. Each possible coalition is assigned with a real value that comes from a value function val(.). Let us also denote $[1 : m] \equiv 1, 2, \ldots, m$, $u \subseteq [1 : m]$, and $\{-u\} = [1 : m] \backslash u$. For a coalition u, the value function is then defined as val(u). For a regression/ML model, the players and the value function are the input variables/features and the prediction, respectively. The value of val(u) can change if a player or a group of players leave or enter the game.

Consider a player j, the corresponding Shapley value for a player j (i.e., ϕ_j) is written as

$$\phi_j = \frac{1}{m} \sum_{u \subseteq \{-j\}} \binom{m-1}{|u|}^{-1} (\text{val}(u \cup \{j\}) - \text{val}(u)), \tag{11}$$

where $(\text{val}(u \cup \{j\}) - \text{val}(u))$ is the marginal contribution of player j to a coalition u. Notice that the marginal contribution of player j is weighted and then summed for every possible coalition. An important property of Shapley value is efficiency, which essentially means that the sum of Shapley value for all players equals that of the grand coalition $[1 : m]$, that is

$$\phi_0 + \sum_{j=1}^m \phi_j = \text{val}([1 : m]). \tag{12}$$

The efficiency property is important because it eases interpretation.

In the context of design optimization and exploration, creating an explanation up to the single prediction level will

be insightful. This is because there is often an interest in analyzing how a prediction model arrives at a certain prediction. The next section details the SHAP method, which utilizes Shapley values to explain the output of any regression model in a quantitative and informative way.

3.2 SHAP

The main idea of SHAP is that it creates individual explanations (for a single prediction) that can be aggregated, hence allowing local and global explanations of the model (Lundberg and Lee 2017). SHAP can be applied to any regression or ML model, thus making it a model-agnostic explanation technique. The coalition u in SHAP consists of the subset of input variables. However, the value function in SHAP now applies for a single prediction at an arbitrary $\boldsymbol{x}$. That is, val(u) at $\boldsymbol{x}$ now equals to $\hat{f}_u(\boldsymbol{x}_u)$, where $\hat{f}_u(\boldsymbol{x}_u)$ is the prediction model that takes all variables in u as the input variables and $\boldsymbol{x}_u = (x_j)_{j\in u}$. The value function for the grand coalition equals the original prediction of the model itself, that is, $\hat{f}_{[1:m]}(\boldsymbol{x}) = \hat{f}(\boldsymbol{x})$. With this in mind, SHAP works by decomposing the prediction at $\boldsymbol{x}$ as follows

$$\hat{f}(\boldsymbol{x}) = \phi_0 + \sum_{j=1}^{m} \phi_j(\boldsymbol{x}) \tag{13}$$

where ϕ_j is the Shapley value for the j-th variable, and ϕ_0 equals the prediction without any input variables included, i.e., $\hat{f}_\emptyset(\boldsymbol{x})$. In practice, $\hat{f}_\emptyset(\boldsymbol{x})$ is usually set to the average of the regression model or the training samples. In this paper, we set $\hat{f}_\emptyset(\boldsymbol{x}) = \hat{f}(\boldsymbol{x}_m)$, where $\boldsymbol{x}_m$ is the centre of the input space. It can be seen now that the sum of the Shapley values in SHAP for all individual variables, including the empty set, equals the main model's prediction. The individual Shapley value explains how each variable contributes to the prediction. Note that the Shapley value can be either a positive or negative value, depending on how the addition of the variable affects the prediction.

To get a better grasp on the concept of SHAP, let us define u^{clo} as follows

$$u^{\text{clo}} = \bigcup_{v\subseteq u} v. \tag{14}$$

That is, u^{clo} is defined as the union of the power set of a set u, in which u itself is a subset of $[1:m]$. Next, we define $\hat{f}_u^{\text{clo}}(\boldsymbol{x}_u)$ as a predictive model that takes all combinations of inputs as in u^{clo}, written as

$$\hat{f}_u^{\text{clo}}(\boldsymbol{x}_u) = \sum_{v\subseteq u} \hat{f}_v(\boldsymbol{x}_v). \tag{15}$$

Here, the value function for a coalition u is set to $\hat{f}_u^{\text{clo}}(\boldsymbol{x}_u)$. As for the empty set and the grand coalition, we can define $\hat{f}_\emptyset^{\text{clo}}(\boldsymbol{x}_\emptyset) = \hat{f}(\boldsymbol{x}_m)$ and $\hat{f}_{[1:m]}^{\text{clo}}(\boldsymbol{x}) = \hat{f}(\boldsymbol{x})$. In other words, the value function is the prediction of the ML/regression model that only uses the variables included in u as the inputs. Finally, by using the definition of Shapley values in Eq. (11), the SHAP value for the j-th variable is defined as

$$\phi_j(\boldsymbol{x}) = \frac{1}{m} \sum_{u\subseteq\{-j\}} \binom{m-1}{|u|}^{-1} \left(\hat{f}_{u\cup\{j\}}^{\text{clo}}(\boldsymbol{x}_{u\cup\{j\}}) - \hat{f}_u^{\text{clo}}(\boldsymbol{x}_u) \right). \tag{16}$$

Equation (16) is point-wise and applies at an arbitrary input $\boldsymbol{x}$. The rightmost term in Eq. (16) is the marginal contribution of the j-th variable to a coalition u.

Calculation of SHAP is computationally intensive due to the need to construct $2^m + 1$ models. Thus, the SHAP values are usually estimated using several approximation techniques. The most common method is KernelSHAP which employs a special weighted linear regression for each feature (Lundberg and Lee 2017). Kernel SHAP estimates Shapley values by considering subsets of input features, and a weighted average is computed. The weights represent the number of ways each feature can be included in different subsets, ensuring that each feature's contribution is accounted for accurately. It is important to emphasize that performing weighted linear regression is required for each combination of input variables. Consequently, the use of KernelSHAP may incur substantial computational expenses, particularly as the number of input variables increases. Nonetheless, it is worth noting the computational cost of KernelSHAP is still typically cheaper compared to the expense associated with evaluating a single computer simulation, e.g., one computational fluid dynamics (CFD) simulation.

3.3 Explainable surrogate models

It is worth noting that the goal of design exploration is not just a mere prediction but also to extract insight. Therefore, the surrogate serves as a model that helps interpret the input–output relationship. It is worth noting that we use the term "surrogate model" as equal to the "supervised ML model". In ML literature, the term "surrogate model" usually denotes the proxy for the original ML model, e.g., to calculate SHAP. The KernelSHAP technique uses interpretable models such as linear regression to act as a proxy for the original complicated ML models. We use the term "surrogate model" in the engineering context, that is, the approximation model of the original black-box function.

The first necessary step is to build the surrogate model, which should be as accurate as possible. The validity of the knowledge extracted depends on the model's accuracy. Therefore, it is important to check the quality of the model

first before extracting any knowledge, e.g., via k-fold or LOOCV.

In this paper, we demonstrate several techniques that can be used to extract insight from SHAP values of two objectives. The main interest is to extract global trends as an aggregate of SHAP values in multiple instances. The techniques include (1) the averaged SHAP, (2) the SHAP summary plot, (3) the single-objective SHAP dependence plot, (4) the bi-objective SHAP dependence plot, and (5) the SHAP correlation matrix, as summarized in Fig. 1. The details of these techniques are discussed further in the following sections.

3.3.1 Averaged SHAP

In the context of GSA, the averaged SHAP values serve as a sensitivity metric/feature importance at the global level. The averaged SHAP for the j-th variable can be written as

$$\bar{\phi}_j = \frac{1}{n_r} \sum_{i=1}^{n_r} \left| \phi_j\left(\boldsymbol{x}^{(i)}\right) \right| \tag{17}$$

where n_r is the sample size used for calculating $\bar{\phi}_j$. In general ML applications, the training set is usually used to calculate $\bar{\phi}_j$, thus, $n_r = n$. Because we are interested not just in the training set, we calculate $\bar{\phi}_j$ by generating an independent set $\mathcal{X}_{\text{ind}}$ that is randomly generated in the input space, where $n_r \gg n$. Other alternatives for GSA include the Sobol indices and activity scores (see Appendices 1 and 2). We believe that using the aggregated SHAP values is a more natural way to assess the sensitivity of the input variables for design exploration. The reason is that the input variables are not random, making it unsuitable for applying variance-based decomposition techniques such as Sobol indices. Since SHAP is based on how the input variable contributes to the prediction, the aggregated SHAP values are more reflective of the variation in the input–output relationship of a design exploration problem (i.e., the inputs are not random variables).

3.3.2 SHAP summary plot

A SHAP summary plot presents a summary of each feature's impact on the model's output for a set of samples. A typical SHAP summary plot displays the input variables on the y-axis, ranked in descending order by their importance according to the averaged SHAP values. However, it is also possible to display the order of the variable arbitrarily, which we use in this paper. The x-axis represents the SHAP values, which quantify the impact of each input on the surrogate's predictions. By examining a SHAP summary plot, one can quickly identify which inputs significantly influence the model's predictions and understand the direction and magnitude of their impact. Furthermore, the SHAP summary plot of the two objectives can be shown side by side so that it is possible to analyze the two-directional impact (in the sense of objectives) of changing one input variable. Suppose one detects an interesting individual trend from the SHAP summary plot. In that case, one can then analyze the single- and multi-objective SHAP dependence plot to uncover the trend further, as discussed next.

3.3.3 Single-objective SHAP dependence plot

Visualization of SHAP through the dependence plot reveals important information, including the level of nonlinearity and interaction. Specifically, the single-objective SHAP dependence plot visualizes the relationship between the magnitude of the single input variable versus the corresponding SHAP value at multiple instances. While Sobol indices compute the strength of such interactions through variance decomposition, they do not reveal how the two or more variables exactly interact. On the other hand, SHAP can reveal both the magnitude and structure of the interaction through

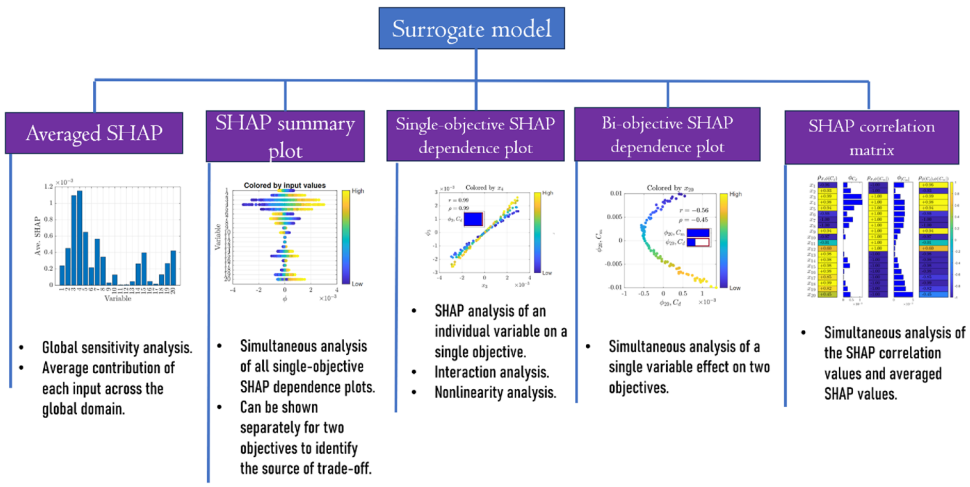


Fig. 1 A schematic illustration of several means to explore SHAP from a surrogate model for knowledge discovery

visualization. The way to visualize SHAP is by depicting one input variable and the corresponding SHAP values in the abscissa and ordinate, respectively. Furthermore, the dots are colored according to the input values of another variable that strongly interacts with the respective variable (inferred from the second-order Sobol indices); a distinct trend will appear if a sufficiently strong interaction exists.

To add more meaningful information, we suggest adding the correlation coefficients in the plot to aid in the interpretation (see Sect. 3.3.5). Further, a bar that indicates the relative strength of the averaged SHAP values of the respective variable is also added to the plot. The role of adding such a bar is to help interpret the magnitude of the strength of the input variable relative to the strongest input variable. Taking advantage of bounded variables typical in design optimization, we can plot several or all SHAP partial dependences on a single plot by first normalizing the inputs to the same scale, e.g., $[0, 1]^m$ or $[-1, 1]^m$. It is worth noting that the simultaneous depiction is only useful if the SHAP values are not too scattered (meaning weak interactions). Otherwise, the simultaneous depiction will be too cluttered.

3.3.4 Bi-objective SHAP dependence plot

Although SHAP can identify the difference in the importance of the input variables on the two or more QOIs, such knowledge can also be obtained from Sobol indices analysis. The main advantage of SHAP within a multi-criteria context is that it allows richer analyses of the input–output relationship. As shown later in the examples, SHAP enables the simultaneous analysis of the impact of changing an input variable on multiple quantities of interest. The most beneficial part of SHAP is that it can reveal how each input variable affects the prediction because it works on the level of individual prediction.

