RECENT ADVANCES IN FUSING HIGH-DIMENSIONAL AERODYNAMIC WIND TUNNEL AND CFD DATA

A. Bertram*, M. Held†

* Deutsches Zentrum für Luft- und Raumfahrt (DLR), Institut für Aerodynamik und Strömungstechnik, Abteilung $C^2A^2S^2E$, Lilienthalplatz 7, 38108 Braunschweig, Deutschland

† Airbus Operations GmbH, Aircraft Aerodynamics, Airbus-Allee 1, 28199 Bremen, Deutschland

Abstract

During the development of an aircraft, a multitude of aerodynamic data is required for different flight conditions throughout the flight envelope. Nowadays, a large portion of this data is acquired through Computational Fluid Dynamics (CFD) simulations. However, due to modeling and convergence issues especially for extreme flight conditions, numerical data cannot be reliably generated throughout the entire flight envelope. Numerical data is therefore complemented by data from wind tunnel experiments and flight testing. Because of errors mainly introduced by the physical modeling and the discretization of the problem on the one hand and experimental limitations on the other hand, the data from these different sources will always show some discrepancies to deal with. Data fusion methods aim at combining the individual strengths of data from different sources in order to provide a consistent data set throughout the entire parameter domain.
For the past four years, DLR and Airbus have been working together in the field of aerodynamic data fusion and have jointly funded a dedicated research position. This work provides an overview on the advances made within this collaboration.

Keywords

Data Fusion; Data-driven Modelling; Gappy POD

NOMENCLATURE

Symbols

α	Angle of attack
Cov	Covariance
E	Expect value
$\mathcal{GP}$	Gaussian Process
M	Mach number
$\mathcal{N}$	Gaussian distribution
Re	Reynolds number
Var	Variance
x/c	Chord normalized airfoil length

Acronyms

CFD	Computational Fluid Dynamics
GPR	Gaussian Process Regression
RMSE	Mean squared error
POD	Proper Orthogonal Decomposition
RMSE	Root mean squared error
SMARTy	DLR's Surrogate Modeling for AeRo-data Toolbox in Python

1. MOTIVATION AND INTRODUCTION

Climate change and digitalization have resulted in a transformation of energy and design concepts in the aerospace industry. An acceleration of the development process of new aircraft configurations is required to meet the challenges of the future. For the development of new aircraft, large amounts of high-dimensional aerodynamic data is required for different flight conditions throughout the flight envelope. A large portion of this data is nowadays acquired through Computational Fluid Dynamics (CFD) simulations. Due to convergence issues and modeling inaccuracies, computer simulations do not produce reliable, high-fidelity data for every flow condition. Especially for highly turbulent flows and extreme flight conditions, large discrepancies to experimental data can be observed. Data fusion methods combine data from different sources according to their individual strengths. They can therefore serve as a key technology for future aircraft design.
In the last decades, advances have been made in developing robust data fusion techniques which are able to deal with industrial-scale problems. A strategy for the fusion of scalar-valued data which have a clear hierarchy in terms of accuracy is the use of variable-fidelity surrogate models. Popular choices of these methods are bridge functions, [1, 2], Cok-

doi: 10.25967/570233

riging, [3–5] and Hierarchical Kriging, [6]. Recently, a non-hierachical data fusion approach for scalar-valued quantities using a weighted combination of Gaussian Process Regression (GPR) models was introduced in [7].
Depending on the particular setting, different methods exist for vector-valued quantities of interest: If data from different fidelity levels is given on a common spatial grid, variable-fidelity methods for vector-valued quantities can be used, [8, 9], which reduce the problem to a variable-fidelity surrogate modeling of the scalar-valued Proper Orthogonal Decomposition (POD) basis coefficients. A new Bayesian approach was proposed in [10], where the fused solution is defined as weighted sum of experimental and numerical data to the same input parameters. Both approaches are based on the assumption that both data sources provide data on the same spatial grid. In a setting where experimental data from few, sparsely distributed sensors on the one hand and high-fidelity CFD simulations on the other hand is given, these methods are no longer applicable. The coarse spatial grid of experimental sensor information can, in general, not be accurately interpolated to a high-dimensional spatial grid as given for CFD simulations. Gappy POD methods are designed to address this issue. The idea is that within the POD subspace derived from a CFD data set, a solution can be found which minimizes the differences to reference data at a few discrete locations in a least-square sense. The method was first developed for the reconstruction of human face images from incomplete data sets [11] and was later extended to fluid dynamic applications in [12], where it was applied for the reconstruction of missing data in CFD snapshots for steady aerodynamic flow around an airfoil. In [13], the method was used for the reconstruction of unsteady flow data. To avoid overfitting, especially when dealing with real experimental data, different regularization methods for Gappy POD have been compared and applied to an industrial test case. The approach was further extended such that the fused surface data fulfills constraints on its aerodynamic integral coefficients in [14]. One of the limitations of the Gappy POD approach stems from the assumption of linearity within the POD model and the consecutive least-squares problem. Furthermore, the acceptance of such data-driven approaches is often limited by the fact that it is hard to estimate the uncertainty in the predicted data. An alternative approach to the Gappy POD methods was recently proposed in [15]. The idea is to directly map the sensor measurements to the full field solution via a shallow neural network. In a case study with CFD data, the shallow neural network approach was able to outperform ordinary and regularized Gappy POD. Up to our knowledge, there is no study available in literature in which the method is tested for real data fusion tasks using experimental and numerical data. A problem might be that for data fusion tasks, experimental training data, i.e. pairs of sensor data and corresponding full field solution, are in general not available.
From 2018 until 2022, Airbus and DLR have been working closely together in the field of aerodynamic data fusion and jointly funded a research position dedicated to the topic. The focus of this collaboration was the development and investigation of methods for the robust prediction of aerodynamic data of full aircraft configurations across the flight envelope. In the scope of the project, the Gappy POD methods have been further extended and new data fusion methods have been investigated. A weighted Gappy POD was introduced which enables to incorporate expert knowledge on the measurements. By taking a regression perspective on the least-squares problem, a natural generalization of Gappy POD was found from which the so-called Bayesian Gappy POD was derived. The method overcomes the problem of linearity of ordinary Gappy POD approaches and was demonstrated to yield significantly improved results on an industrial research test case, [16]. Furthermore, the data fusion result is given in terms of a probability distribution which can be assessed to obtain information on the associated modeling uncertainty. As an alternative to Gappy POD, the shallow neural network approach proposed in [15] was investigated for real data fusion tasks.
This report provides an overview of the most important results of the project. In Section 2, the general problem setup is described and notation is introduced. Section 3 gives a short introduction to the state-of-the-art of data fusion with Gappy POD. Its extensions, Weighted Gappy POD and Bayesian Gappy POD, are described in Section 5 and exemplary results are displayed. A study investigating the shallow neural network approach for data fusion tasks is given in Section 6. The report concludes with closing remarks and an outlook in Section 7.

