Aerospace Science and Technology 163 (2025) 110301

Contents lists available at ScienceDirect

Aerospace Science and Technology

journal homepage: www.elsevier.com/locate/aescte

Multi-fidelity transonic aerodynamic loads estimation using Bayesian neural networks with transfer learning

Andrea Vaiuso [a], Gabriele Immordino [a,b,*], Marcello Righi [a], Andrea Da Ronch [b]

[a] School of Engineering, Zurich University of Applied Sciences ZHAW, Winterthur, 8400, Switzerland
[b] Faculty of Engineering and Physical Sciences, University of Southampton, Southampton, SO17 1BJ, United Kingdom

ARTICLE INFO

Communicated by Mehdi Ghoreyshi

ABSTRACT

Multi-fidelity surrogate models are of particular interest in aerospace applications, as they combine the computational efficiency of low-fidelity simulations with the accuracy of high-fidelity models. This methodology, often implemented via data fusion, aims to reduce the cost of data generation while preserving predictive accuracy. Despite the widespread use of traditional machine learning techniques to improve surrogates and perform data fusion tasks, there remains a need for novel approaches that further improve predictive reliability—particularly in terms of uncertainty quantification—without substantially increasing the computational cost of generating high-fidelity training samples. In this study, we propose a Bayesian neural network framework designed for multi-fidelity prediction of transonic aerodynamic data, employing transfer learning to integrate computational fluid dynamics data of varying fidelities. The probabilistic nature of the model allows also quantification of the uncertainty in the input space, making it well suited for analyzing the inherently complex and nonlinear behavior of the transonic aerodynamic responses under investigation. Our results demonstrate that the proposed multi-fidelity Bayesian model outperforms classical data fusion Co-Kriging method, both in accuracy and generalization capabilities on unseen data.

1. Introduction

In aerospace engineering, the accurate prediction of aerodynamic loads is extremely important for the design and optimization of aircraft. In particular, the transonic regime is characterized by complex flow phenomena, such as shock wave formation and boundary-layer separation, which introduce significant nonlinearities into the aerodynamic behavior of aircraft components. These nonlinearities pose challenges for traditional modeling techniques, which often struggle to balance accuracy with computational efficiency.

To address these challenges, there is increasing interest in combining data from models of different fidelities. By leveraging the strengths of both low- and high-fidelity models, this approach enhances predictive accuracy while reducing computational costs, making it valuable not only for aerodynamic load estimation but also for applications in aeroelasticity, flight dynamics, and so forth [64,18,1,62].

In general, low-fidelity (LF) models are employed to calculate physical quantities of interest in complex systems with a cost-effective approach in situations where rapid assessments are required [8,4,57]. This is achieved through the identification of the key physical phenomena, the application of simplifying assumptions to reduce degrees of freedom, and the use of basic mathematical models or empirical data to approximate the behavior of the system [36,29,60,40,38]. These models, such as simplified vortex approaches [51,63] and reduced-order methods [36,30,31,14], provide essential aerodynamic characteristics without incurring the significant computational demands of high-fidelity (HF) Computational Fluid Dynamics (CFD) simulations [48]. However, these LF models often struggle to achieve sufficient accuracy when predicting regions of the flight envelope that exhibit a highly nonlinear behavior [17,65].

To bridge this gap, a valid approach is represented by Data Fusion (DF) techniques, which aim to leverage the strengths of models based on different fidelities to create a multi-fidelity surrogate model [23,49]. A common DF approach involves combining a data-driven model based on LF samples with a small number of HF points, with the goal of improving the overall prediction accuracy without incurring the prohibitive costs of fully HF studies [41,34,45,61]. This approach is particularly beneficial when the generation of HF data at specific physical conditions is computationally prohibitive, yet LF and inexpensive models are available. A traditional and widely used DF technique, Co-Kriging (CK), utilizes

* Corresponding author at: Faculty of Engineering and Physical Sciences, University of Southampton, Southampton, SO17 1BJ, United Kingdom.
E-mail address: G.Immordino@soton.ac.uk (G. Immordino).