The bi-objective SHAP dependence plot shows the respective SHAP values of the first versus those of the second objective, colored by the magnitude of the input variables. The plot was first introduced in Takanashi et al. (2023) with the name "SHAP trade-off plot". In this paper, we prefer to use the term "bi-objective SHAP dependence plot" since trade-off as a function of an input variable does not always exist. This plot is particularly useful for simultaneously investigating the impact of an input variable on the two objectives. Furthermore, the bi-objective SHAP dependence plot can also reveal how the two objectives correlate to each other with respect to the contribution of the input variable being investigated. Subsequently, the plot is useful for identifying the variables that contribute to the trade-off and those that simultaneously improve the objectives.

The bi-objective SHAP dependence plot, as its name implies, is designed to illustrate the dependency of two objectives on a single input variable. This plot can be expanded into three objectives by incorporating an additional axis corresponding to the third objective, although this introduces added complexity to the analysis. An alternative approach is to generate three separate bi-objective SHAP dependence plots-namely, 1st vs 2nd, 1st vs 3rd, and 2nd vs 3rd objectives to facilitate analysis. Another possibility is to leverage data visualization tools for many-objective optimization (He and Yen 2017; Meneghini et al. 2018) and combine it with SHAP. For the scope of this paper, our emphasis is on the two-objective scenario, with the exploration of three-objective dependence plots reserved for future works in other applications.

3.3.5 SHAP correlation matrix

Let us first define the "SHAP correlation value", which measures the correlation between one of the following: (1) the correlation between input variables and the corresponding SHAP values for a single objective, and (2) the correlation between the SHAP values of the two objectives, as a function of a single input variable. The correlation can take any form; however, we use the Pearson or Spearman correlation matrix (denoted as ρ and r, respectively). Several insights can be obtained from analyzing the SHAP correlation value. First, by investigating the correlation between input variables and the SHAP values, it is possible to check whether an input is positively or negatively correlated with the output. Suppose one variable yields a perfect positive Pearson correlation with the output (i.e., $\rho = 1$). This means that increasing that particular input would lead to a linear increment in the objective (and vice versa for $\rho = -1$). A Spearman correlation can also be used to identify a monotonous relationship. The low correlation itself does not mean no correlation exists. Instead, it might signal strong interaction or neither a monotonous nor linear relationship (i.e., a more complex relation).

Alternatively, examining the SHAP correlation between the two objectives based on a specific input variable provides insight into whether changes in the input result in simultaneous improvements or the opposite. If the aim is to minimize both objectives, a perfect positive correlation indicates that the variable in question contributes to the simultaneous enhancement of both objectives. Conversely, a perfect negative correlation suggests that the input variable is responsible for trade-offs between the objectives. A low correlation signifies a more intricate relationship (or no correlation) between the two objectives concerning a single input variable.

We propose a visualization method called the SHAP correlation matrix, which encodes the two information above (either Pearson or Spearman, see Appendix 3) and the averaged SHAP values for the two objectives. The corresponding coloring also accompanies the correlation value to ease the

reading. The SHAP correlation matrix does not visualize the possible interaction and nonlinearity; however, one can quickly analyze the correlation and possible interaction/nonlinearity in a single plot. Suppose there are some interesting variables similar to the SHAP summary plot. In that case, one can then visualize the individual single-objective SHAP dependence plot or the bi-objective SHAP dependence plot for further exploration.

All experiments in this paper were executed using MATLAB^{TM} R2022a with the personal computer specification as follows: Processor 11th Gen Intel(R) Core(TM) i7-1165G7 @ 2.80 GHz, 2803 Mhz, 4 Core(s), 8 Logical Processor(s) and 16 GB RAM (see Replication of Results for more details).

3.4 Pedagogical demonstration: four-bar trusses test problem

The capabilities of SHAP are first demonstrated on a simple four-bar test problem (see Fig. 2) adopted from Stadler (1988). The goal is to minimize the weight and the vertical displacement of the outer loaded node subjected to several load conditions, expressed as follows:

$$\begin{aligned} f_1(\boldsymbol{x}) &= L\left(2x_1 + \sqrt{2}x_2 + \sqrt{x_3} + x_4\right) \\ f_2(\boldsymbol{x}) &= \frac{FL}{E}\left(\frac{2}{x_1} + \frac{2\sqrt{2}}{x_2} - \frac{2\sqrt{2}}{x_3} + \frac{2}{x_4}\right) \end{aligned} \tag{18}$$

where f_1 is the weight of the truss system, f_2 is the displacement, E is the elastic modulus, L is the length of a truss section, and σ is a characteristic stress. The design variables, which correspond to the cross-sectional areas, are defined as follows: $(F/\sigma) \le x_1 \le 3(F/\sigma)$, $\sqrt{2}(F/\sigma) \le x_2 \le 3(F/\sigma)$, $\sqrt{2}(F/\sigma) \le x_3 \le 3(F/\sigma)$, and $(F/\sigma) \le x_4 \le 3(F/\sigma)$, where $F = 10$ kN, $E = 2 \times 10^5$ kN/cm^2, $L = 200$ cm, and $\sigma = 10$ kN/cm^2. The problem originally involves a constraint, but we neglect the constraint since this paper focuses on the objective functions for design exploration.

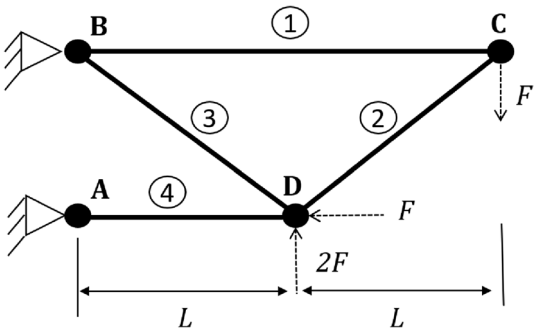
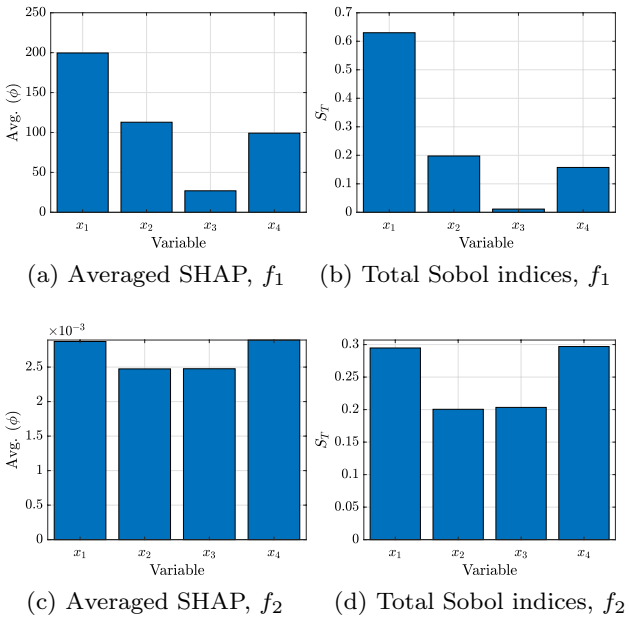


Fig. 2 A schematic illustration of the four-bar trusses problem

The first objective is linear in terms of the input variables. On the other hand, f_2 varies reciprocally with the input variables, which becomes the origin of the objectives trade-off. The simple form of the four-bar truss problem makes it a good test case for demonstrating the capabilities of SHAP for multi-objective design. Therefore, the SHAP extracted from a surrogate model should be able to reflect such trends well. GPR models with 200 samples from Latin hypercube sampling (McKay et al. 2000) were built to approximate the two objectives, yielding a highly accurate model with LOOCV error in the order of 10^{-4} for both weight and displacement. The SHAP values are sampled from 10,000 realizations on the input space, primarily for calculating the averaged SHAP. This example shows various methods to visualize SHAP values for the two objectives. However, a more comprehensive visualization and comparison are given and discussed on the non-analytical problems shown in Sect. 4.

Let us begin by comparing the GSA metric derived from total indices and averaged SHAP as shown in Fig. 3. For the sake of simplicity, we do not show the activity scores and uncertainties associated with the GSA metrics for this problem. The wall-clock time to compute Sobol indices using 10,000 samples is about 2 s while the cost for the averaged SHAP using the same amount of samples is about 43 s. The averaged SHAP gives the same ranking as the total Sobol indices, but the interpretation is different. Note that the averaged SHAP values quantify the average contribution of an



(a) Averaged SHAP, f_1 (b) Total Sobol indices, f_1

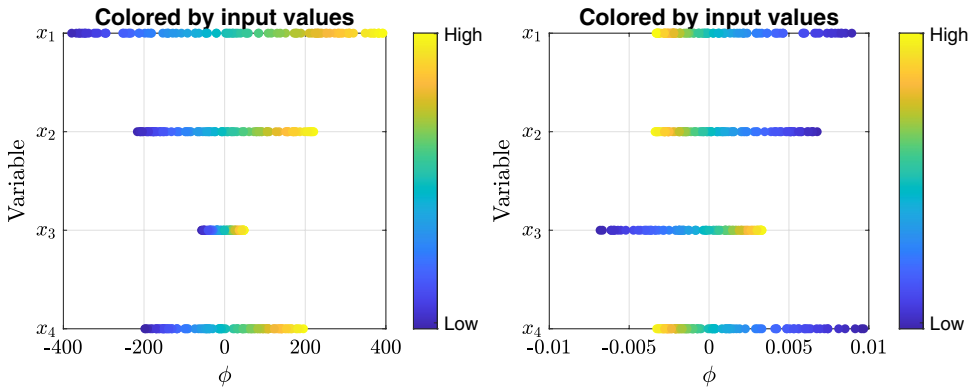
(c) Averaged SHAP, f_2 (d) Total Sobol indices, f_2

Fig. 3 Averaged SHAP barplot and total Sobol indices of f_1 and f_2 for the four-bar problem

input to the prediction. In contrast, Sobol indices quantify the impact of input variables and their interactions with the total variance, which is a squared quantity. In the context of the design exploration problem, we think that one obtains more useful information from how the design variables change the output (on average) compared to partial variances. For example, the most significant variable for f_1, i.e., x_1, yields an average SHAP contribution to the weight with a magnitude of 200.44. The third variable contributes only 26.95 to the weight, which is clear from the expression shown in Eq. (18) and the corresponding upper and lower bounds. On the other hand, the total Sobol indices yield $S_{T,x_1} = 0.63$ and $S_{T,x_3} = 0.01$. Averaged SHAP makes more sense in the context of design exploration because it maintains the same unit as the objective function.

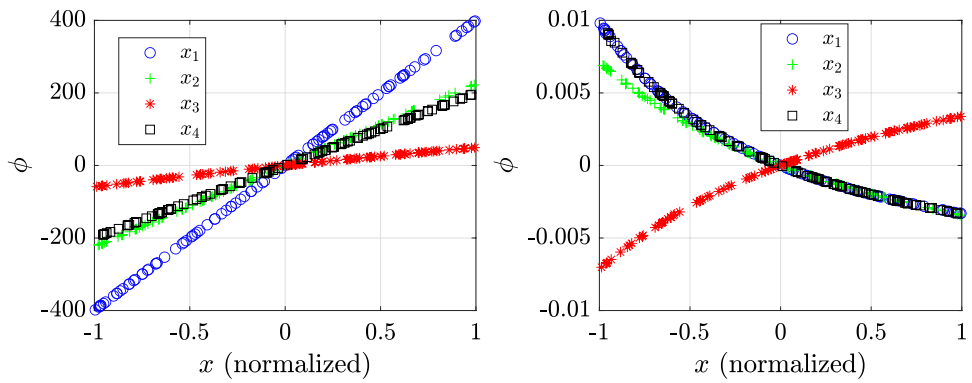
The SHAP summary plot for the four-bar problem is shown in Fig. 4. This plot basically shows the direction of the change in the objective function with respect to the input variables. What is clear from this plot is that both objectives change monotonously as the function of each input variable, as indicated by the gradual change in color. In essence, it is clear that increasing the cross-sectional area results in a heavier system (the SHAP value of f_1 increases when the design variable is increased). On the other hand, increasing the cross-sectional area of the first, second, and fourth truss (i.e., x_1, x_2, and x_4) would reduce the displacement. The trend for the third bar (i.e., x_3) is the opposite; decreasing its cross-sectional area would lead to smaller displacement. The reason is that reducing the flexibility of the third bar leads to a larger absorption of the applied loads, leading to a redistribution of forces within the structure. The knowledge extracted from the SHAP summary plot then matches with that from basic statics.