2. PROBLEM SETUP

Assume that there is a functional dependency between the input parameters $\xi \in \mathcal{D} \subset \mathbb{R}^d$ and the resulting high-dimensional quantity of interest, $y\colon \mathcal{D} \to \mathbb{R}^N$. This functional dependency is not explicitly given but can be assessed for given input parameters $\xi^* \in \mathbb{R}^d$ using two different data sources: A primary source, which provides accurate but incomplete information, i.e. the quantity of interest $y(\xi^*)$ can only be observed at $s \ll N$ pairwise distinct components $\{j_1, \dots, j_s\} \subset \{1, \dots, N\}$. In addition, a secondary data source $\tilde{y}\colon \mathcal{D} \to \mathbb{R}^N$ is available which describes the same quantity of interest with a lower accuracy. Based on the incomplete information given by the primary data source, we are aiming to reconstruct the full quantity of interest $y(\xi^*)$ with the aid of sampled data from the secondary data source. This setup is very general and may arise in different fields of application. In the special application we consider in this work, the goal is to reconstruct the pressure coefficient distribution at the surface of an

aircraft on a highly resolved computational mesh from sparse experimental pressure sensor data using information given by a set of pre-computed CFD solutions. In the subsequent sections, different methods are introduced addressing this problem. For simplicity, we will in the remainder of this work use the common terms related to our target application, i.e. the primary data is referred to as experimental or sensor data and the secondary data is referred to as simulation data. Nevertheless, all methods and approaches are not restricted to this special setting and may transfer to other applications as well.

3. DATA FUSION WITH GAPPY POD – STATE OF THE ART

In the following, an overview of the current state-of-the-art in data fusion, the Gappy POD methodology, is provided based on the initial work [17] and it's extensions [18] and [14]. Large parts of this overview have been published in our work [16].

4. ORDINARY GAPPY POD

Let $\{\tilde{y}^1 \coloneqq \tilde{y}(\xi^1), \ldots, \tilde{y}^n \coloneqq \tilde{y}(\xi^n)\} \subset \mathbb{R}^N$ be a set of sampled data obtained via CFD simulations for different input parameters $\mathcal{P} = \{\xi^1, \ldots, \xi^n\} \subset \mathcal{D}$ and let the corresponding outputs be stored in the *snapshot matrix* $Y \in \mathbb{R}^{N \times n}$,

$$Y \coloneqq [\tilde{y}^1, \ldots, \tilde{y}^n]. \tag{1}$$

For simplicity of the notation and without loss of generality assume that the snapshots are centered with respect to their mean, that is

$$\sum_{i=1}^{n} \tilde{y}^i = 0. \tag{2}$$

Performing a singular value decomposition (SVD) [19, Sec. 2.4] of the snapshot matrix yields

$$Y = U \Sigma V^{\mathrm{T}}, \tag{3}$$

where $U = [u^1, \ldots, u^N] \in \mathbb{R}^{N \times N}$ and $V = [v^1, \ldots, v^n] \in \mathbb{R}^{n \times n}$ are orthonormal matrices, i.e. $U^{\mathrm{T}}U = UU^{\mathrm{T}} = I_N$ and $V^{\mathrm{T}}V = VV^{\mathrm{T}} = I_n$, and $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_n) \in \mathbb{R}^{N \times n}$ contains the singular values $\sigma_1 \geq \ldots, \geq \sigma_n \geq 0$ in descending order. Suppose the rank of the snapshot matrix Y is $r = \mathrm{rank}(Y)$, then only the first $r \leq n$ singular values are non-zero. The corresponding r left singular vectors, which are the first r columns of the matrix U, constitute an orthonormal basis $\{u^1, \ldots, u^r\}$ of the space spanned by the snapshots $\tilde{y}^1, \ldots, \tilde{y}^n$, the so-called *POD basis*, [19, Corollary 2.4.6].
A key idea of Gappy POD is to interpret a given vector $t \in \mathbb{R}^s$ of experimental data, where $s < N$ is the number of experimental sensors, as a vector $y \in \mathbb{R}^N$ from which only the components $y_{j_1}, \ldots, y_{j_s}$ with $j_1, \ldots, j_s \in \{1, \ldots, N\}$ are known, that is

$$t = \begin{bmatrix} t_1 \\ \vdots \\ t_s \end{bmatrix} = \begin{bmatrix} y_{j_1} \\ \vdots \\ y_{j_s} \end{bmatrix} = P^{\mathrm{T}} y \tag{4}$$

for a mask matrix $P \coloneqq [e_{j_1}, \ldots, e_{j_s}] \in \mathbb{R}^{N \times s}$. The components of the vector t are identified with the components of the vector y via a nearest neighbor search of the s sensor coordinates in the highly resolved computational grid.
Assuming that the vector y can be approximated in the POD subspace, POD basis coefficients $\hat{a} = (\hat{a}_1, \ldots, \hat{a}_r)^{\mathrm{T}} \in \mathbb{R}^r$ can be found such that

$$y \approx \hat{y} = \sum_{j=1}^{r} \hat{a}_j u^j = U_r \hat{a}, \tag{5}$$

where $U_r = [u^1, \ldots, u^r] \in \mathbb{R}^{N \times r}$ is the matrix of POD basis vectors, i.e. the first r columns of U. The basis coefficient vector $\hat{a} \in \mathbb{R}^r$ which yields the smallest L_2 error regarding the observed entries of the vector y is defined by the least squares problem

$$\hat{a} = \arg\min_a \left\| P^{\mathrm{T}} U_r a - t \right\|_2^2. \tag{6}$$

Usually, $X = P^{\mathrm{T}} U_r \in \mathbb{R}^{s \times r}$ has full column rank and therefore Eq. (6) has a unique solution given by

$$\hat{a} = (X^{\mathrm{T}} X)^{-1} X^{\mathrm{T}} t. \tag{7}$$

Substituting this basis coefficient vector $\hat{a}$ into Eq. (5) yields the ordinary Gappy POD approximation of the vector y.