https://doi.org/10.1016/j.ast.2025.110301
Received 3 February 2025; Received in revised form 29 April 2025; Accepted 4 May 2025
Available online 9 May 2025
1270-9638/© 2025 The Author(s). Published by Elsevier Masson SAS. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

Gaussian Process to correct LF predictions using correlations between different fidelity levels, thereby improving precision and quantifying uncertainty effectively [35,45]. Forrester et al. [19] demonstrated CK capability in multi-fidelity wing optimization, highlighting its utility in integrating diverse data sources. This method is still prevalent in recent aerodynamic studies and airfoil optimizations [9,22,35]. Other recent popular strategies include Recursive Cokriging [61], extended hierarchical Kriging [45], and multi-fidelity Gaussian processes [41], which have been applied to a wide range of aerospace problems such as high-pressure compressor rotors, eVTOL aircraft, and the NASA Common Research Model in transonic conditions.

The integration of machine learning (ML) has significantly advanced the DF field, offering rapid and accurate predictions despite the costs associated with the initial training, particularly with Deep Learning (DL) [53,56] and more novel and emergent solutions such as graph neural networks [54]. The demand for large quantities of high-quality training data, often limited and costly to obtain through CFD, underscores the importance of DF techniques for leveraging multi-fidelity sources. Transfer Learning (TL) is a frequently employed approach in DL networks for DF tasks, involving initial training with LF data followed by fine-tuning with sparse HF samples to enhance precision [66,32]. In Chakraborty study [6], a multi-fidelity physics-informed deep neural network uses TL to predict reliability analysis outcomes effectively, surpassing standalone models, while in Liao et al. approach [33], TL integrates LF and HF data to improve a CNN-based model accuracy for aerodynamic optimization.

Despite these advances, many ML-based DF surrogate models still lack direct mechanisms for robust uncertainty quantification—a critical aspect for ensuring prediction reliability in many aerospace applications. Although ML-based approaches have shown superior performance over traditional methods like CK in capturing data nonlinearity [17,21,46], they typically do not provide inherent measures of how confident they are in their predictions. Bayesian Neural Networks (BNNs) address this limitation by incorporating Bayesian inference to estimate distributions over network weights [58], providing probabilistic interpretations of predictions that capture both model and data uncertainties [26]. This is particularly useful in conceptual design and optimization workflows. By examining the confidence intervals around predicted quantities, it is possible to assess the reliability of these estimates and identify regions of the flight envelope that require further refinement. In addition, these uncertainty bounds play an important role in robust optimization, where they help avoid solutions that appear optimal under a single deterministic prediction, but carry a high risk of underperformance when potential variations and modeling inaccuracies are considered. Recent research, such as Meng et al. [39] and Sharma et al. [52], explores BNNs in multi-fidelity models, highlighting challenges in computational complexity and data integration mismatches. Kerleguer et al. [28] propose a hybrid approach combining Gaussian process regression and BNNs to mitigate these challenges, demonstrating promising avenues for integrating diverse data sources effectively. Multi-fidelity frameworks with BNNs have demonstrated robust and reliable uncertainty estimates for complex problems—even in scenarios where HF data are scarce [39,52,28].

The aim of our paper is to develop a ML-based multi-fidelity surrogate model that predicts integrated aerodynamic loads, using exclusively BNN layers, and integrating TL for the data fusion process. This approach has led to the design of a more straightforward architecture, solely based on the DL paradigm, capable of capturing the complex nonlinearities of transonic flows. This framework is implemented using open-source code to ensure replicability and accessibility for further research and applications.

This paper is structured as follows: Section 2 introduces the concepts of Bayesian Neural Networks and Transfer Learning, and presents the Multi-Fidelity Bayesian Neural Network with Transfer Learning (MF-BayNet) model. Section 3 focuses on the Benchmark Super Critical Wing (BSCW) test case, describing how the transonic aerodynamic loads were generated at multiple fidelities and demonstrating that integrating numerous LF samples with a limited number of mid- to high-fidelity CFD points improves both predictive accuracy and uncertainty quantification. Section 4 extends the methodology to a more complex full–configuration aircraft, highlighting how multi-fidelity data fusion captures complex rotor–wing interactions more effectively than classical surrogate approaches. Finally, Section 5 summarizes the conclusions drawn from these studies and outlines avenues for future research.