To further analyze the dependency of the objectives on each input variable, Fig. 5 shows simultaneously the SHAP dependence plots for f_1 and f_2, which depict the average impact of changing all input variables on the two objectives separately (shown in the normalized inputs for easier depiction). It can be seen that all variables impact the weight (f_1) linearly (see Fig. 5a), with x_1 coming out as the most important variable. Further, increasing all input variables straightforwardly leads to the increase of f_1, which makes sense since increasing cross-sectional area leads to higher weight. It is not trivial to tell from the SHAP summary plot that f_2 varies reciprocally, but the plot clearly shows that the input variables affect f_2 in a more nonlinear fashion compared to



(a) f_1 (b) f_2

Fig. 4 SHAP summary plot for f_1 and f_2, the four-bar problem



(a) f_1 (b) f_2

Fig. 5 Combined SHAP dependence plot of f_1 and f_2 for the four-bar problem

f_1. The SHAP dependence plot for f_2 also further reveals that all variables but x_3 negatively correlated with f_2. The figures also show that there are no interactions between input variables for both f_1 and f_2 since no dispersion is observed in the SHAP values; this is clear from the expression in Eq. (18). This is also why all four variables are shown simultaneously in the SHAP dependence plot since the number of variables is few and no interactions are involved, so the plot would not be too cluttered. It is clear from the SHAP dependence plot that the functions can be expressed as an additive model. In essence, the knowledge obtained from SHAP matches the expression of the four-bar problem.

We now move to discuss the four-bar problem from the multi-objective design viewpoint. Figure 6 shows the bi-objective SHAP dependence plots of f_1 versus f_2 with respect to x_1 and x_3 in two separate plots. These two variables are selected as representatives because x_1 is the most important variable for both objectives, while x_3 displays an interesting trend. The plots are colored with respect to the magnitude of the respective inputs. From Fig. 6a, it can be seen that the increase in x_1 leads to the increase in f_1 and a decrease in f_2 (pay attention to the color gradation). The plot shows that the change in x_1 leads to a clear trade-off between f_1 and f_2. Indeed, the correlation between the objectives with respect to the change in x_1 is nearly linear and strictly decreasing or increasing, as evidenced by the perfect Spearman coefficient and sufficiently high Pearson coefficient (both are negative values). On the other hand, the change in x_3 positively correlates with f_1 and f_2. Thus, if we isolate the impact of x_3, a decrease in weight due to x_3 also decreases displacement. From the pure viewpoint of the two objectives defined, the variable x_3 is not the source of the trade-off between the objectives, although its impact on f_1 is small (see the inset bar). It can then be said that the value of x_3, without involving constraints, should be purely decreased to lead to optimal designs. It is also worth noting that the impact of x_1 is significantly larger than x_3 in changing f_1

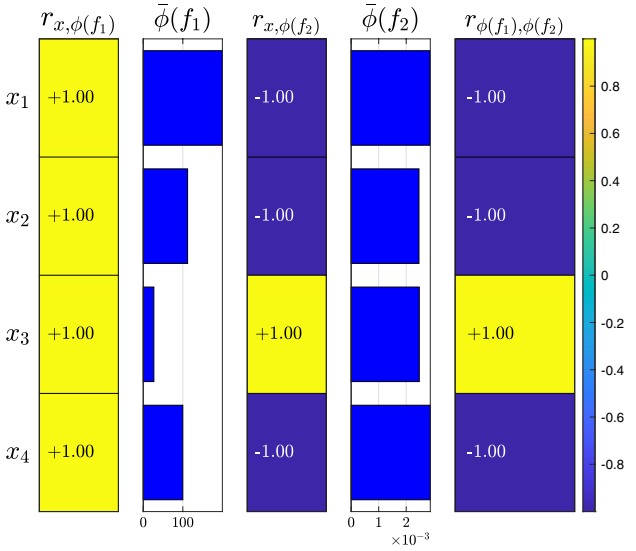
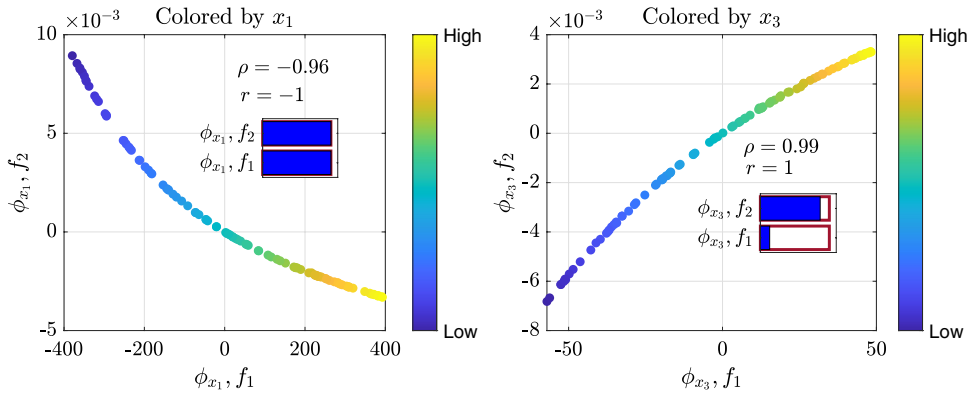


Fig. 7 SHAP correlation matrix for the four-bar trusses problem

(see the different scale in the x-axis). Similar plots can also be created for x_2 and x_4; however, the plots are not depicted since the trend is similar to that of x_1.

Finally, the SHAP correlation matrix with Spearman correlation coefficient is shown in Fig. 7. The SHAP correlation matrix simultaneously shows all the information mentioned above. The advantage is that one can simultaneously see how changing all cross-sectional areas affects the weight and displacement and how it affects the two objectives simultaneously. It is important to interpret the correlation matrix in conjunction with the averaged SHAP values to analyse the dependency and magnitude comprehensively.

This simple demonstration shows how SHAP values are used for trade-off analysis in a multi-criteria design problem. The next section further demonstrates the capabilities of SHAP on two non-analytical case studies, in which the responses of interest are evaluated using computational simulations.



(a) x_1. (b) x_3.

Fig. 6 Bi-objective SHAP dependence plot for the four-bar problem according to x_1 and x_3

Table 1 Design variables for the inviscid transonic airfoil problem

Variable	l_b	u_b	Definition
r_{LE}	0.0065	0.0092	Radius of leading edge
x_{up}	0.3466	0.5198	Upper crest absissca
y_{up}	0.0503	0.0755	Upper crest ordinate
$y_{xx_{up}}$	− 0.5094	− 0.3396	Upper crest curvature
x_{lo}	0.2894	0.4342	Lower crest absissca
y_{loo}	− 0.0707	− 0.0471	Lower crest ordinate
$y_{xx_{lo}}$	0.5655	0.8483	Lower crest curvature
α_{TE}	− 0.1351	− 0.0901	Trailing edge direction
β_{TE}	0.1317	0.1975	Trailing edge wedge angle

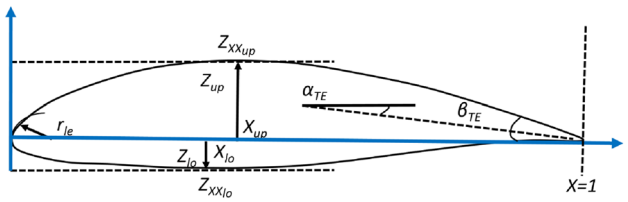


Fig. 8 A schematic illustration of the PARSEC parameterization used in the inviscid transonic airfoil problem

4 Case studies

This section provides a comprehensive discussion of the abilities of SHAP in investigating various engineering design optimization and exploration problems. SHAP values are estimated for all these problems using various surrogate models. It is important to note that the accuracy of the model used for extracting SHAP values is crucial for obtaining reliable information. The comparison of SHAP with other GSA methods, namely, Sobol indices and ASM is also carried out. The ASM is particularly intriguing as it serves as both a design exploration and GSA method, making it a valuable point of comparison with SHAP.

4.1 Case 1: nine-variable inviscid airfoil design problem

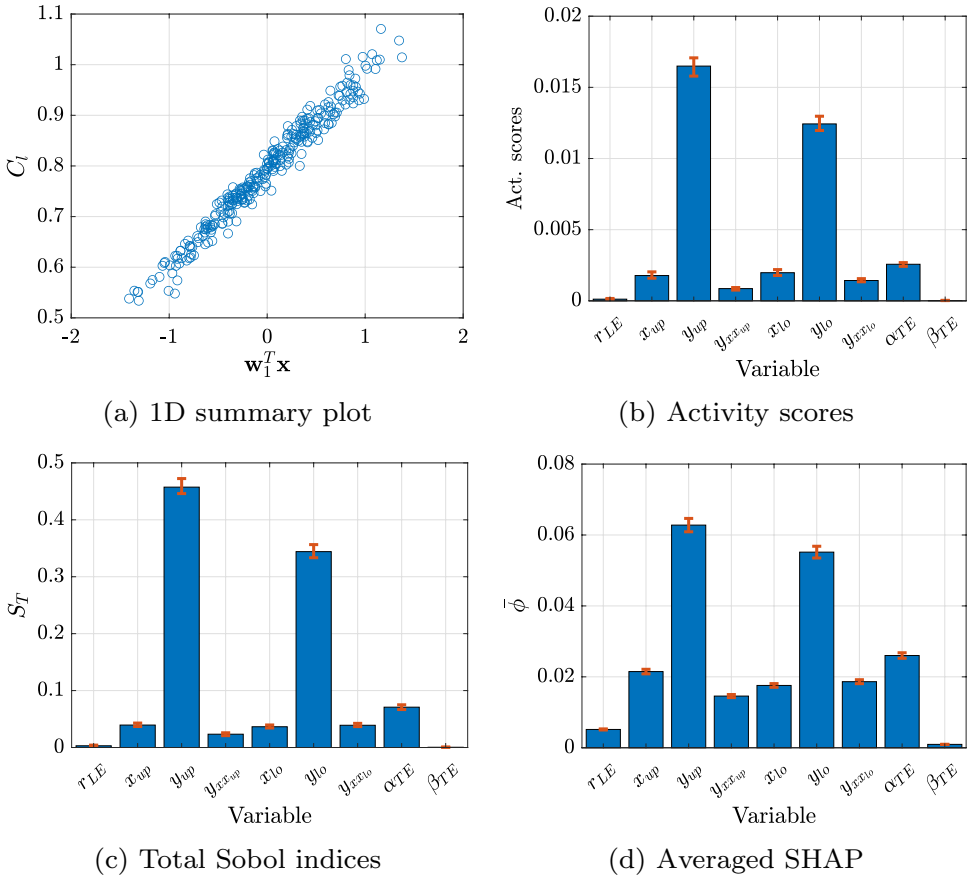
The first non-analytical test problem is an inviscid airfoil design problem with nine-variable PARSEC parameterization (Sobieczky 1999) (see Fig. 8). There are two outputs of interest, namely, the lift coefficient (C_l) and drag coefficient (C_d) evaluated at a fixed angle of attack (AoA) of 2° and Mach number of 0.73. The inviscid solver from an open-source CFD code SU2 (Economon et al. 2016) is used to evaluate the aerodynamic coefficients, with the upper and the lower bounds of the variables shown in Table 1. With our computing resources, a single Euler simulation takes approximately one minute. The first goal of this problem is to investigate the impact of the geometrical variables on the quantities of interest in terms of nonlinearity and strength. The second goal is to investigate the interactions between the two quantities of interest when the input variable is changed. This problem serves as a good benchmark case because we already know the underlying inviscid aerodynamic phenomenon; thus, we can critically assess the insight obtained from SHAP analysis.

GPR models for each output of interest were constructed using an experimental design set with $n = 315$ generated from Latin hypercube sampling (McKay et al. 2000), yielding normalized LOOCV errors equal 6.3×10^{-3} and 7.9×10^{-3} for C_l and C_d, respectively. This indicates that the models are highly accurate, which is deemed sufficient for knowledge extraction via SHAP. An external sampling set consisting of 10,000 samples was generated to compute the mean SHAP values. However, we only use a few samples to depict the dependence plots to avoid too cluttered plots. For this problem, the average time of SHAP calculation for a single combination of input variables is about 0.02 s. Hence, the time required to compute the averaged SHAP from 10,000 samples is roughly about 200 s. In contrast, the calculation of Sobol indices and activity scores from the model using 10,000 samples only took about 9 and 10 s, respectively.

4.1.1 Global sensitivity analysis

The accuracy of the GPR models, although high, is not as accurate as the pedagogical problem discussed earlier. Consequently, the GSA metrics need to be supplemented with the uncertainty linked to the random sampling points. To address this, 95% confidence intervals were established through bootstrapping (Dubreuil et al. 2014), involving 50 repetitions, each was constructed from sampling with replacement from the experimental design. The results are shown in Figs. 9 and 10, together with the active subspace summary plots. The narrow band of uncertainty associated with the obtained GSA metrics indicates a level of precision that allows considering the metrics as accurate.