4.1. Regularized Gappy POD

To avoid overfitting, especially when dealing with data from different sources, it is often necessary to complement the least squares problem with regularization terms on the basis coefficients $\hat{a}$, [14, 18]. Shrinkage methods give preference to smaller basis coefficients $\hat{a}$ by imposing a penalty on their value, cf. [20, Sec. 3]. In order to restrict the influence of less important POD modes, i.e. basis vectors which correspond to small singular values, using these techniques, the orthonormal basis vectors need to be scaled with their corresponding singular values

$$\tilde{U} = [\tilde{u}^1, \ldots, \tilde{u}^r] \coloneqq [\sigma_1 u^1, \ldots, \sigma_r u^r]. \tag{8}$$

A popular choice for the regularization of least squares problems is Ridge regression, sometimes also called Tikhonov regularization, [20, Sec. 3.4.1], [21]. It imposes a L_2 penalty on the basis coefficient vector a. The corresponding Gappy POD problem reads,

$$\hat{a}_{\mathrm{rr}} = \arg\min_a \left\| P^{\mathrm{T}} \tilde{U} a - t \right\|_2^2 + \lambda \|a\|_2^2, \tag{9}$$

where $\lambda \in \mathbb{R}_+$ is a parameter which controls the strength of the regularization: the larger the value of λ, the higher the amount of shrinkage towards 0. If $X = P^{\mathrm{T}}U_r$ has full column rank, the Gappy POD problem has a unique solution,

$$\hat{a}_{\mathrm{rr}} = (X^{\mathrm{T}}X + \lambda I)^{-1}X^{\mathrm{T}}t. \tag{10}$$

As in the ordinary case, the Gappy POD approximation of the vector y is obtained by evaluating the corresponding linear combination of POD basis modes,

$$\hat{y} = \sum_{j=1}^{r} \hat{a}_j u^j = \tilde{U}_r \hat{a}_{\mathrm{rr}}, \tag{11}$$

cf. Eq. (5).

5. GAPPY POD EXTENSIONS

In this project, several extensions of the ordinary Gappy POD approach have been developed. The two main extensions are described in the subsequent sections: In Section 5.1, the Weighted Gappy POD approach is introduced. A description of Bayesian Gappy POD, is given in Section 5.2 followed by exemplary results for an industrial research test case in Section 5.3.

5.1. Weighted Gappy POD

The Gappy POD method enables to easily incorporate weights for the sensor data: Let the weights for the vector $t = [t_1, \ldots, t_s]^{\mathrm{T}} \in \mathbb{R}^s$ of experimental data be given by $\omega_1, \ldots, \omega_s \in \mathbb{R}_+$. Weighting can be performed by applying the weighting matrix

$$W = \mathrm{diag}(\omega_1, \ldots, \omega_s) \in \mathbb{R}^{s\times s} \tag{12}$$

to the residual term in the ordinary Gappy POD problem Eq. (6),

$$\hat{a}_W = \arg\min_a \left\| WP^{\mathrm{T}}U_r a - Wt \right\|_2^2, \tag{13}$$

or the regularized Gappy POD problem Eq. (9),

$$\hat{a}_{\mathrm{rr,\,W}} = \arg\min_a \left\| WP^{\mathrm{T}}\tilde{U}a - Wt \right\|_2^2 + \lambda\|a\|_2^2. \tag{14}$$

Expert knowledge about the accuracy of individual sensors can be considered by giving large weighting values to measurements for which the expected accuracy is high. Conversely, measurement values with lower expected accuracy can receive a lower weighting.

5.2. Bayesian Gappy POD

By interpreting the least squares problem in Gappy POD as a linear regression task, a natural generalization of the ordinary approach can be found. The regression perspective gives rise to the definition of Bayesian Gappy POD, first introduced in our work [16]. In the following, the method is briefly described. For more details on the method, the reader is referred to the initial work.

The Gappy POD least squares problem Eq. (6) can be interpreted as a linear regression problem

$$f(x) = x^{\mathrm{T}}w, \qquad w \in \mathbb{R}^r \tag{15}$$

for the sample data set

$$\{(x_i, t_i) \mid i = 1, \ldots, s\}, \tag{16}$$

where $x_i \coloneqq (X)_i = (P^{\mathrm{T}}U_r)_i = (U_r)_{j_i} \in \mathbb{R}^r$ denotes the i-th row of the matrix $X = P^{\mathrm{T}}U_r$ or, in other words, the j_i-th row of the matrix of POD modes U_r and $t_i = (t)_i$ is the corresponding sensor response. In this setting, we aim at evaluating the regression model f for all rows of the matrix of POD modes $(U_r)_i, i = 1, \ldots, N$, to obtain the Gappy POD solution,

$$\hat{y} = f(U_r) = \left(f\left((U_r)_i\right)\right)_{i=1,\ldots,N}. \tag{17}$$

A natural generalization to this approach is to replace the simple regression model (15) by the generalized linear regression model with Gaussian noise,

$$f(x) = \phi(x)^{\mathrm{T}}w, \qquad t(x) = f(x) + \varepsilon, \tag{18}$$

with a fixed set of basis functions

$$\phi(x) = (\phi_1(x), \ldots \phi_m(x))^{\mathrm{T}}, \tag{19}$$

corresponding weight vector $w \in \mathbb{R}^m$, and additive independent, identically distributed Gaussian noise ε with zero mean and stationary variance $\sigma^2 > 0$. In the Bayesian Gappy POD extension proposed in our work [16], the Gappy POD regression problem is solved by using GPR, also known as Kriging [22, 23]. Assuming that $f(x)$ is a zero-mean Gaussian process whose covariance $\mathrm{Cov}[f(x), f(x')]$ for any two inputs $x, x' \in \mathbb{R}^r$ is given by a positive semi-definite covariance kernel $k(x, x')$,

$$f(x) \sim \mathcal{GP}(0, k(x, x')), \tag{20}$$

results in a Gaussian predictive distribution of $f(x^*) = f^*$ for an input $x^* \in \mathbb{R}^r$ with

$$\begin{aligned} \mathrm{E}[f_*] &= k(x_*)\left(K + \sigma^2 I\right)^{-1} t, \\ \mathrm{Var}[f_*] &= k(x_*, x_*) - k(x_*)^{\mathrm{T}}\left(K + \sigma^2 I\right)^{-1} k(x_*), \end{aligned} \tag{21}$$

where $K \coloneqq (k(x_i, x_j))_{i,j=1,\ldots,s} \in \mathbb{R}^{s\times s}$ is the covariance matrix and $k(x_*) \coloneqq (k(x_*, x_i))_{i=1,\ldots,s} \in \mathbb{R}^s$ is the vector of covariances of the sample points and the location x_* , [22, Sec. 2.2].
The choice of the covariance function of the Gaussian process has a large impact on the predictions, see e.g. [24, Sec. 6.4.2]. A widely used class of covariance functions is given by

$$k(x, x') = \theta_0 \cdot \exp\left(-\theta_1\|x - x'\|^2\right) + \theta_2 x^{\mathrm{T}}x', \tag{22}$$