2. Methodology

This section details the steps involved in creating the surrogate model. We implemented a multi-fidelity framework integrating BNNs with TL technique to harness diverse data sources and enhance model generalization. The approach includes quantification of uncertainty to ensure reliable predictions, along with systematic optimization of model hyperparameters for optimal performance. CK model is introduced in order to perform a comparative analysis, benchmarking the efficacy of our proposed method.

2.1. Bayesian neural networks and variational inference

Bayesian Neural Networks (BNNs) extend standard neural networks by incorporating a Bayesian inference to estimate uncertainty in model predictions. Instead of learning a single set of weights, BNNs place a prior distribution over the network weights θ, denoted by $p(\theta)$, which is considered a model hyperparameter and is usually randomly initialized. Given a training dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, the objective is to compute the posterior distribution of the weights,

$$p(\theta \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid \theta)\, p(\theta)}{p(\mathcal{D})}, \tag{1}$$

where $p(\mathcal{D} \mid \theta) = \prod_{i=1}^{N} p(y_i \mid x_i, \theta)$ is the likelihood of the data under the weights θ, and $p(\mathcal{D})$ is the marginal likelihood (or evidence).

For a regression task, the predictive distribution at a new input x^* marginalizes out the uncertainty in the weights:

$$p(y^* \mid x^*, \mathcal{D}) = \int p(y^* \mid x^*, \theta)\, p(\theta \mid \mathcal{D})\, d\theta. \tag{2}$$

This integral is typically intractable due to the complexity of $p(\theta \mid \mathcal{D})$. Hence, approximate methods such as the Laplace approximation [37], Hamiltonian Monte Carlo [42], and other Markov Chain Monte Carlo techniques [7] are commonly employed.

Variational Inference (VI) [20] is another popular approach to approximate the true posterior $p(\theta \mid \mathcal{D})$ by introducing a simpler *variational* distribution $q(\theta)$ and then minimizing the Kullback–Leibler (KL) divergence between the two distributions:

$$\mathrm{KL}\big(q(\theta) \,\|\, p(\theta \mid \mathcal{D})\big) = \int q(\theta) \log\left(\frac{q(\theta)}{p(\theta \mid \mathcal{D})}\right) d\theta. \tag{3}$$

Because $p(\theta \mid \mathcal{D})$ is unknown in closed form, one typically derives an equivalent objective known as the *Evidence Lower BOund* (ELBO). Under a regression setting with mean squared error (MSE) as the negative log-likelihood (assuming Gaussian noise), the variational objective can be written (up to constants independent of θ) as:

$$\mathcal{L}(\theta) = \underbrace{\sum_{i=1}^{N} \mathbb{E}_{q(\theta)}\Big[(y_i - f_\theta(x_i))^2\Big]}_{\mathcal{L}_{\mathrm{MSE}}} + \underbrace{\mathrm{KL}\big(q(\theta) \,\|\, p(\theta)\big)}_{\text{Regularization}}, \tag{4}$$

where $p(\theta)$ is the prior over the weights. This objective encourages $q(\theta)$ to produce accurate predictions for the training data while remaining close to the prior. In practice, the gradient of this loss is estimated via stochastic optimization, making it amenable to large-scale problems.

We implement our Bayesian Neural Network framework in `PyTorch` using the `torchbnn` library, which provides flexible modules for varia-

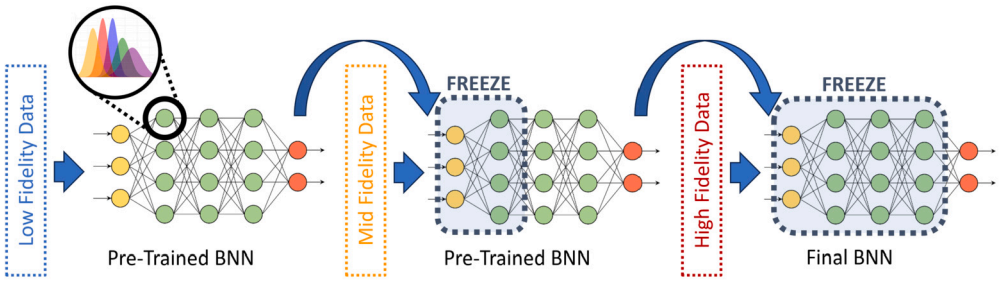


Fig. 1. Schematic of the MF-BayNet architecture. Yellow and red neurons represent the input and output layers, while the green ones the hidden layers. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

tional layers and inference. For optimization, we employ the ADAptive Moment estimation (`Adam`) optimizer during the backpropagation phase to update the variational parameters of $q(\theta)$ and thereby approximate the posterior distribution effectively.