First, the averaged SHAP values are computed for the two objectives and compared with activity scores and Sobol indices. The mean SHAP values generally agreed well with the total Sobol indices and activity scores in terms of variable ranking. In this regard, all GSA metrics indicate that y_{up} is the most important input for C_l and C_d. The two next important variables for C_d are x_{up} and y_{lo}. On the other hand, GSA shows that y_{lo} and α_{TE} are the next important variables for C_l. The interpretation of the averaged SHAP goes as follows. Take y_{up} as an example, this means that, on average, y_{up}



(a) 1D summary plot (b) Activity scores

(c) Total Sobol indices (d) Averaged SHAP

Fig. 9 One-dimensional ASM summary plot and activity scores for the inviscid transonic airfoil problem, C_l case

affects C_d by the order of close to $\times 10^{-2}$ (similar interpretation goes for C_l). The activity score is probably the hardest to interpret since it is based on the direction of the variables.

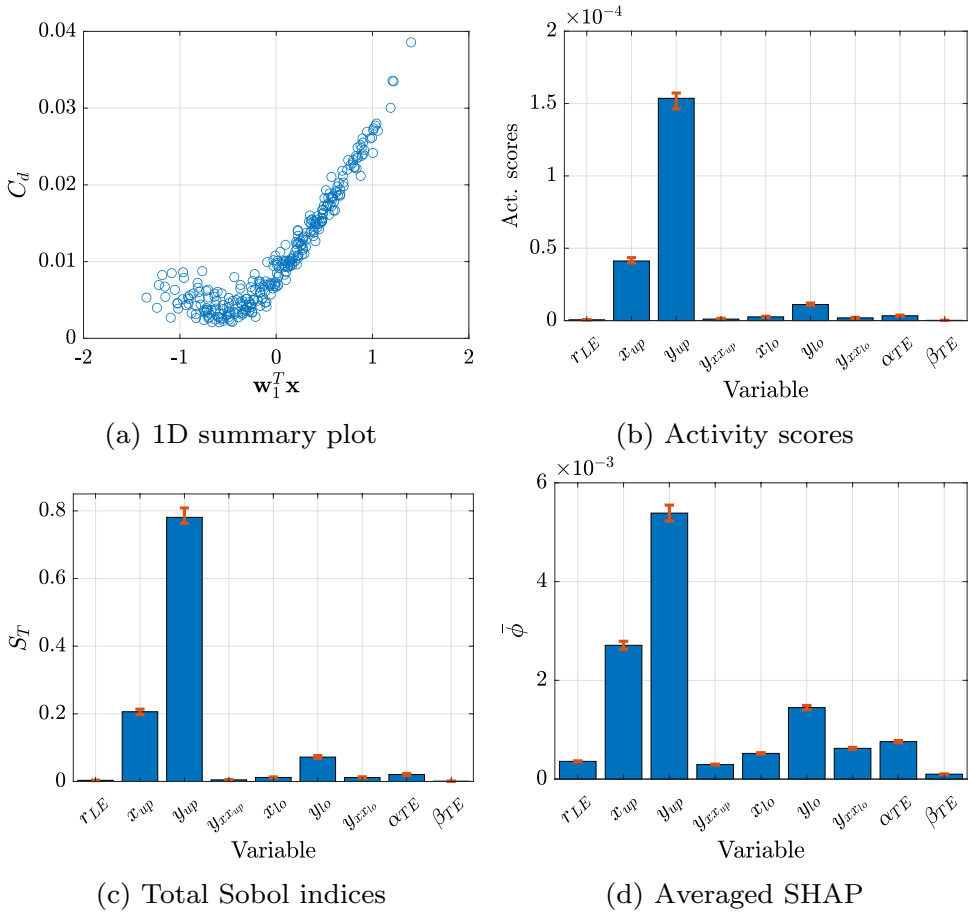
The lift response shows a strong, almost linear, one-dimensional active subspace (see Fig. 9a). Therefore, the component of the 1st eigenvector is sufficient to show the impact and the direction of how the variables affect lift. Interestingly, the drag response also shows a strong one-dimensional active subspace (see Fig. 10a). Unlike lift, however, the trend in the one-dimensional active subspace for drag is not monotonous. The nonlinear drag response exhibits a valley of local optimum, indicating a single global optimum of C_d for this problem. However, the disadvantage of the active subspace summary plot is that it does not reveal which variables contribute to such nonlinearity. Furthermore, the capabilities of the active subspace method to reveal how variables interact are limited. As explained next, SHAP analysis helps in deciphering such information.

4.1.2 SHAP summary plot

The SHAP summary plots are shown in Fig. 11. From these plots, we can see trace of nonlinearity due to some variables, e.g., x_{up}, in how of they affect lift and drag production (pay attention to the disordered coloring). However, as shown later in the SHAP dependence plot, y_{up} also affects the drag nonlinearly to a certain degree. The plots also show that the SHAP values of C_l and C_d for y_{up} and y_{lo} positively correlated with their respective inputs. Such a trend indicates that increasing y_{up} and y_{lo} increases lift and drag. The result is as expected since the increase in y_{up} and y_{lo} leads to positive camber, thus increasing lift and drag. On the other hand, the plot also reveals that α_{TE} negatively correlated with lift and drag (the trend is also clearly linear). The next section further discusses the SHAP dependence plots to analyze the general findings from the SHAP summary plot.

4.1.3 SHAP values visualization

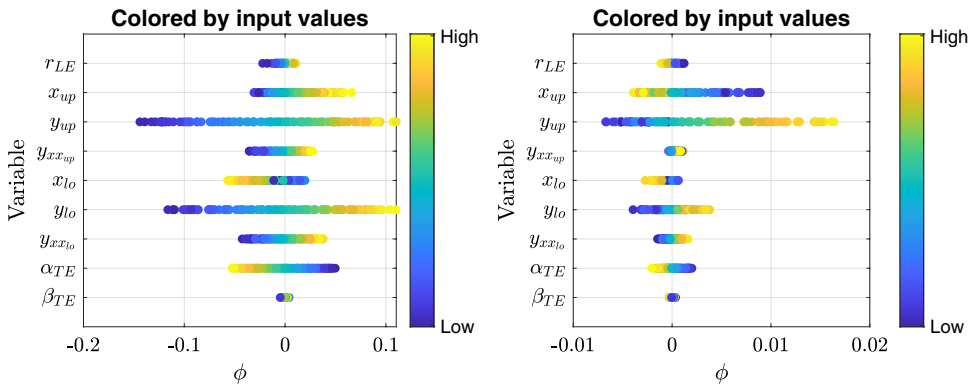
Figures 12 and 13 show the single-criteria SHAP dependence plot of selected variables for C_l and C_d (note that the figures show the normalized inputs). In particular, x_{up} and y_{up} are chosen due to their strong impact on C_d. On the other hand, y_{lo} is selected due to its strong impact on C_l. Unlike the four-bar case, the drag response features a relatively strong interaction between variables, as indicated by the dispersed values of SHAP for the three representative variables. Conversely, the lift response displays weaker interactions, except for x_{up}. Upon analyzing the SHAP dependence plots, we can



(a) 1D summary plot (b) Activity scores

(c) Total Sobol indices (d) Averaged SHAP

Fig. 10 One-dimensional ASM summary plot and activity scores for the inviscid transonic airfoil problem, C_d case



(a) C_l (b) C_d

Fig. 11 SHAP summary plots for C_d and C_l, the inviscid transonic airfoil case

see the combination of the average effect of y_{up} and x_{up} on drag (with positive and negative correlation, respectively), together with strong interactions, lead to the nonlinearity on the drag response. The reason why both variables are important in drag production is that they control the location and magnitude of the shock wave. On the other hand, in general, the single-criteria SHAP dependence plots for C_l reveal that the input variables tend to change the lift linearly. However, the change in x_{up} affects lift nonlinearly, depending on how it interacts with y_{up}. We observe that the trend in C_l due to x_{up} becomes more nonlinear if the value of y_{up} is high, and vice versa for the drag; this observation is not evident from the active subspace summary plot since it focuses on finding a representative one-dimensional subspace.

The bi-objective SHAP dependence plot is discussed next (see Fig. 14). Let us first consider x_{up}, in which the plot makes it feasible to simultaneously show the averaged effect of x_{up} on C_l and C_d, and also the interaction between

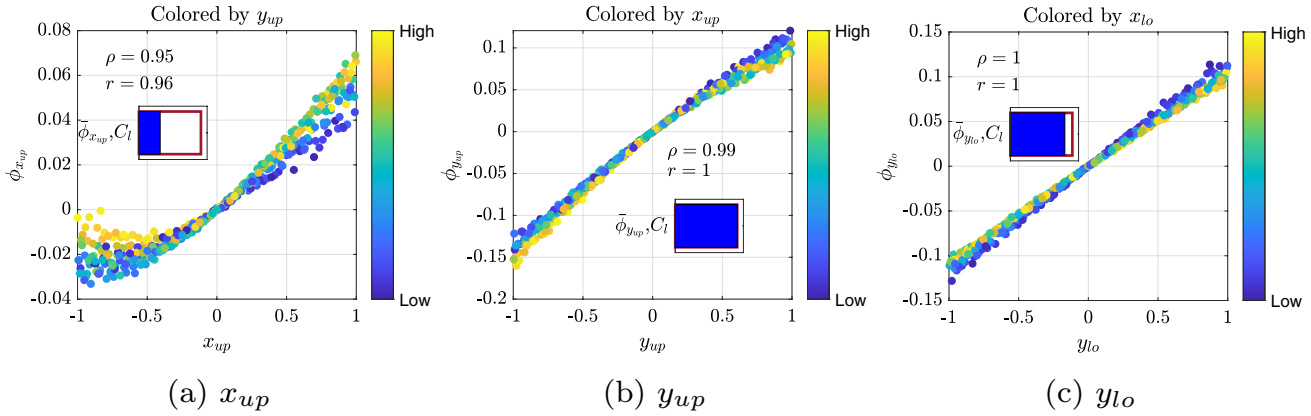


Fig. 12 SHAP dependence plots of select variables for the inviscid airfoil problem, C_l case

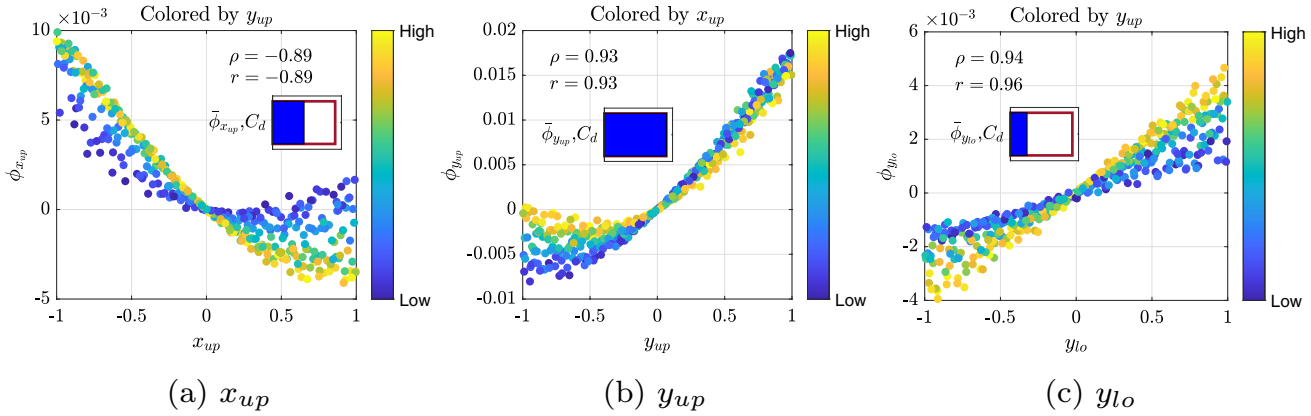


Fig. 13 SHAP dependence plots of select variables for the inviscid airfoil problem, C_d case

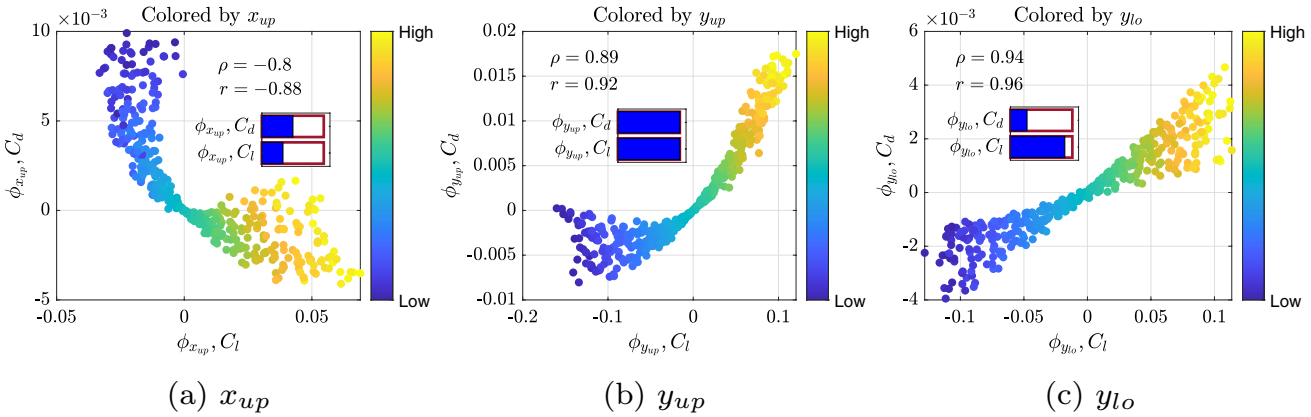


Fig. 14 Bi-objective SHAP dependence plots of select variables for the inviscid transonic airfoil problem

them. It is particularly interesting to see the impact of x_{up}, in which C_l and C_d are negatively correlated (thus, simultaneous improvement), but the trend becomes rather complex for either low or high C_l (achieved by moving x_{up} to the upstream or downstream direction, respectively) due to the interaction between x_{up} and other variables. The higher dispersion of SHAP for x_{up} on the high C_l and low drag region indicates that controlling x_{up} to achieve such characteristic should also be done with the other variables (due to the interaction). The bi-objective SHAP dependence plot

shows the tendency for a trade-off between both objectives when controlling y_{up}. However, one should also pay attention to the interaction between y_{up} and the x_{up} since the latter negatively correlates with drag. Finally, a positive correlation which indicates trade-off is observed in the bi-objective SHAP values due to y_{lo} (i.e., increasing y_{lo} would increase lift but also increase drag).