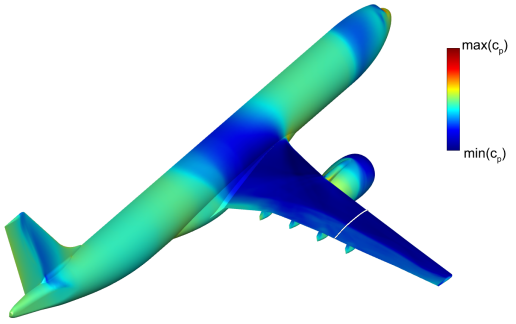
cf. Eq. (6.63) in [24, Sec. 6.4, p. 307], with hyperparameters $\theta_0, \theta_1, \theta_2$. Instead of defining the hyperparameters and the noise variance σ^2 in advance, they are usually determined from the data by maximizing the log marginal likelihood function. For a detailed discussion of covariance functions for GPRs and different methods for hyperparameter estimation, the reader is referred to Sec. 4 and Sec. 5.4 of the textbook [22].
The new Bayesian Gappy POD extension, as well as the established approach have been implemented in DLR's python-based Surrogate Modeling for AeRodata Toolbox SMARTy, [25].

5.3. Application to a Transport Aircraft Test Case

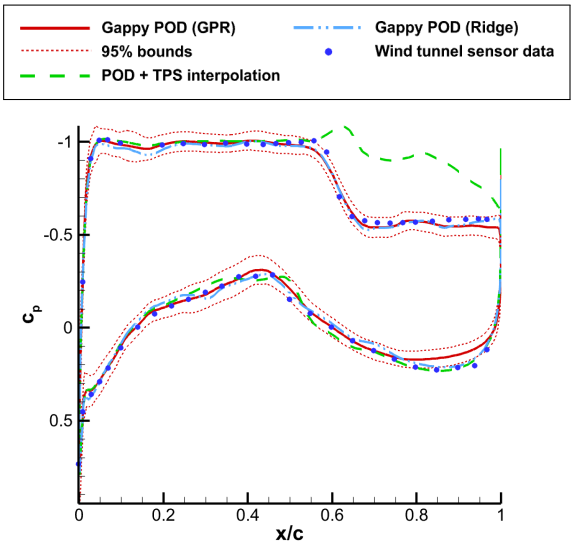
Results presented in this section are extracted from the detailed work [16]. The results shown here slightly differ from those presented in the previous work by the fact that in this paper, we applied the scaling of the POD basis vectors, as introduced for regularized Gappy POD in Eq. (8), to both Gappy POD methods which further improved the results.

5.3.1. Case study description

The test case configuration considered here is an industrial-relevant aircraft configuration known as XRF1. The XRF1 is an Airbus provided industrial standard multi-disciplinary research test case representing a typical configuration for a long range wide body aircraft. It is used by Airbus to engage with external partners on development and demonstration of relevant capabilities and technologies. The surface grid for the high-fidelity RANS-CFD simulations consists of $N = 388,918$ grid points. A total number of $n = 100$ pressure coefficient (c_p) distributions on the surface of the aircraft were computed for different Mach numbers $\mathrm{M} \in [0.5, 0.96]$ and angles of attack $\alpha \in [-11°, 14°]$ and are considered as CFD snapshots during the following investigations. The Reynolds number was fixed to $\mathrm{Re} = 25 \times 10^6$. Wind tunnel tests for the XRF1 configuration were carried out in the European Transonic Windtunnel (ETW). The surface pressure coefficient was obtained at 314 pressure taps distributed over 26 section cuts along the wing for a number of 196 different combinations of Mach number and angle of attack.
Data fusion was employed using regularized Gappy POD with Ridge regression, cf. Sec. 4.1, and the Bayesian Gappy POD approach with Gaussian Process Regression as introduced in Sec. 5.2. In the latter case, the kernel function was chosen from the class Eq. (22), where the hyperparameters $\theta_0, \theta_1, \theta_2$ were determined by maximizing the marginal likelihood function. The noise variance σ^2 is assumed to be stationary and is estimated along with the kernel hyperparameters in the maximum likelihood optimization. Note that it is also possible to provide pre-defined values instead, e.g. when measurement uncertainties are known. An article with a comparative study with measurement uncertainties from the wind tunnel experiment is currently in preparation, [26]. For comparison, a surrogate model was constructed solely based on the CFD data set by interpolating the POD basis coefficients via TPS interpolation, [27].



(a) Bayesian Gappy POD result (mean).



(b) Result of different methods at the marked section cut.

FIG 1. Pressure coefficient approximation for $\mathrm{M} = 0.92, \alpha = 7.01°$.

Exemplary results for a Mach number $\mathrm{M} = 0.92$ and an angle of attack $\alpha = 7.01°$ are shown in Fig. 1. In Fig. 1a a contour plot of the mean of the predictive distribution of the Bayesian Gappy POD result using GPR is given. Fig. 1b shows the pressure coefficient distribution using the different investigated methods at the marked section cut.
Both Gappy POD methods accurately predict the strength and position of the shock. While Bayesian Gappy POD with GPR shows a slightly smoother trend as the state-of-the-art approach, the flow behavior at the trailing edge is better captured by Gappy POD with Ridge regression. The $95\,\%$ credible intervals are small, indicating a high confidence in the Bayesian Gappy POD prediction. The CFD-based surrogate model fails to predict the strength and position of the occurring shock.

The root mean squared error,

$$\mathrm{RMSE}(\hat{y}) := \sqrt{\frac{1}{s}\|P^{\mathrm{T}}\hat{y} - t\|_2^2}, \tag{23}$$

was evaluated for all available 196 data sets from the wind tunnel experiment. In Fig. 2, the average root mean squared error for the predictions of the investigated methods is displayed.

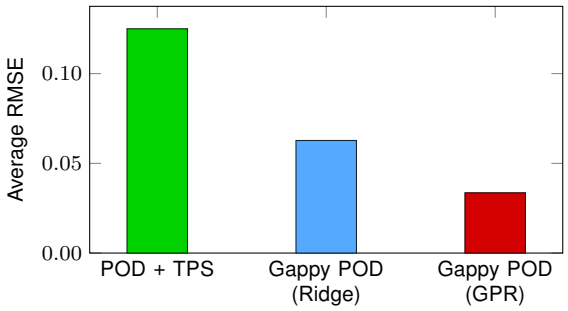


FIG 2. Average root mean squared error with respect to the 314 wind tunnel pressure probes for Gappy POD with Ridge regression, Bayesian Gappy POD with GPR and POD with TPS interpolation for 196 wind tunnel data sets in the different domains of the envelope.