2.2. MF-BayNet

The core of our methodology revolves around a multi-fidelity surrogate model that combines BNNs with TL, herein referred to as MF-BayNet, which is designed to predict integrated transonic aerodynamic loads, specifically lift and pitching moment coefficients, across different flight conditions. First, the process begins by training a BNN on a large dataset of LF data to capture general trends of aerodynamic behavior over the full design space. Then, the model undergoes TL on mid-fidelity (MF) data where certain layers of the model are frozen, so that low-level features remain fixed; then a subset of layers is retrained using the MF dataset. This partially corrects LF inaccuracies—especially in regions with transonic shocks or mild boundary-layer interactions. Finally, a limited number of the remaining trainable layers are fine-tuned using a few expensive HF samples. This final step adds important fine-scale corrections to the network, enabling it to capture strong shocks, boundary-layer separations, and other highly nonlinear phenomena.

2.2.1. Transfer learning process

In the context of DF, once the model has captured the basic understanding of the general behaviors during initial training, the first layers of the model are frozen by increasing the fidelity of the training set. This means that all frozen neurons have fixed values of weights during training, which in the case of BNN are represented by the mean and standard deviation of each probability distribution. After this phase, we get a pre-trained network that has gained a basic knowledge of the problem. Then, the subset of trainable parameters of the model is retrained on one or more small sets of increasingly higher fidelity data. A schematic of the architecture is illustrated in Fig. 1.

The TL approach significantly reduces the computational cost and time required for training due to the reduced subset of trainable neurons, while improving the model ability to make accurate predictions using different fidelities and emphasizing the importance of higher fidelity data. In addition, this method relies only on DL principles, which makes it more straightforward and robust compared to other methods. Two TL processes were executed: the first on MF data and the second on HF data points to fine-tune the model. In the second process, fewer layers were frozen compared to the first, allowing for more refined adjustments. By progressively transferring learned representations from one fidelity level to the next, the MF-BayNet framework fully exploits large amounts of approximate LF data while carefully incorporating MF and HF points to correct errors in the more challenging regions. This approach drastically reduces the need for large HF datasets while achieving high accuracy and built-in uncertainty quantification. By restricting this last stage of training to a minimal number of layers, or even just the output layer, the network retains most of the generalized aerodynamic knowledge acquired from the LF and MF datasets. Consequently, the model maintains predictive capabilities even in areas lacking HF data, potentially outperforming models trained exclusively on limited HF samples. The TL process is explained in Algorithm 1.

Algorithm 1 MF-BayNet Training with Low-, Mid-, and High-Fidelity Datasets.

Input: Low-Fidelity Dataset D_{LF}, Mid-Fidelity Dataset D_{MF}, High-Fidelity Dataset D_{HF}, Initial Model M
Output: Trained Multi-Fidelity Model $M_{MF-BayNet}$
Step 1: Train on Low-Fidelity Data
$M \leftarrow$ Initialize Bayesian Neural Network (BNN)
$M_{LF} \leftarrow \text{train}(M, D_{LF})$
Step 2: Transfer Learning to Mid-Fidelity Data
$\text{freeze}(M_{LF}, N_{frz}^{LF \to MF})$
$M_{MF} \leftarrow \text{train}(M_{LF}, D_{MF})$
Step 3: Transfer Learning to High-Fidelity Data
$\text{freeze}(M_{MF}, N_{frz}^{MF \to HF})$
return $M_{MF-BayNet} \leftarrow \text{train}(M_{MF}, D_{HF})$