4.1.4 SHAP correlation matrix

The SHAP correlation matrix with Spearman correlation coefficient for the inviscid transonic airfoil design is shown in Fig. 15. From this plot, one can simultaneously see the SHAP correlation for all variables, which eases analysis. In particular, only two variables negatively correlate with lift, i.e., x_{lo} and α_{TE}. On the other hand, the correlations are more complex for the drag, with alternating sign and near-zero correlation. The near-zero correlation does not always indicate that the variables are not affecting the objectives. Rather, it might indicate that a strong interaction exists.

The intricate relationship between drag components ultimately results in complexity within the bi-objective correlation. Most variables show a positive correlation, indicating the change in these variables leads to the trade-off between objectives. Conversely, variables exhibiting negatively correlated bi-objective SHAP values indicate that these variables are the source of simultaneous improvement in the two objective functions (most notably x_{up}). The same thing can also be said for r_{le}; however, note again that its overall impact on drag and lift is minuscule.

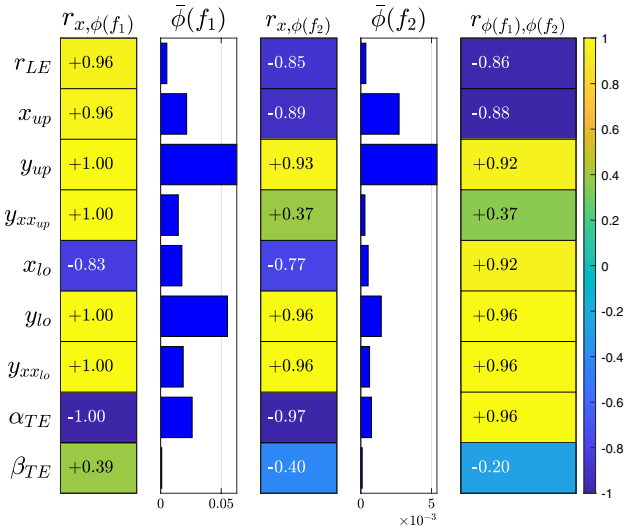


Fig. 15 SHAP correlation matrix for the inviscid transonic airfoil design (f_1 and f_2 denote the C_l and C_d, respectively)

4.2 Case 2: viscous transonic airfoil

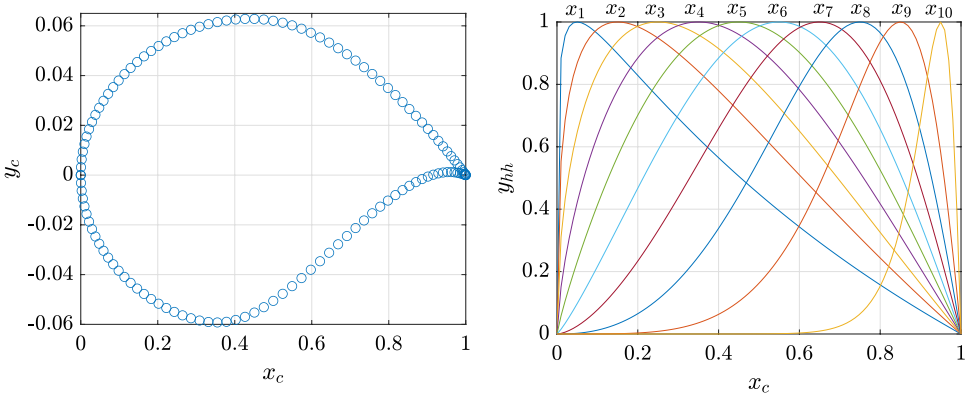
The next test case is the design of a viscous transonic airfoil (i.e., the RAE 2822 airfoil) under geometrical change parameterized by 20-variable Hicks–Henne bump function (Hicks and Henne 1978), see Fig. 16a. The Hicks–Henne bump function alters the geometry of the airfoil by a set of disturbance functions. Let us denote the abscissa and ordinate of the airfoil as x_c and y_c, respectively, with k as the number of bump functions. The expression for y_c, given the base ordinate $y_{c,base}$ reads as

$$\bar{y}_c = y_{c,base} + \sum_{i=1}^{k} c_i \sin^{w_i}(\pi x_c^{v_i}) \tag{19}$$

where

$$v_i = \ln(0.5)/\ln(x_{J_i}), \tag{20}$$

with x_{J_i} and w_i correspond to the location of the maxima and the width of the basis function, respectively. The design variables are the vector of coefficients $\boldsymbol{c} = \{c_1, \ldots, c_k\}$. The abscissa locations of the bump function are set from x_{J_1} to $x_{J_{10}}$ in a step of 0.05, while w_i is fixed to $w = 3$ for all variables. Hence, there are 10 basis functions on each surface (i.e.,



(a) Coordinates (b) Hicks-Henne function

Fig. 16 The coordinates of the RAE 2822 airfoil and visualization of the 10 Hicks–Henne bump functions

a total of 20 design variables). The first ten Hicks–Henne bump functions (scaled to unity) are visualized in Fig. 16b. Shown in the figure are the Hicks–Henne bump functions for the upper surface, while the lower surface (i.e., x_{11} to x_{20}) are simply the mirror of their upper surface counterparts. Note that x_1 and x_{11} are located on the leading edge of the airfoil. The range of the bumps' magnitude is set to $[-0.003, 0.003]$[20]. It is worth noting that the value of the Hicks–Henne bump function corresponds to their scale. In other words, positive coefficients, regardless of the upper and lower surface, always lead to thicker airfoils.

The outputs of interest are the C_d and absolute moment coefficient measured at 0.25 chord (i.e., $|C_m|$), both are to be minimized, evaluated using the Reynolds-Averaged Navier–Stokes (RANS) solver with Spalart–Allmaras turbulence model from an open-source SU2 CFD code. The case is evaluated at a fixed C_l condition, i.e., $C_\mathrm{l} = 0.723$, at a Mach number of 0.729. As such, the solver automatically tunes the angle of attack to satisfy the lift constraint. The wall-clock time elapsed for a single RANS evaluation is about 20 min. This problem was first studied in the context of preference-based multi-objective of a transonic airfoil (Palar et al. 2018). In this paper, we aim to shed light on the impact of design variables on objective functions rather than pure optimization per se. PC-Kriging was preferred for this problem due to its higher accuracy compared to GPR. The PC-Kriging uses total order truncation with a maximum polynomial order of $p = 2$.

The PC-Kriging model for C_d yielded a LOOCV error of 0.0631, sufficient for extracting knowledge from the model. The PC-Kriging model for $|C_m|$ is even more accurate with LOOCV error equals 9.6×10^{-3}. The fact that the model is accurate despite the relatively high dimensionality of the problem indicates that the responses are either linear or only have a few effective dimensions. The SHAP plots, in conjunction with the active subspace plot, are used to reveal such knowledge.

4.2.1 Global sensitivity analysis comparison

Evaluating SHAP for this problem is expensive, with each SHAP evaluation taking approximately 0.65 s. Consequently, the assessment of SHAP using 1000 samples consumes around 10.8 min. Due to this constraint, we opted for 1000 samples to estimate averaged SHAP values, noting that even with this reduced number, SHAP values appear to converge effectively. In contrast, computing activity scores and total Sobol indices using 10,000 samples requires only about 120 and 30 s, respectively.

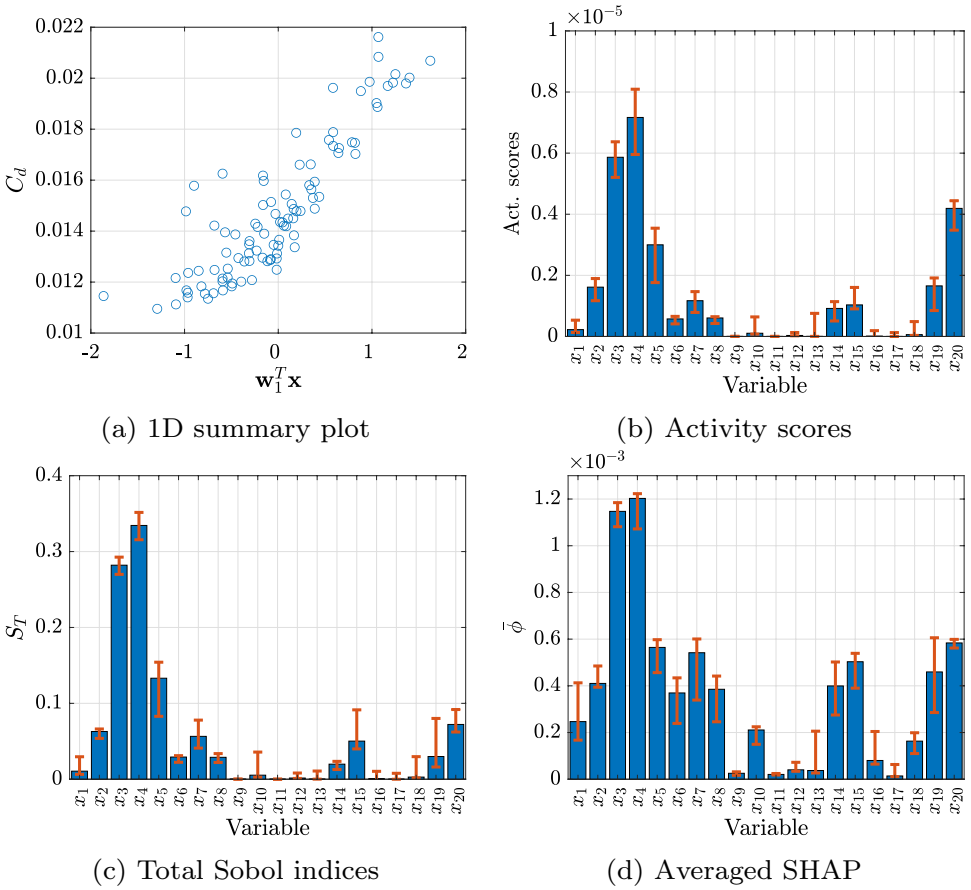
First, as shown in Fig. 17a, the active subspace summary plot indicates that the trend of C_d is not monotonous. Instead, there is an evident trend of nonlinearity, as indicated by the valley of minimum in the one-dimensional summary plot. However, the active subspace summary plot does not indicate which variable is the source of such nonlinearity. Such knowledge can be useful to identify which part of the airfoil geometry leads to such change, as revealed by the SHAP dependence plot shown later. Furthermore, inferring the direction of the variable's impact is difficult because the problem does not feature a strong one-dimensional active subspace.

A comparison of the three GSA metrics for C_d reveals a slightly different trend (see Fig. 17b–d). It is worth noting that the uncertainty derived from the bootstrap with 50 samples is relatively high compared to the previous problem. Nevertheless, all metrics agree that x_3 and x_4 are crucial in drag production, followed by x_{20} or x_5, in which the three metrics differ in how they interpret the third important variable. The interpretation is easier for total Sobol indices and averaged SHAP than the activity scores. One particular challenge in understanding activity scores is that their values are sensitive to the range of the input variables. At the same time, this is not the case for total Sobol indices and averaged SHAP. It is also worth noting that activity scores and total Sobol indices are squared quantities. Hence, the contribution of variables appears larger for the averaged SHAP than for the others.