As can be seen from the bar plot, the Gappy POD approaches outperform the simple POD + TPS interpolation. Averaged over all 196 wind tunnel test cases, Bayesian Gappy POD with GPR gives a root mean squared error which is about $47\,\%$ smaller than the root mean squared error of the established Gappy POD with Ridge regression.

6. SHALLOW NEURAL NETWORKS FOR FLUID FLOW RECONSTRUCTION

A drawback of the Gappy POD approaches is however that the POD basis is constructed without taking information on the measurements into account. This may lead to observability issues as discussed in [15]. To overcome this issue, the authors proposed to train a neural network approach directly on the mapping between sensor measurements and full solution. In the scope of our project, we investigated the new approach as alternative to Gappy POD methods. The main setup, as well as our studies are presented in the following.

6.1. Setup

6.1.1. Architecture

A two hidden layer, fully-connected MLP is used to describe the mapping between the sensor data $t \in \mathbb{R}^s$ and the corresponding full surface solution $\hat{y} \in \mathbb{R}^N$. The numbers of neurons of the two hidden layers, n_1 and n_2, are chosen such that it increases from layer to layer, $s \leq n_1 \leq n_2 \leq N$. The architecture of the network is illustrated in Fig. 3.

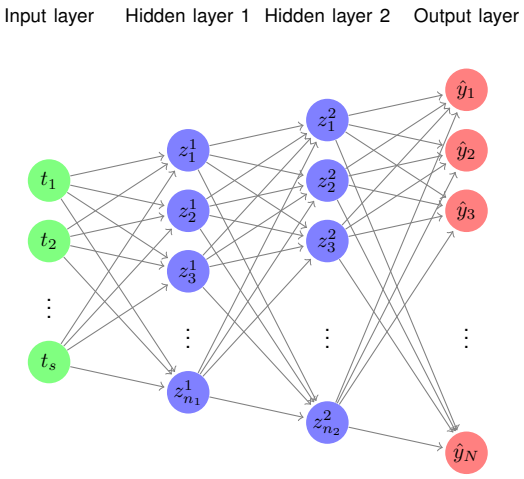


FIG 3. Architecture of the shallow neural network

The last hidden layer is connected to the output layer via a linear activation function,

$$\hat{y} = \Phi z^2 + b, \tag{24}$$

with weights $\Phi \in \mathbb{R}^{N \times n_2}$, bias $b \in \mathbb{R}^N$ and where $z^2 \in \mathbb{R}^{n_2}$ denotes the output of hidden layer 2. In this setting, the columns of the weighting matrix Φ and the bias b define an affine subspace in which the network output is contained:

$$\mathcal{V} = \{y \mid y = \Phi z + b, z \in \mathbb{R}^{n_2}\} \subset \mathbb{R}^N. \tag{25}$$

6.1.2. Training issue

In contrast to Gappy POD methods, pairs of input and output data are needed for model training, i.e. the full surface solution must be available for the sensor data used for training. Note that this is in general not the case when dealing with wind tunnel sensor data. To overcome this problem, the following training configurations are possible:

1) **Sensor and CFD data:** A set of sensor measurements is used as input data for training. As outputs, the corresponding full surface solution is computed for the same flow conditions using CFD simulation.
2) **CFD data only:** The training of the model is restricted to CFD data only. While the full solution serves as output, the information at the nearest neighbors of the sensor locations in the CFD mesh is used for input.
3) **Sensor and PSP data:** Pairs of sensor measurements and their corresponding pressure sensitive paint (PSP) result at a discrete mesh are used for training. CFD data is not considered.

Each of the three configurations has certain advantages and drawbacks. Option 1 seems to be the most intuitive way to overcome the training issue. However, by taking the CFD data as output for sensor data input, it is assumed that the CFD data is the absolute truth for the observed wind tunnel data. The neural network therefore learns to find the CFD result based

on wind tunnel measurements which is in general undesired.
Only relying on CFD data for training is a more consistent setting and is very similar to the Gappy POD methods. On the other hand, option 2 has the clear drawback that, the final model was not trained on the data which it will be later used on.
Using sensor data together with PSP data for training, option 3, has the advantage that both kinds of data are obtained from the same wind tunnel experiment. Thus, the data should show a good consistency. The final model output can be viewed as a surrogate for the PSP data based on sensor inputs. Note that this is completely different from what Gappy POD methods are aiming for. The results can therefore hardly be compared with each other.

Because of the similarities with the Gappy POD methods in terms of the setup, we decided to investigate option 2 in our project.

6.2. Application to an airfoil test case

The test case used to assess the shallow neural network for fluid flow reconstruction is the RAE2822 airfoil, [28], shown in Fig. 4. CFD and wind tunnel

FIG 4. The RAE2822 airfoil

data were provided by Airbus within the RWC.01 data base. The Airbus RWC.01 data base gathers aerodynamic experimental data acquired in 2016 using the pilot facility of the European Transonic Wind Tunnel (pETW) for a series of 2D airfoil sections. In particular, the reference RAE2822 section geometry equipped with thicker trailing edge was tested to cross-check results with legacy data and extends the range of operating conditions. For different combinations of Reynolds number, Mach number and angle of attack, the pressure coefficient was measured with 36 pressure taps located at the surface of the airfoil. The computational grid consists of 531 surface grid points. A total number of 11466 RANS-CFD simulations for different combinations of the three parameters are available – all of them were generated using the DLR flow solver TAU, [29] with the SST turbulence model. The design of experiment is displayed in Fig. 5.
A fraction of $70\,\%$ of the CFD data base was used for the training of the shallow neural network, $20\,\%$ for validation and $10\,\%$ for testing. While the full CFD surface solution serves as output, the information at the grid points which are closest to the 36 wind tunnel sensor locations is used as input.
The neural network was set up using PyTorch, [30]. The size of the second hidden layer was chosen to be $n_2 = 268$, as this corresponds to the dimension of a

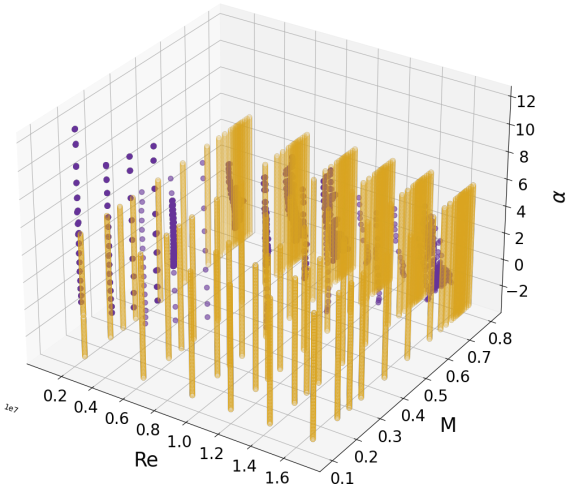


FIG 5. Parameter combinations for which wind tunnel data (purple) and CFD data (orange) is available.