2.2.2. Prediction means and confidence interval

Once the TL process is completed, we use Monte Carlo Sampling [3] (MCS) to obtain the prediction means and standard deviations from the surrogate model, which are used for estimating the uncertainty and reliability in predictions. First, multiple forward passes are performed through the BNN. Each pass generates a different set of weights due to the nondeterministic behavior of the model, effectively creating an ensemble of models. Next, the outputs of these forward passes are averaged to obtain the mean prediction. This gives us an estimate of the expected value of the prediction. Finally, the standard deviation of the outputs from multiple forward passes is calculated to assess the uncertainty in the predictions, incorporating both epistemic and aleatoric components. The choice of a good number of passes is therefore a trade-off between computational efficiency and the accuracy of the uncertainty estimation. A higher number of passes leads to a more precise evaluation of both the mean prediction and its associated uncertainty, but at the cost of increased inference time. In practical applications, an optimal balance must be found based on the specific requirements of the task. For example, in scenarios where fast predictions are needed, a lower number of passes may be preferable despite a slight reduction in accuracy, whereas in offline analyses, a larger number of passes can be used to maximize reliability. To determine the optimal setting, conduct sensitivity analyses or compare the uncertainty estimates against reference solutions is suggested.

2.2.3. Uncertainty quantification

Uncertainty quantification (UQ) is essential for understanding the reliability of the model predictions. The MF-BayNet surrogate model provides a probabilistic interpretation of predictions, offering insights into both model and data uncertainty. The total uncertainty in the prediction, the predictive uncertainty (P_u), is defined as the sum of epistemic (E_u) and aleatoric uncertainty (A_u) [5,12,13].

$$P_u = E_u + A_u \tag{5}$$

Epistemic uncertainty refers to the uncertainty in the model parameters. This can be visualized as the spread of the posterior weight distribution $p(w|D)$. In ML, this type of uncertainty emerges when the model has not encountered data that adequately represents the entire design domain, or when the domain itself needs further refinement or completion. This type of uncertainty arises due to deficiencies from a lack of knowledge or information [10,47]. In contrast, aleatoric uncertainty arises from the inherent variability in the input data. Given a specific input and fixed weight parameters, high aleatoric uncertainty indicates that the output estimate is subject to noise. This kind of uncertainty refers to the intrinsic randomness in the data, which can derive from factors such as data collection errors, sensor noise, or noisy labels [47].

This distinction is particularly important in multi-fidelity modeling with TL, where the TL process impacts the aleatoric uncertainty associated with different fidelity inputs. Only the subset of neurons with trainable parameters captures the probabilistic information from these inputs, making discrepancies between fidelity levels a significant source of aleatoric uncertainty. Although higher-fidelity datasets are generally more reliable, this is not always clear beforehand, complicating the assessment of overall uncertainty during TL. These complexities make it challenging to fully isolate aleatoric uncertainty, as some degree of epistemic uncertainty remains intertwined with it. In the first part of the results presented in this study, the uncertainty primarily reflects epistemic contributions from the model itself. In contrast, the second part specifically examines aleatoric uncertainty, though a complete separation between the two remains difficult due to their inherent interactions in the multi-fidelity TL process.

2.3. Model optimization

Choosing the right hyperparameters of the network is a complex task. This requires a deep understanding of the model architecture and the specific characteristics of the data at different fidelity levels in order to create a good hyperparameter optimization process. We implemented a Bayesian optimization [55] strategy to obtain the best set of hyperparameters for each model. Bayesian optimization strength lies in its iterative approach to fine-tuning hyperparameters using Bayesian probability distributions, rather than exhaustively testing every possible combination. Each iteration involves training the network with a defined set of hyperparameters and optimizing them based on past trials performance with respect to the validation set metric. This cycle repeats until the optimal outcome is attained.