On the other hand, the trend of $|C_m|$ is monotonous, as shown by the 1D summary plot in Fig. 18a. The insight from ASM is perhaps sufficient for trend identification of $|C_m|$. There is no urgent need to perform a deeper nonlinearity analysis regarding which variable contributes to nonlinearity since the trend is extremely close to linear. A comparison of the three GSA metrics also shows clear agreement (see Fig. 18), with the uncertainty being sufficiently small to draw meaningful conclusions.

4.2.2 SHAP summary plot

The SHAP summary plots of the objectives for the viscous airfoil problem are shown in Fig. 19. By comparing the two summary plots, it is possible to see, in general, which variables contribute to the trade-off and those that lead to the simultaneous improvement in C_d and $|C_m|$. First, it can be seen that the set of Hicks–Henne functions on the upper surface (i.e., x_1 to x_{10}) is primarily responsible for drag production. It is also interesting to see that the direction of drag change is not the same for all upper surface variables (i.e., x_1 to x_{10}). Decreasing the three most important variables for drag (i.e., x_3 and x_4) leads to decreased drag. This basically means that large drag reduction can be achieved by reducing the upper surface of the airfoil in between the leading edge and maximum thickness location (i.e., 37.9% of the chord). However, it is worth noting that increasing x_1, which primarily corresponds to increasing the leading edge radius, leads to reduced drag.



(a) 1D summary plot (b) Activity scores

(c) Total Sobol indices (d) Averaged SHAP

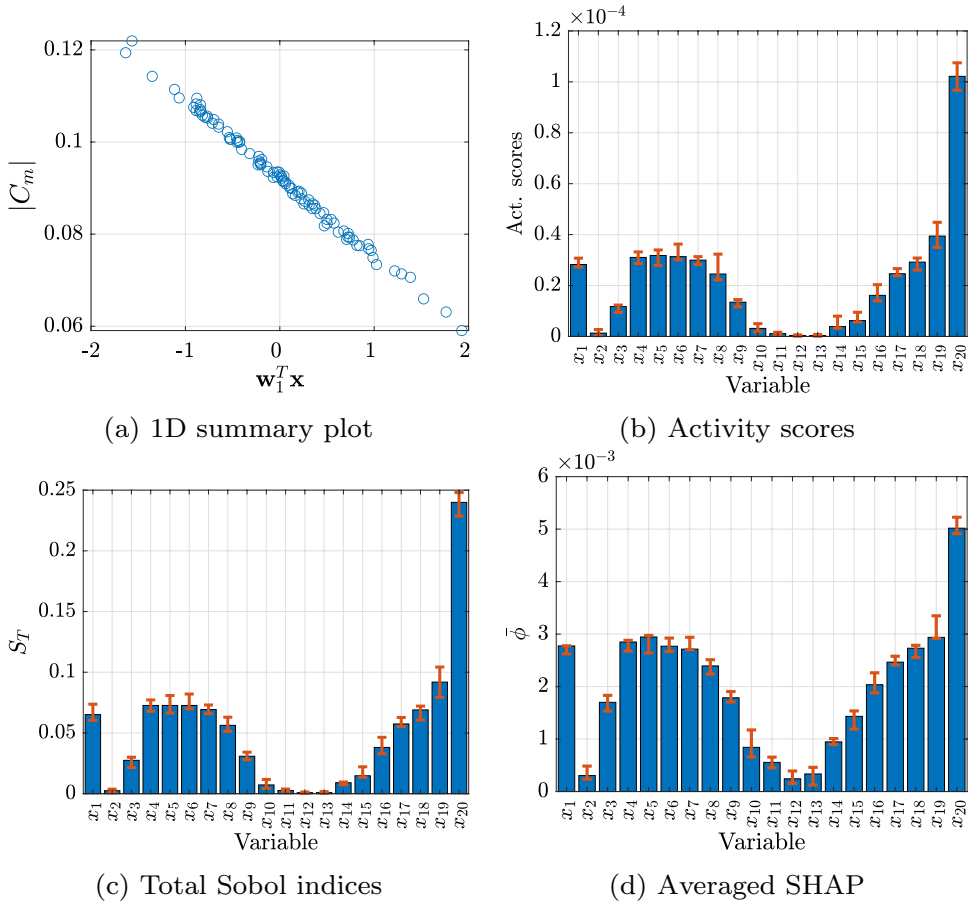
Fig. 17 One-dimensional ASM summary plot and global sensitivity indices for the viscous transonic airfoil case, C_d case

The trend is more evident and linear for $|C_m|$. Analysis of the impact of the Hicks–Henne bump functions on $|C_m|$ should be understood in the sense of how they affect the pressure distribution changes (subsequently, the centre of pressure). In this regard, the switch in the trend (i.e., from x_2 to x_3 and x_{12} to x_{13}) is due to the respective location of the bump to the 1/4 chord. It makes sense that increasing the lower surface Hicks–Henne bump function leads to reduced $|C_m|$, and the trend is persistent for almost all lower surface variables, but x_{11} and x_{12} (i.e., those that are closer to the leading edge). The impact on $|C_m|$ is substantially higher on the trailing edge of the lower surface variables. In comparison, middle-upper surface variables are also important for the pitching moment due to their proximity to the shock wave location. In general, increasing the upper surface leads to a higher absolute moment coefficient.

4.2.3 SHAP values visualization

The SHAP dependence plots of selected variables for C_d and $|C_m|$ (i.e., x_4, x_{14}, and x_{20}) are shown in Figs. 20 and 21, respectively. Also shown in the dependence plots is the relative impact of the respective variables on the objective in the form of a barplot. Note that the impact is measured relative to the most impactful variable (i.e., the most impactful variable yields a full-filled bar). The variable x_4 is chosen since it contributes the most to drag production, while x_{14} and x_{20} are selected due to their nonlinearity and high impact on $|C_m|$. It can be seen that x_4 affects the drag almost linearly. However, the impact of the interacting variable can be observed. That is, the average impact of changing x_4 decreases when the magnitude of the interacting variable (i.e., x_5) is decreased. In other words, greater reduction in C_d can be achieved by controlling x_4 and x_5 together.

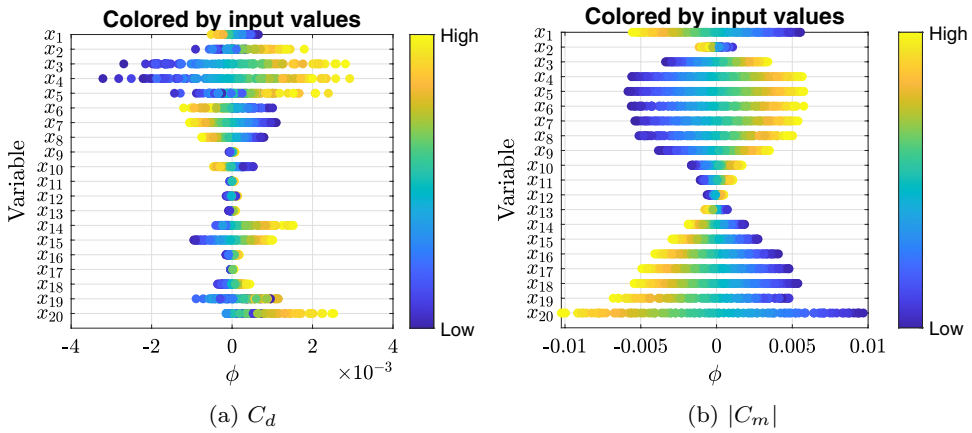
The relative impact of interaction on drag is notable for x_{14}, in which the dynamics of the change depending on the value of x_5. Further, x_{14} also changes the drag nonlinearly, despite its overall impact on drag is relatively small. The SHAP dependence plot of C_d according to x_{20} indicates that the nonlinearity of C_d is primarily due to x_{20} (i.e., the point on the lower surface and closest to the trailing edge). Upon closer inspection, it can be seen that x_{20} controls the change in the curved aft section at the trailing edge, which is also one defining feature of a supercritical airfoil. It then makes sense why x_{20} affects the drag nonlinearly. Although the impact of changing x_{20} is not as prominent as x_4, its impact on C_d is still notable. The impact of the Hicks–Henne variables on $|C_m|$ is strongly linear, as can be seen from the



(a) 1D summary plot (b) Activity scores

(c) Total Sobol indices (d) Averaged SHAP

Fig. 18 One-dimensional summary plot and global sensitivity indices for the viscous transonic airfoil case, $|C_m|$ case



(a) C_d (b) $|C_m|$

Fig. 19 SHAP summary plots for C_d and $|C_m|$, the viscous transonic airfoil case

SHAP dependence plot. Further, the effect of interactions on $|C_m|$ is almost non-existent (the disordered color gradation also indicates no interaction).

The bi-objective SHAP dependence plots, simultaneously show the impact of changing a specific input variable on the two objectives (see Fig. 22). Figure 22a, shows that the change in x_4, in general, simultaneously improves or deteriorates C_d and $|C_m|$. The high value of Pearson and Spearman correlation coefficient indicates that the two-way impact is close to linear and monotonous. On the other hand, we observe that x_{14} is the source of the trade-off between the two objectives since, e.g., increasing x_{14} would lower the $|C_m|$ but with a penalty on drag increment. However, one should also pay attention to the relative magnitude of the SHAP. In this sense, the impact of x_{14} on $|C_m|$ is larger than that on C_d. The scattered values are primarily due to the effect of interactions on C_d. Finally, one can see that the interplay between objectives

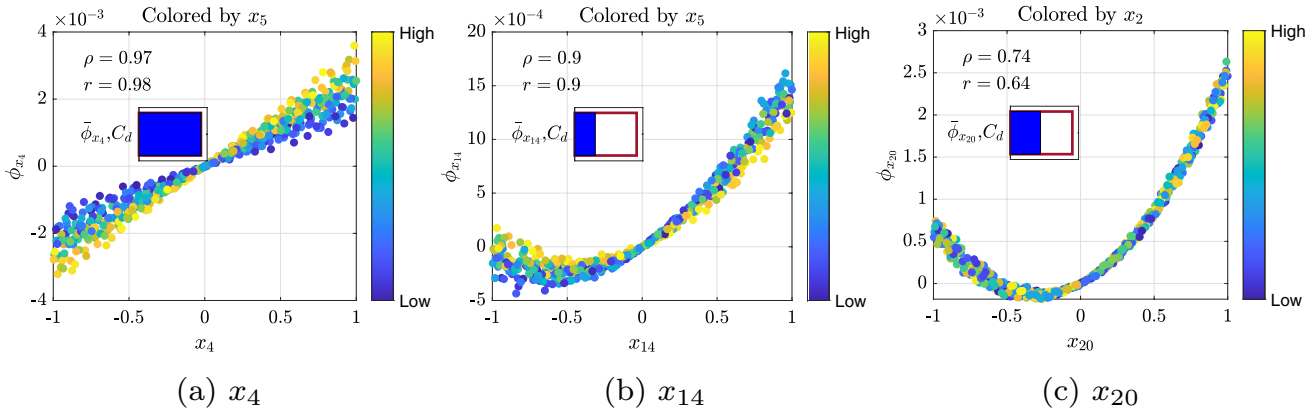


Fig. 20 SHAP dependence plots of select variables for the RAE 2822 problem, C_d case

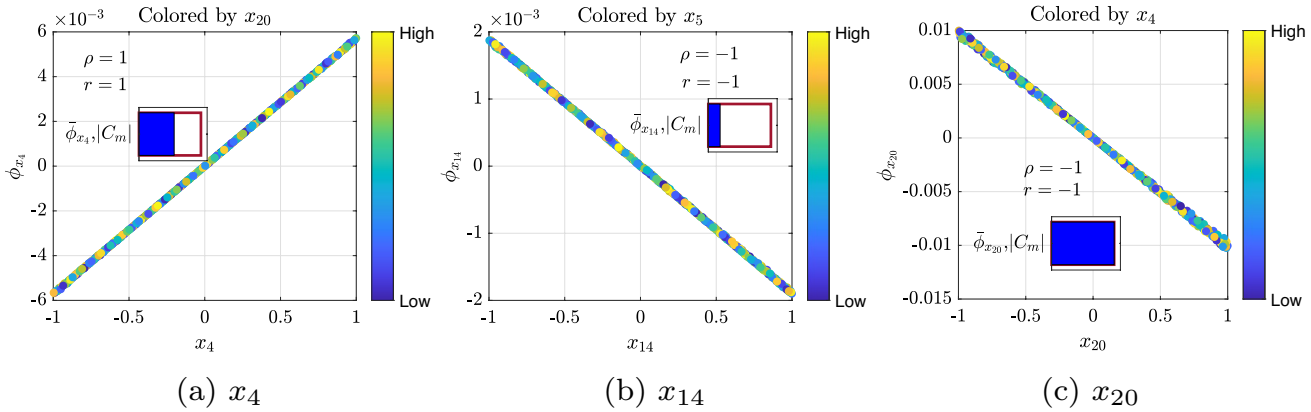


Fig. 21 SHAP dependence plots of select variables for the RAE 2822 problem, $|C_m|$ case