POD subspace with a relative information content of $99.99\,\%$. For determining the number of neurons for the first hidden layer, the learning rate and the activation function, a study was carried out using the hyperparameter optimization framework Optuna, [31], with 500 samples generated with the TPESampler. A list of the optimized and fixed hyperparameters is given in Tab. 1.

Parameter	Value
Size of input layer s	36
Size of first hidden layer n_1	185
Size of second hidden layer n_2	268
Size of output layer N	531
Learning rate	0.00248
Activation function	ReLU

TAB 1. Hyperparameters of the shallow neural network and their corresponding values

The model was trained for 5000 epochs with batch sizes of 64 training samples and 12 validation samples using the mean squared error to the full surface solution. For comparison with Gappy POD models, a POD subspace of dimension $n_2 = 268$ was computed based on the training and validation data set.

6.2.1. Prediction of CFD data

In a first step, the shallow neural network model, as well as Gappy POD using Ridge regression and Bayesian Gappy POD with GPR have been assessed on the testing data set, i.e. on CFD data which was not included in the training and validation of the models. For all three models, the CFD information at the closest grid points of the 36 wind tunnel sensor locations has served as input. Fig. 6 shows a result of the neural network in comparison to the target CFD solution for a parameter combination from the testing data set.

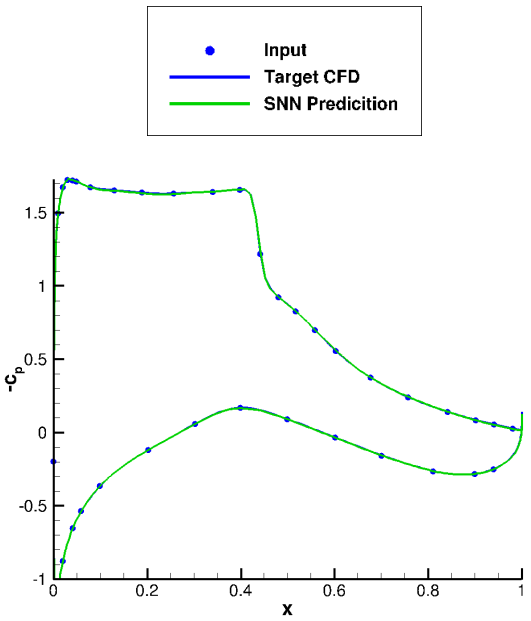


FIG 6. Exemplary result for the reconstruction of CFD data from pseudo-sensor information; $\mathbf{Re = 9.6 \times 10^6, M = 0.71, \alpha = 5.4°}$

As can be seen in the figure, an almost perfect match with the target CFD solution is obtained for a Reynolds number of $\mathrm{Re} = 9.6 \times 10^6$, a Mach number $\mathrm{M} = 0.71$ and angle of attack $\alpha = 5.4°$. The same behavior can be observed for every other testing data point. For a quantitative comparison of the shallow neural network result with the two Gappy POD methods, the averaged mean squared error on the testing data set with respect to the target CFD solution is visualized in Fig. 7.

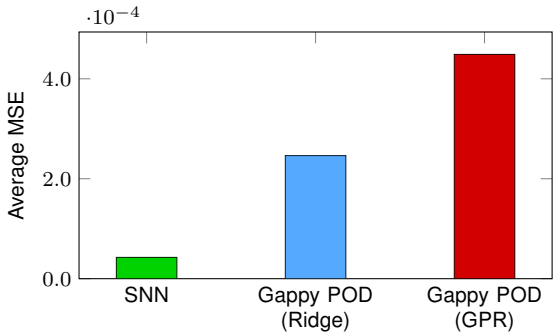


FIG 7. Average mean squared error with respect to the target CFD solution for the Shallow Neural Network, Gappy POD with Ridge regression and Gappy POD with GPR evaluated at the testing data set.

As one can see from the bar plot, the average mean squared error of the shallow neural network prediction is one order of magnitude smaller than for the two Gappy POD methods. Overall, the shallow neural network generalizes very well on the testing data set and clearly outperforms the Gappy POD methods in reconstructing CFD data.

6.2.2. Prediction of wind tunnel data

Next, we go a step further and use the same neural network for the flow reconstruction based on wind tunnel sensor data. Note, that due to the training issue discussed in Section 6.1.2, the experimental data was not part of the training data set and thus, the neural network has never seen this kind of data before. A result for a Reynolds number of $\mathrm{Re} = 2.90 \times 10^6$, a Mach number $\mathrm{M} = 0.20$ and an angle of attack $\alpha = -0.54°$ is displayed in Fig. 8.

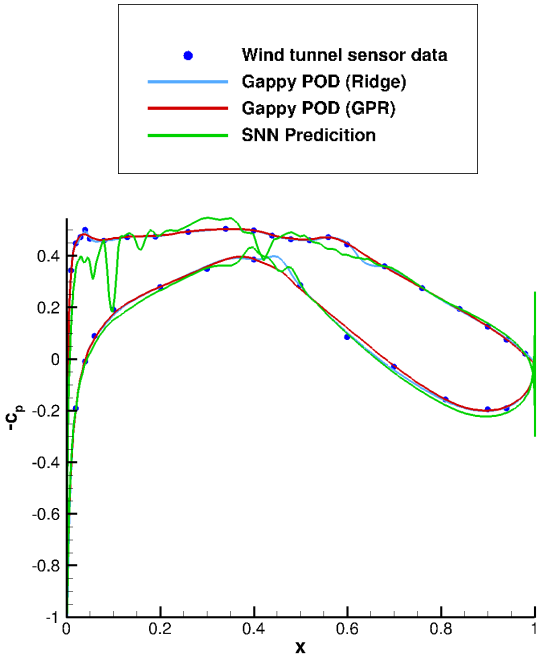


FIG 8. Exemplary results for the reconstruction of the pressure coefficient distribution from wind tunnel sensor data; $\mathbf{Re = 2.90 \times 10^6, M = 0.20, \alpha = -0.54°}$