The design parameters targeted for the optimization process include the number of units per layer (N_{units}), the total number of layers in the model (N_{layers}), activation functions, optimization function, batch size, number of epochs, and the learning rate for each training phase. After the $i-th$ TL phase, the model needs to be retrained with a different learning rate value (lr_i). Other parameters include the prior distribution (initial probability distribution for each weight), determined by the mean (μ_{prior}) and variance (σ_{prior}) values of a Gaussian function, and the number of layers to freeze during TL (N_{frz}). The last step is a critical task, as freezing too many layers might prevent the model from adapting to the new, higher-fidelity data, while freezing too few layers can lead to excessive retraining and potentially overfitting. The design space for all hyperparameters, including the possible values and step size for each variable, is presented in Table 1. A total number of 300 trials per model tested has been executed, with an average time per trial of 3 minutes. The dataset was divided into 70% for training and 30% for validation.

During each trial of the optimization process, the model parameters, which are represented by the mean and variance values of the probability distributions of each neuron in the BNN model, along with bias values of the activation functions, are optimized based on the Mean Absolute Error (MAE) on the validation set.

Table 1
Hyperparameters design space in Bayesian optimization.

Hyperparameter	Value	Step size
N_{layers}	3 to 6	1
N_{units}	16 to 176	16
lr_i	$1 \cdot 10^{-4}$ to $1 \cdot 10^{-1}$	$5 \cdot 10^{-3}$
μ_{prior}	-1.5 to 1.5	$5 \cdot 10^{-2}$
σ_{prior}	$1 \cdot 10^{-4}$ to $1 \cdot 10^{-2}$	$5 \cdot 10^{-4}$
N_{frz}	1 to $(N_{layers} - 1)$	1
Activation f	ReLU, PReLU, LeakyReLU	–

2.4. Co-Kriging

Co-Kriging (CK) is a traditional approach for integrating low- and high-fidelity simulation data and serves as a benchmark in this study. Following the methodology outlined in Da Ronch et al. [9], the CK function, denoted as $\hat{\eta}$, is first computed from LF evaluations and applied at HF sample points. The input parameters at these HF locations, x_i, are then expanded to include the CK-estimated LF values, forming the augmented vector $x_i^{aug} = [x_i, \hat{\eta}(x_i)]$. This expanded dataset enables a refined CK function, $\hat{\eta}(x_i^{aug})$, to enhance correlation modeling between fidelity levels.

To implement this, the CK model was initially trained on the MF dataset, augmented with LF data interpolated via a single-fidelity Bayesian neural network (introduced in Section 3). Once the CK model converged, it was used as a surrogate to further interpolate MF points onto the HF dataset. This process was iterated once more, generating an augmented HF dataset that incorporated all three fidelity levels, ultimately yielding a Gaussian process-based model capable of multi-fidelity predictions.

2.5. Performance metrics

The error metrics ε_μ [%], ε_σ [%], and ε_{tot} [%] are computed to evaluate the performance of the models predictions on test cases for each output variable. The *Percentage Error on Mean Prediction* (ε_{μ_i} [%]) is derived by calculating the MAE between prediction means $\tilde{\mu}_y$ calculated on MCS, and ground truth values y, normalized by the output label range, thus $\text{Range}_i = |\max(\mathcal{D}_i) - \min(\mathcal{D}_i)|$ where $\mathcal{D}_i$ represents the vector of values on dataset column i:

$$\varepsilon_{\mu_i}[\%] = \left(\frac{\text{MAE}(y, \tilde{\mu}_y)}{\text{Range}_i} \right) \times 100. \tag{6}$$

The *Percentage Error on Standard Deviation* (ε_{σ_i} [%]) measures the model uncertainty in its predictions (lower is better). The predicted standard deviations $\tilde{\sigma}_y$ calculated on MCS are averaged and then normalized by the range of the corresponding output label:

$$\varepsilon_{\sigma_i}[\%] = \left(\frac{\text{avg}(\tilde{\sigma}_y)}{\text{Range}_i} \right) \times 100. \tag{7}$$

The *Total Percentage Error* (ε_{tot} [%]) provides an aggregated measure of the overall prediction error across all output labels. This metric is computed as the root mean square of the individual ε_μ [%] values, given by:

$$\varepsilon_{tot}[\%] = \sqrt{\frac{1}{n} \sum_{i=1}^{n} \left(\varepsilon_{\mu_i}[\%] \right)^2}, \tag{8}$$

where n is the number of output labels.