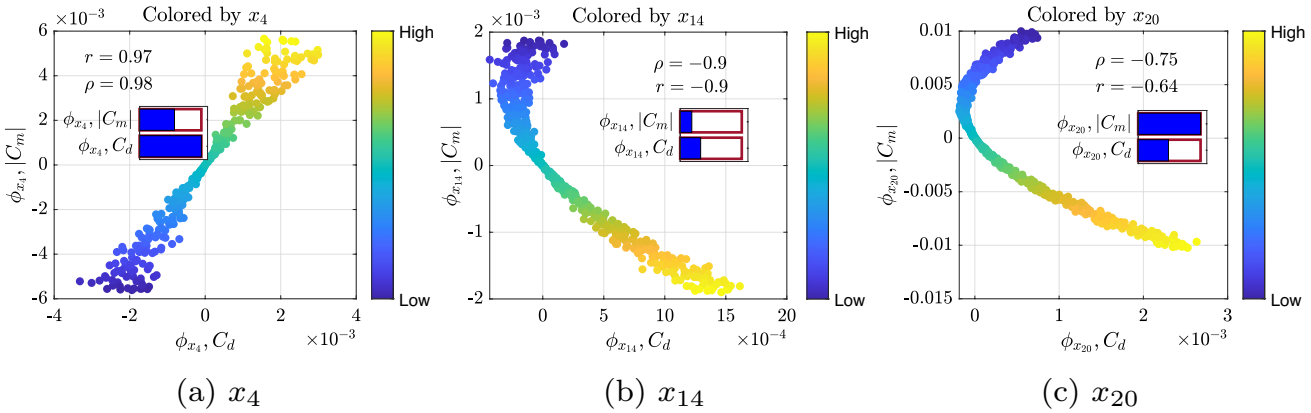


Fig. 22 Bi-objective SHAP dependence plots of select variables for the RAE 2822 problem

due to x_{20} has no linear nor monotonous relationship due to its nonlinear effect on C_d. From this viewpoint, the correlation between C_d and $|C_m|$ due to x_{20} is positive for small x_{20}, but turns to negative for large value of x_{20}.

4.2.4 SHAP correlation matrix

Finally, in Fig. 23, we can observe the SHAP Spearman correlation matrix representing the viscous transonic airfoil

design. This visualization provides a clearer understanding of the relationships between variables. Regarding the SHAP correlation matrix for $|C_m|$, it is evident that each variable exhibits a strong correlation with the pitching moment, with correlations either being perfect or close to perfect positive or negative (the values are rounded to the third decimal). In contrast, the C_d trend is more intricate, particularly for the upper surface, where the correlation alternates between positive and negative values. Conversely, the trend for the lower surface is less complex, indicating that an increase in all corresponding variables almost consistently leads to higher drag. Additionally, the bi-objective SHAP correlation (see the rightmost barplot) highlights the more intricate trend among the upper surface variables, while the trend in the lower surface is much simpler. It is worth noting that the correlation analysis should be performed while also interpreting the magnitude of the averaged SHAP values, which is also shown in the SHAP correlation matrix.

4.3 Discussions on practical aspects of SHAP

Concluding the results of numerical experiments, we can now assess the practical advantages and limitations of SHAP. The findings highlight several practical benefits of SHAP in terms of uncovering additional information that is challenging or impossible to deduce from Sobol indices and active subspaces. Sobol indices, while effective for global sensitivity analysis, fall short when the objective is to gain deeper insights into the interdependence of output and input variables. ASM offers a means to achieve this by revealing aspects such as whether the function exhibits multiple optima and potential nonlinearity. However, in the case

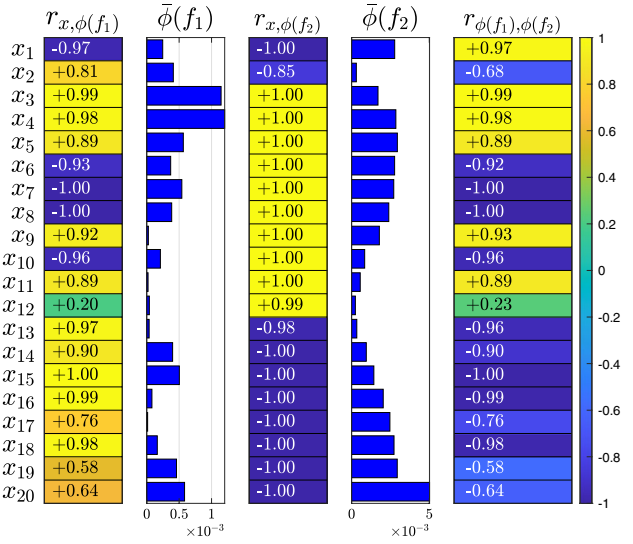


Fig. 23 SHAP correlation matrix for the viscous transonic airfoil problem (f_1 and f_2 denote C_d and $|C_m|$, respectively)

of a nonlinear function, ASM is unable to pinpoint which input variables contribute to the nonlinearity. In contrast, SHAP's advantage lies in its capability to identify variables that influence the function linearly and those that contribute to its nonlinearity. SHAP is capable of breaking down the intricacy of the function into the individual effect of the input variable. However, in contrast to ASM, discerning the unimodal or multimodal nature of the objective function via SHAP analysis is not straightforward, primarily because SHAP's decomposition nature poses challenges in identifying such characteristics.

Regarding interaction, Sobol indices are very useful in deciphering the magnitude of interaction through the second-order Sobol indices, although it is not capable of visualizing the interaction. A drawback of the current form of the ASM lies in its limited capability to explore interactions between variables in depth. While strong second or third eigenvectors from ASM might hint at interactions, the method lacks the tools to visually represent how input variables interact and influence the output. The absence of means to effectively visualize the intricate relationships between input variables and their impact on the output stands as a notable limitation within ASM. In contrast, SHAP offers a mechanism to visually represent interactions between variables using the single-output SHAP dependence plot. In cases where interaction is present, the plot exhibits scattered dots, offering valuable insights into the nature of interactions among input variables and how they collectively influence the output.

Specifically in the context of multi-objective design, SHAP offers an intuitive approach, facilitated by the bi-objective SHAP dependence plot and correlation values depicting the relationship between the SHAP scores of two objectives. The localized and decomposition nature of SHAP paved the way to investigate the contribution of input variables on two objectives simultaneously. This aspect is not attainable through Sobol indices. Furthermore, utilizing information from active subspaces of the two objectives to deduce crucial insights in multi-objective design is not straightforward. This is particularly true when trying to discern which variable contributes to the simultaneous enhancement or trade-offs between objectives, especially in cases where the active subspace demonstrates strong second or third eigenvectors.

A significant drawback of SHAP lies in the computational expense linked to the KernelSHAP algorithm. When used for visualization, a limited number of samples may be adequate to uncover the general trend. However, computing the averaged SHAP necessitates a substantial number of samples to accurately estimate their values. Nonetheless, this heightened computational cost is justified given that the execution of a single simulation could require minutes or even hours. Despite this, it is crucial to note that the expense

of conducting KernelSHAP becomes more substantial as the input dimensionality increases, and this factor should be considered in practice.

The insights derived from SHAP analysis prove highly valuable when the number of input variables is sufficiently small such that it leads to meaningful interpretation. In cases involving a substantial number of input variables, particularly in applications like optimization, ASM can be more advantageous as it offers methods for dimensionality reduction (Lukaczyk et al. 2014; Li et al. 2019). In contrast, SHAP does not directly offer the means for more efficient optimization besides identifying the importance of input variables on the objectives, which is a task that can also be accomplished by ASM. However, SHAP analysis is meaningful when the goal is to delve deeper into the complexity of the problem and the dependency of the objective on input variables.

After comparing the three techniques, we conclude that the insight obtained from the three techniques is complementary to each other. Therefore, for practical complex multi-criteria engineering problems, the three techniques can be used together to reveal information from multiple viewpoints, which eventually enrich the analysis.

5 Conclusions and future works

The present study explores the effectiveness of SHAP to help interpret a surrogate model for multi-objective design exploration. The goal is to evaluate how SHAP can uncover valuable insights from the design optimization space, which will eventually be useful for practitioners and designers. The SHAP is coupled with either GPR or PC-Kriging although other models can also be used since SHAP is a model-agnostic method. The primary objective is to utilize SHAP in multi-criteria design optimization, which is also directly applicable to single-criteria problems.

The use of SHAP in conjunction with a surrogate model allows for the identification of various design insights based on the input–output relationship. SHAP enables visualization of the nonlinearity of this relationship, as well as the interaction between variables and the strength of input variables' impact on the output. This paper discusses several means to explore such a relationship, including the newly presented SHAP correlation values and the accompanying SHAP correlation matrix. This research demonstrates that SHAP is valuable in quantifying and visualizing how an input variable affects multiple outputs simultaneously. Such knowledge is particularly useful in situations where trade-off analysis is necessary, especially in multi-criteria design.

Our study showcases how SHAP can be beneficial in solving two engineering problems: a nine-variable inviscid airfoil design and a 20-variable viscous transonic airfoil design. We also compare the effectiveness of SHAP with other techniques, namely, Sobol indices and ASM, to highlight the advantages of SHAP. Additionally, Sobol index cannot illustrate the trade-off between diverse design criteria as it is grounded in variance decomposition. On the other hand, although ASM is advantageous, it is rather complex to comprehend how an individual variable influences the output, such as which variable contributes to the nonlinear relationship. Furthermore, analyzing the trade-off between design criteria is not straightforward with ASM, particularly concerning the impact of a single input variable. SHAP analysis takes steps further, thanks to its capability to break down the individual averaged impact of each input variable. The use of SHAP also eases the analysis of the objectives trade-off study to the level of a single input variable.

In practice, SHAP can be used in complementary with ASM and Sobol indices for design exploration. This is because each method has its own advantages. ASM is primarily useful to detect the existence of one- or two-dimensional active directions, which can be used to preliminary detect unimodality and nonlinearity. The active direction information itself is not directly straightforward to infer from SHAP. On the other hand, Sobol indices are particularly useful to quantify the interactions, while it is not trivial to do so using SHAP.

For future works, one interesting research avenue is to exploit information from SHAP to perform design optimization. Also, note that the SHAP estimated from the model does not provide uncertainty, which is why we used bootstrapping. Swift uncertainty estimation of SHAP emerges as a crucial topic for future exploration, especially in the context of probabilistic models like GPR. Furthermore, it would be interesting to investigate the capability of SHAP for trade-off analysis in problems involving constraints and more than two objectives.

Appendix 1: Active subspace method

ASM (Constantine 2015) works by first calculating the averaged outer product of the output gradient $\nabla f(\boldsymbol{x})$ from a set of samples, written as

$$\frac{1}{n_r}\sum_{i=1}^{n_r}\nabla f\left(\boldsymbol{x}^{(i)}\right)\nabla f\left(\boldsymbol{x}^{(i)}\right)^{\mathrm{T}}. \tag{21}$$

Subsequently, the eigendecomposition of $\mathbf{C}$ is performed:

$$\mathbf{C} = \mathbf{W}\boldsymbol{\Lambda}\mathbf{W}. \tag{22}$$

where

$$\mathbf{W} = [\mathbf{w}_1, \ldots, \mathbf{w}_m], \quad \boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_m). \tag{23}$$

The eigenvectors $\mathbf{w}$ are arranged in a descending order based on their corresponding eigenvalues. This means that the

eigenvector with the highest eigenvalue corresponds to the most active direction.

A global sensitivity metric called activity scores, which is compared with the SHAP averaged value in this paper, can also be used (Constantine and Diaz 2017), reads as

$$\alpha_j = \sum_{i=1}^{m} \lambda_i w_{j,i}^2, j = 1, \ldots, m. \tag{24}$$

In the context of surrogate modeling, the output gradient is calculated from the surrogate model, either from finite differencing or automatic differentiation. The inputs should be normalized first so that they have the same scale before applying ASM.

Appendix 2: Sobol indices

The main principle of Sobol indices is the variance-based decomposition, which decomposes the total variance of $f(\boldsymbol{x})$ into the partial variances of the single variable and their interactions (Sobol 2001).