For this test case, the neural network prediction shows strong unphysical fluctuations at the front part of the wing and was not able to match the wind tunnel data accurately. In comparison, the two Gappy POD methods accurately describe the measurement data whilst having a smooth physical trend.
This qualitative observation is also in line with the quantitative comparison of the three methods: The mean squared error of the three different methods with respect to the 36 sensor measurements averaged for all available 7719 wind tunnel data sets is displayed in Fig. 9.
As indicated by the bar plot, the average mean squared error of the shallow neural network prediction is 1.76×10^{-2} and thus one order of magnitude larger than the one for Gappy POD with Ridge regression of 3.74×10^{-3}. By far the best agreement with the wind tunnel measurements is obtained with Bayesian Gappy POD with GPR which gives an average mean squared error of 3.01×10^{-5}, which is again two orders of magnitude lower than the one of Gappy POD with Ridge regression.

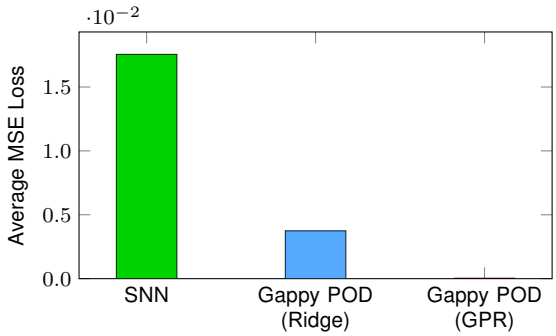


FIG 9. Average mean squared error of the three methods with respect to the 36 wind tunnel sensors for the Shallow Neural Network, Gappy POD with Ridge regression and Gappy POD with GPR evaluated for all available wind tunnel data sets.

In summary, for the test case at hand, the shallow neural network trained on CFD data is not able to compete with the two Gappy POD methods for the reconstruction of flow fields based on wind tunnel data.

7. CONCLUSION

In this report, the most important results of the DLR-Airbus research project on data fusion from 2018 until 2022 have been presented. After a short introduction to ordinary and regularized Gappy POD, new extensions have been derived: Weighted Gappy POD allows to define individual weighting factors for the sensor measurements and is therefore a way to incorporate expert knowledge during the data fusion. Bayesian Gappy POD is a natural generalization of Gappy POD as it includes ordinary and regularized Gappy POD as special cases. Moreover, more advanced regression techniques can be considered for the solution of the Gappy POD problem, which makes Bayesian Gappy POD more flexible than the ordinary approach and the problem of linearity can be overcome. Using Bayesian regression techniques, the data fusion result is given in terms of a probability distribution. Mean and standard deviation as well as samples drawn from the probability distribution provide valuable information on the data fusion result. Results were shown for the industrial-relevant XRF1 test case. In comparison to regularized Gappy POD, a better agreement with the wind tunnel measurements can be obtained which is underlined by a root mean squared error which is $47\,\%$ smaller than for the state-of-the-art approach. The results serve as an indication for the applicability of the method within an industrial design process.
In the current approach, we take a Bayesian perspective on the regression problem of Gappy POD which allows to include information on the variance of the sensor measurements. Future work is needed towards a fully probabilistic approach which also enables to take information on the uncertainty associated with the CFD data into account. Furthermore, the incorporation of constraints for Bayesian Gappy POD might be a valuable extension. Intelligent sensor placement and techniques for the detection of outliers can help to further improve the results.
As an alternative to the Gappy POD methods, a shallow neural network approach proposed in [15] was investigated for data fusion tasks and evaluated on a 2D airfoil test case. The shallow neural network was able to outperform regularized and Bayesian Gappy POD when trained and evaluated on CFD data only. However, it showed significantly decreased performance in the fusion of numerical and experimental data. A fundamental problem of this approach can be seen in the training issue: While in data fusion, we are aiming to assess the shallow neural network on wind tunnel sensor data, we in general do not have proper wind tunnel training data. The neural network has perfectly learned how to map sparse CFD-based pseudo sensor data onto the corresponding full solution, but has never seen wind tunnel data before it enters the offline phase. Further research could investigate whether results improve if pressure sensitive paint data is included in the training data set. Another idea would be to use the shallow neural network as dimensionality reduction technique in a “Gappy SNN” approach: After the training of the network based on CFD data, its solution space takes the role of the POD subspace in the Gappy POD approach. This has the advantage that the solution space of the neural network is not only learned from the CFD data, as is the case for POD. Because (pseudo) sensor information serve as input to the network, the solution space is created such that the observability of the full solution from the sensor measurements is taken into account which may be beneficial. An investigation of this approach has to be also postponed to future work.

Acknowledgement

The first author kindly thanks Airbus for providing the test cases, as well as the numerical and experimental data used in the scope of this project. Furthermore, the first author is very grateful to DLR and Airbus for providing the funding for this research.

Contact address:

anna.bertram@mathemagier.de

References

[1] Zhong-Hua Han, Stefan Görtz, and Ralf Zimmermann. Improving variable-fidelity surrogate modeling via gradient-enhanced kriging and a generalized hybrid bridge function. *Aerospace Science and Technology*, 25(1):177–189, 2013. DOI: 10.1016/j.ast.2012.01.006.

[2] A. J. Keane. Wing Optimization Using Design of Experiment, Response Surface, and Data Fusion Methods. *Journal of Aircraft*, 40(4):741–750, 2003. DOI: 10.2514/2.3153.

[3] Marc C. Kennedy and A. O'Hagan. Predicting the output from a complex computer code when fast approximations are available. *Biometrika*, 87(1):1–13, 2000. DOI: 10.1093/biomet/87.1.1.

[4] Zhong-Hua Han, Zimmermann, and Stefan Görtz. Alternative Cokriging Method for Variable-Fidelity Surrogate Modeling. *AIAA Journal*, 50(5):1205–1210, 2012. DOI: 10.2514/1.J051243.

[5] Anna Bertram and Ralf Zimmermann. Theoretical investigations of the new Cokriging method for variable-fidelity surrogate modeling. *Advances in Computational Mathematics*, 44(6):1693–1716, 2018. DOI: 10.1007/s10444-017-9585-1.

[6] Zhong-Hua Han and Stefan Görtz. Hierarchical Kriging Model for Variable-Fidelity Surrogate Modeling. *AIAA Journal*, 50(9):1885–1896, 2012. DOI: 10.2514/1.J051354.