3. Finite wing test case

This section outlines the application of the MF-BayNet surrogate model for predicting aerodynamic loads on the Benchmark Super Critical Wing (BSCW) by leveraging a combination of low-, mid-, and high-fidelity datasets. The identified input parameters include angle of attack

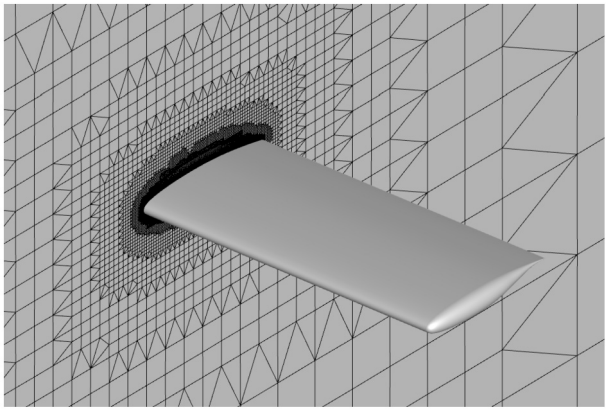


Fig. 2. Impression of BSCW high-fidelity grid.

(AoA) and Mach number, which characterize the influence of the transonic flow regime on the aerodynamic performance. Lift coefficient (C_L) and pitching moment coefficient (C_M) represent the two outputs to predict. To evaluate the effectiveness of the MF-BayNet approach, multiple models were developed, optimized, and compared. Three single-fidelity BNN models were trained to evaluate the impact of TL technique. Each model was individually optimized to minimize the MAE on its respective validation dataset, simulating the development of models using one fidelity source at a time. Additionally, the same optimized LF-BNN model was used to provide LF interpolated samples to expand MF dataset for the CK. Finally, the MF-BayNet surrogate model was trained using the TL framework that sequentially incorporated LF, MF, and HF datasets on subset of layers. The number of layers frozen from the left for each TL phase is also defined as a hyperparameter and optimized. The optimized hyperparameters for all models used in this test case are reported in Appendix B.1.

3.1. Multi-fidelity datasets

The BSCW, featured in the AIAA Aeroelastic Prediction Workshop [24], is a transonic, rigid, semi-span wing with a rectangular planform and a supercritical airfoil shape, specifically the NASA SC(2)-0414 profile. The wing is elastically suspended on a flexible mount system with two degrees of freedom, pitch and plunge. However, in our case study, we focus solely on the wing itself, excluding the pitch-plunge system. The BSCW exhibits complex aerodynamic phenomena such as shock wave motion, shock-induced boundary-layer separation, and interactions between shock waves and detached boundary layers. These nonlinearities present significant challenges for model predictions. The BSCW configuration and geometry make it an ideal test case for generating low-, mid-, and high-fidelity aerodynamic data using various techniques. An impression of the half-span BSCW is shown in Fig. 2.

The datasets include a large number of LF samples and a limited number of mid- and high-fidelity samples. LF data come from panel method, providing quick but approximate solutions. Mid- and high-fidelity data, derived from CFD simulations with different number of grid points, offer detailed and accurate results but are computationally expensive. This hierarchical arrangement of fidelity levels is a key to our multi-fidelity approach, providing both broad aerodynamic coverage and targeted refinement where nonlinear effects become significant. In particular, at relatively low AoA and Mach numbers, the aerodynamic response remains weakly nonlinear, with Mach exhibiting a stronger influence on the loads than AoA. As both parameters increase, nonlinear phenomena become more significant - shock waves form on the wing and boundary-layer separation occur, leading to pronounced changes in lift and pitching moment. In particular, stall onset is observed at relatively low AoA under transonic conditions, highlighting the strong coupling between Mach and AoA in this regime. For this study, AoA ranges from 0 to 4 deg and Mach from 0.70 to 0.84 (see Fig. 3(a)). By combining these multi-fidelity datasets, the MF-BayNet surrogate model is trained and fine-tuned to capture the full spectrum of aerodynamic behaviors relevant to the BSCW.

Table 2
Summary of datasets used for training the MF-BayNet surrogate model.