Define the set $[1:m]$ as $1, \ldots, m$, and let u be a subset of this set. The complement of u can be defined as $-u = [1:m] \backslash u$, and the cardinality of u is denoted by $|u|$. For an index set u, we define Ω_u as the subset of $\boldsymbol{\Omega}$ such that $\Omega_u = \prod_{i \in u} \Omega_i$. The decomposition of a function into its constituent parts can involve main effects and interactions between variables, which can be expressed as

$$y = f(\boldsymbol{x}) = \sum_{u \subseteq [1:m]} f_u(\boldsymbol{x}_u). \tag{25}$$

All the summands in Eq. (25) are orthogonal to each other, and the first term corresponds to the mean of $f(\boldsymbol{x})$.

The total variance of $f(\boldsymbol{x})$ can be expressed as the sum of partial variances V_u, reads as

$$\mathbb{V}[f(\boldsymbol{x})] = \int_{\Omega} \left(\sum_{\emptyset \neq u \subseteq [1:m]} f_u(\boldsymbol{x}_u) \right)^2 \rho(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} = \sum_{\emptyset \neq u \subseteq [1:m]} V_u = V \tag{26}$$

where $\rho(\boldsymbol{x})$ is the joint probability density function of the input, $V_u = \mathbb{V}[f_u(\boldsymbol{x}_u)] = \sigma_u^2$ is the partial variance for a subset u. Following the variance decomposition, the Sobol indices for a non-empty subset u is calculated by dividing V_u with the total variance V, written as

$$S_u = \frac{V_u}{V}, \tag{27}$$

where $\sum_{\emptyset \neq u \subseteq [1:m]} S_u = 1$. The total Sobol indices are defined as the sum of the contribution of a single variable, including the main effect and all interactions, reads as

$$S_{T_j} = \sum_{u \in \mathcal{K}_j} S_u. \tag{28}$$

where $\mathcal{K}_j = \{(u_1, \ldots, u_{|u|}) : \exists k, 1 \leq k \leq |u|, u_k = j\}$ and $\mathcal{K}_j \subseteq [1:m]$.

Appendix 3: Correlation coefficients

The Pearson correlation coefficient measures the linear correlation between two inputs. Let us consider two variables Y_1 and Y_2, the sample version of the Pearson correlation coefficient (ρ) is calculated as

$$\rho_{y_1,y_2} = \frac{\sum_{i=1}^{n_r} \left(y_1^{(i)} - \bar{y}_1\right)\left(y_2^{(i)} - \bar{y}_2\right)}{\sqrt{\sum_{i=1}^{n_r} \left(y_1^{(i)} - \bar{y}_1\right)^2 \sum_{i=1}^{n_r} \left(y_2^{(i)} - \bar{y}_2\right)^2}}, \tag{29}$$

where $\bar{y}_1$ and $\bar{y}_2$ denote the sample mean of Y_1 and Y_2, respectively, and n_r is the number of samples. In contrast to the Pearson correlation coefficient, the Spearman correlation coefficient (r) is capable of measuring the monotonous relationship between two inputs. The observations are first assigned a rank, that is, $R(y_1^{(i)})$ and $R(y_2^{(i)})$. The Spearman correlation can then be computed by the well-known formula:

$$r_{y_1,y_2} = 1 - \frac{6 \sum_{i=1}^{n_r} d_i^2}{n_r(n_r^2 - 1)} \tag{30}$$

where $d_i = R(y_1^{(i)}) - R(y_2^{(i)})$ is the difference between the two ranks of each sample point.

Acknowledgements Pramudita Satria Palar, Lavi Rizki Zuhal, and Yohanes Bimo Dwianto acknowledge the financial support provided by Institut Teknologi Bandung under the Riset Unggulan ITB for this research. Part of the work was also carried out under the Collaborative Research Project 2023 of the Institute of Fluid Science, Tohoku University.

Author contributions PSP developed the algorithms and codes and wrote the main manuscript text. LRZ secured the research funding and worked with PSP in the algorithmic development. YBD prepared and ran the aerodynamic test cases. JM worked with PSP in algorithmic development. SO secured the research funding and directed the research. KS worked with PSP in algorithmic development. All authors reviewed the manuscript.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflicts of interest.

Replication of results The SHAP package for multi-objective design exploration is written in MATLAB and available online in https://github.com/optimuspram/SHAPMODE. All surrogate models were constructed using the UQLab package (Marelli and Sudret 2014).

References

Apley DW, Zhu J (2020) Visualizing the effects of predictor variables in black box supervised learning models. J R Stat Soc Ser B (Stat Methodol) 82(4):1059–1086

Azodi CB, Tang J, Shiu SH (2020) Opening the black box: interpretable machine learning for geneticists. Trends Genet 36(6):442–455

Bandaru S, Ng AH, Deb K (2017) Data mining methods for knowledge discovery in multi-objective optimization: part a—survey. Expert Syst Appl 70:139–159

Bartoli N, Lefebvre T, Lafage R, Saves P, Diouane Y, Morlier J, Bussemaker J, Donelli G, de Mello JMG, Mandorino M, Della Vecchia P (2023) Multi-objective Bayesian optimization with mixed-categorical design variables for expensive-to-evaluate aeronautical applications. AeroBest 1:436

Blatman G, Sudret B (2011) Adaptive sparse polynomial chaos expansion based on least angle regression. J Comput Phys 230(6):2345–2367

Brahmachary S, Fujio C, Ogawa H (2020) Multi-point design optimization of a high-performance intake for scramjet-powered ascent flight. Aerosp Sci Technol 107:106362

Bukhsh ZA, Saeed A, Stipanovic I, Doree AG (2019) Predictive maintenance using tree-based classification techniques: a case of railway switches. Transport Res C Emerg Technol 101:35–54

Constantine PG (2015) Active subspaces: emerging ideas for dimension reduction in parameter studies. SIAM. https://epubs.siam.org/doi/book/10.1137/1.9781611973860

Constantine PG, Diaz P (2017) Global sensitivity metrics from active subspaces. Reliab Eng Syst Saf 162:1–13

Doshi-Velez F, Kim B (2017) Towards a rigorous science of interpretable machine learning. arXiv Preprint. arXiv:1702.08608

Dubreuil S, Berveiller M, Petitjean F, Salaün M (2014) Construction of bootstrap confidence intervals on sensitivity indices computed by polynomial chaos expansion. Reliab Eng Syst Saf 121:263–275

Economon TD, Palacios F, Copeland SR, Lukaczyk TW, Alonso JJ (2016) SU2: an open-source suite for multiphysics simulation and design. AIAA J 54(3):828–846

Goldstein A, Kapelner A, Bleich J, Pitkin E (2015) Peeking inside the black box: visualizing statistical learning with plots of individual conditional expectation. J Comput Graph Stat 24(1):44–65

Grapin R, Diouane Y, Morlier J, Bartoli N, Lefebvre T, Saves P, Bussemaker JH (2022) Regularized infill criteria for multi-objective Bayesian optimization with application to aircraft design. In: AIAA AVIATION 2022 Forum. p 4053

Greenwell BM, Boehmke BC, McCarthy AJ (2018) A simple and effective model-based variable importance measure. arXiv Preprint. arXiv:1805.04755

Hansen N, Ostermeier A (2001) Completely derandomized self-adaptation in evolution strategies. Evol Comput 9(2):159–195

He Z, Yen GG (2017) Comparison of visualization approaches in many-objective optimization. In: 2017 IEEE congress on evolutionary computation (CEC). IEEE, pp 357–363

Hicks RM, Henne PA (1978) Wing design by numerical optimization. J Aircr 15(7):407–412

Ho TK (1998) The random subspace method for constructing decision forests. IEEE Trans Pattern Anal Mach Intell 20(8):832–844

Kersaudy P, Sudret B, Varsier N, Picon O, Wiart J (2015) A new surrogate modeling technique combining kriging and polynomial chaos expansions-application to uncertainty analysis in computational dosimetry. J Comput Phys 286:103–117

Kohonen T (1990) The self-organizing map. Proc IEEE 78(9):1464–1480

LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436–444

Li J, Cai J, Qu K (2019) Surrogate-based aerodynamic shape optimization with the active subspace method. Struct Multidisc Optim 59:403–419

Liu DC, Nocedal J (1989) On the limited memory BFGS method for large scale optimization. Math Program 45(1–3):503–528

Lukaczyk TW, Constantine P, Palacios F, Alonso JJ (2014) Active subspaces for shape optimization. In: 10th AIAA multidisciplinary design optimization conference. p 1171

Lundberg SM, Lee SI (2017) A unified approach to interpreting model predictions. In: Proceedings of the 31st international conference on neural information processing systems. pp 4768–4777

Mangalathu S, Hwang SH, Jeon JS (2020) Failure mode and effects analysis of RC members based on machine-learning-based SHapley Additive exPlanations (SHAP) approach. Eng Struct 219:110927

Marelli S, Sudret B (2014) UQLab: a framework for uncertainty quantification in MATLAB. In: Vulnerability, uncertainty, and risk: quantification, mitigation, and management. pp 2554–2563

McKay MD, Beckman RJ, Conover WJ (2000) A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics 42(1):55–61

Meneghini IR, Koochaksaraei RH, Guimaraes FG, Gaspar-Cunha A (2018) Information to the eye of the beholder: data visualization for many-objective optimization. In: 2018 IEEE congress on evolutionary computation (CEC). IEEE, pp 1–8

Obayashi S, Jeong S, Chiba K (2005) Multi-objective design exploration for aerodynamic configurations. In: 35th AIAA fluid dynamics conference and exhibit. p 4666

Obayashi S, Jeong S, Chiba K, Morino H (2007) Multi-objective design exploration and its application to regional-jet wing design. Trans Jpn Soc Aeronaut Space Sci 50(167):1–8

Obayashi S, Jeong SK, Shimoyama K, Chiba K, Morino H (2010) Multi-objective design exploration and its applications. Int J Aeronaut Space Sci 11(4):247–265

Palar PS, Yang K, Shimoyama K, Emmerich M, Bäck T (2018) Multi-objective aerodynamic design with user preference using truncated expected hypervolume improvement. In: Proceedings of the genetic and evolutionary computation conference. pp 1333–1340

Palar PS, Zuhal LR, Shimoyama K, Dwianto YB, Morlier J (2023) Shapley additive explanations for knowledge discovery via surrogate models. In: AIAA SCITECH 2023 Forum. p 0332

Park JH, Jo HS, Lee SH, Oh SW, Na MG (2022) A reliable intelligent diagnostic assistant for nuclear power plants using explainable artificial intelligence of GRU-AE, LightGBM and SHAP. Nucl Eng Technol 54(4):1271–1287

Pawlak Z (1998) Rough set theory and its applications to data analysis. Cybern Syst 29(7):661–688

Pimentel AD (2016) Exploring exploration: a tutorial introduction to embedded systems design space exploration. IEEE Des Test 34(1):77–90

Rasmussen CE (2003) Gaussian processes in machine learning. In: Summer school on machine learning. Springer, pp 63–71

Sacks J, Schiller SB, Welch WJ (1989) Designs for computer experiments. Technometrics 31(1):41–47

Saves P, Diouane Y, Bartoli N, Lefebvre T, Morlier J (2023) A mixed-categorical correlation kernel for Gaussian process. Neurocomputing 550:126472

Shapley LS (2016) 17. A value for n-person games. Princeton University Press, Princeton

Smola AJ, Schölkopf B (2004) A tutorial on support vector regression. Stat Comput 14(3):199–222

Sobieczky H (1999) Parametric airfoils and wings. In: Recent development of aerodynamic design methodologies: inverse design and optimization. Springer, pp 71–87

Sobol IM (2001) Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. Math Comput Simul 55(1–3):271–280

Stadler W (1988) Multicriteria optimization in engineering and in the sciences, vol 37. Springer Science & Business Media, New York

Stiglic G, Kocbek P, Fijacko N, Zitnik M, Verbert K, Cilar L (2020) Interpretability of machine learning-based prediction models in healthcare. Wiley Interdiscip Rev Data Min Knowl Discov 10(5):e1379

Sumimoto T, Chiba K, Kanazaki M, Fujikawa T, Yonemoto K, Hamada N (2019) Evolutionary multidisciplinary design optimization of blended-wing-body-type flyback booster. In: AIAA Scitech 2019 Forum. p 0703

Takanashi S, Nishimura S, Eto K, Hatanaka K (2023) Shapley additive explanations for knowledge discovery in aerodynamic shape optimization. In: AIAA SCITECH 2023 Forum. p 0904

Vollert S, Atzmueller M, Theissler A (2021) Interpretable machine learning: a brief survey from the predictive maintenance perspective. In: 2021 26th IEEE international conference on emerging technologies and factory automation (ETFA). IEEE, pp 01–08

Xiu D, Karniadakis GE (2003) Modeling uncertainty in flow simulations via generalized polynomial chaos. J Comput Phys 187(1):137–167