[7] Alex Feldstein, David Lazzara, Norman Princen, and Karen Willcox. Multifidelity Data Fusion: Application to Blended-Wing-Body Multidisciplinary Analysis Under Uncertainty. *AIAA Journal*, 58(2):889–906, 2020. DOI: 10.2514/1.J058388.

[8] Anna Bertram. *Data-driven variable-fidelity reduced order modeling for efficient vehicle shape optimization*. PhD Thesis, TU Braunschweig, Braunschweig, Germany, 2018. DOI: 10.24355/dbbs.084-201811231243-0.

[9] Anna Bertram, Carsten Othmer, and Ralf Zimmermann. Towards Real-time Vehicle Aerodynamic Design via Multi-fidelity Data-driven Reduced Order Modeling. In *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Reston, USA, 2018. American Institute of Aeronautics and Astronautics. DOI: 10.2514/6.2018-0916.

[10] S. Ashwin Renganathan, Kohei Harada, and Dimitri N. Mavris. Aerodynamic Data Fusion Toward the Digital Twin Paradigm. *AIAA Journal*, 58(9):3902–3918, 2020. DOI: 10.2514/1.J059203.

[11] R. Everson and L. Sirovich. Karhunen-Loève procedure for gappy data. *Journal of the Optical Society of America A*, 12(8):1657, 1995. DOI: 10.1364/JOSAA.12.001657.

[12] Tan Bui-Thanh, Murali Damodaran, and Karen E. Willcox. Aerodynamic data reconstruction and inverse design using proper orthogonal decomposition. *AIAA Journal*, 42(8):1505–1516, 2004. DOI: 10.2514/1.2159.

[13] K. Willcox. Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition. *Computers & Fluids*, 35(2):208–226, 2006. DOI: 10.1016/j.compfluid.2004.11.006.

[14] Michael Mifsud, Alexander Vendl, Lars-Uwe Hansen, and Stefan Görtz. Fusing wind-tunnel measurements and cfd data using constrained gappy proper orthogonal decomposition. *Aerospace Science and Technology*, 86:312–326, 2019. DOI: 10.1016/j.ast.2018.12.036.

[15] N. Benjamin Erichson, Lionel Mathelin, Zhewei Yao, Steven L. Brunton, Michael W. Mahoney, and J. Nathan Kutz. Shallow neural networks for fluid flow reconstruction with limited sensors. *Proceedings. Mathematical, physical, and engineering sciences*, 476(2238), 2020. DOI: 10.1098/rspa.2020.0097.

[16] Anna Bertram, Philipp Bekemeyer, and Matthias Held. Fusing Distributed Aerodynamic Data Using Bayesian Gappy Proper Orthogonal Decomposition. In *AIAA AVIATION 2021 FORUM*. American Institute of Aeronautics and Astronautics, 2021. DOI: 10.2514/6.2021-2602.

[17] T. Bui-Thanh, Murali Damodaran, and Karen Willcox. Proper orthogonal decomposition extensions for parametric applications in compressible aerodynamics. In *21st AIAA Applied Aerodynamics Conference*, page 519, Reston, USA, 2003. American Institute of Aeronautics and Astronautics. DOI: 10.2514/6.2003-4213.

[18] Thomas Franz and Matthias Held. Data fusion of cfd solutions and experimental aerodynamic data. In DLR and ONERA, editors, *ODAS 2017*, 2017.

[19] Gene Howard Golub and Charles F. van Loan. *Matrix computations*. Johns Hopkins studies in mathematical sciences. Johns Hopkins Univ. Press, Baltimore, USA, 4. edition, 2013. ISBN: 9781421408590.

[20] Trevor Hastie, Robert Tibshirani, and Jerome H. Friedman. *The elements of statistical learning: Data mining, inference, and prediction*. Springer series in statistics. Springer, New York, USA, second edition, corrected at 12th printing 2017 edition, 2017. ISBN: 978-0-387-84857-0.

[21] Andrej N. Tikhonov and Vasilij Ja. Arsenin. *Solutions of ill-posed problems*. Scripta series in mathematics. Winston, Washington, D.C., 1977. ISBN: 978-0-470-99124-4.

[22] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, Cambridge, Mass., 3. print edition, 2008. ISBN: 978-0-262-18253-9.

[23] Alexander I. J. Forrester, András Sóbester, and Andy J. Keane. *Engineering design via surrogate modelling: A practical guide*. Wiley, Hoboken, USA, 2008. ISBN: 978-0-470-06068-1.

[24] Christopher M. Bishop. *Pattern recognition and machine learning*. Information science and statistics. Springer, New York, USA, corrected at 8th printing 2009 edition, 2009. ISBN: 978-0-387-31073-2.

[25] Philipp Bekemeyer, Anna Bertram, Derrick A. Hines Chaves, Mateus Dias Ribeiro, Andrea Garbo, Anna Kiener, Christian Sabater, Mario Stradtner, Simon Wassing, Markus Widhalm, Stefan Goertz, Florian Jaeckel, Robert Hoppe, and Nils Hoffmann. Data-Driven Aerodynamic Modeling Using the DLR SMARTy Toolbox. In *AIAA AVIATION 2022 Forum*, Chicago, Illinois, 2022. American Institute of Aeronautics and Astronautics. DOI: 10.2514/6.2022-3899.

[26] Anna Bertram, Philipp Bekemeyer, and Matthias Held. Bayesian Gappy Proper Orthogonal Decomposition for Aerodynamic Data Fusion. *In preparation*, to appear in 2022.

[27] Martin D. Buhmann. *Radial Basis Functions*. Cambridge University Press, Cambridge, UK, 2003. ISBN: 978-0-511-54324-1. DOI: 10.1017/CBO9780511543241.

[28] P.H. Cook, M.C.P. Firmin, and M.A. McDonald. *Aerofoil RAE 2822: pressure distributions, and boundary layer and wake measurements*. RAE, 1977.

[29] Dieter Schwamborn, Thomas Gerhold, and Ralf Heinrich. The DLR TAU-Code: Recent Applications in Research and Industry. In P. Wesseling, editor, *ECCOMAS CFD 2006: Proceedings of the European Conference on Computational Fluid Dynamics*, [s. l.], 2006. Delft University of Technology.

[30] The PyTorch Community. Pytorch. `https://pytorch.org/`, Dezember 2021.

[31] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A Next-generation Hyperparameter Optimization Framework. In Ankur Teredesai, editor, *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ACM Digital Library, pages 2623–2631, New York, USA, 2019. Association for Computing Machinery. DOI: 10.1145/3292500.3330701.