Fidelity Level	Number of Samples	Simulation Approach
Low	625	Panel Method
Mid	49	RANS - Coarse Grid
High	7	RANS - Fine Grid

Fig. 3(b) also highlights differences in aerodynamic coefficient predictions from each fidelity level, emphasizing the necessity for a model capable of effectively distinguishing and emphasizing the key features of each fidelity. The aerodynamic coefficients were calculated using a reference chord length of 0.4064 m, with C_M determined relative to 30% of the chord. Table 2 provides a summary of the datasets used for training the MF-BayNet surrogate model.

3.1.1. Low-fidelity

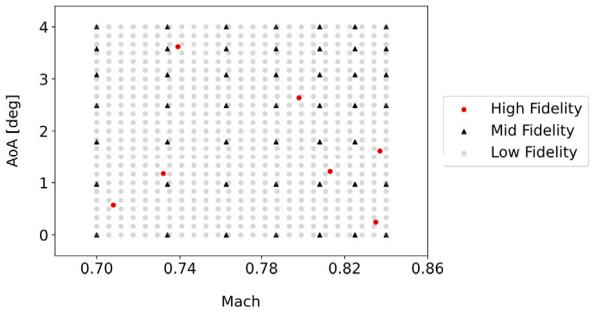
LF data were generated using XFoil, a popular tool for the design and analysis of subsonic airfoils which employs a combination of inviscid panel methods with a boundary layer analysis, allowing it to rapidly generate aerodynamic data. For this study, the BSCW wing profile was used. To create a comprehensive dataset, 25 points equally distributed along each parameter of the design space were used, resulting in a total of 625 samples. This extensive dataset offers a broad base of quick, approximate aerodynamic solutions. The XFoil-generated data were corrected applying the equations from Helmbold [25] to account for three-dimensional effects, thereby improving the accuracy for the low-aspect-ratio straight BSCW configuration. These corrections ensure that the LF data better represent the actual aerodynamic behavior of the wing in three-dimensional flow conditions, making the dataset more valuable for the multi-fidelity model training.

3.1.2. Mid-fidelity

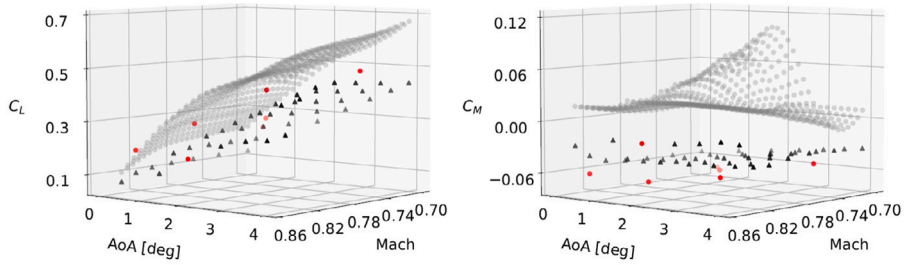
MF data were obtained from RANS simulations using SU2 v7.5.1 software [16] with a relatively coarse grid of $2.5 \cdot 10^6$ elements. The grid is a hybrid type, with structured elements on the wing surface and in the first layers of the boundary layer, while voxel elements were used for the rest of the computational domain. The domain itself extends 100 chord lengths from the solid wall to the farfield. The Grid Convergence Index (GCI) for this coarse grid was calculated to be approximately 5.4%, indicating a moderate level of discretization error. The RANS equations were closed using the one-equation Spalart-Allmaras turbulence model. Convergence was monitored using the Cauchy method applied to the lift coefficient, with a variation threshold of 10^{-7} across the last 100 iterations. A $1v$ multigrid scheme was adopted to accelerate convergence. Convective flow discretization utilized the Jameson-Schmidt-Turkel central scheme with artificial dissipation, and flow variable gradients were computed using the Green Gauss method. The biconjugate gradient stabilization linear solver with an ILU preconditioner was selected. The samples were distributed to refine the highly nonlinear regions, particularly at high combinations of AoA and Mach number. This strategic sampling ensures that the MF data provide enhanced resolution in critical areas, capturing the complex aerodynamic interactions more effectively. A total of 49 samples were generated, capturing viscosity and transonic flow effects better than LF data and at a lower computational cost than HF simulations, but still with moderate discretization errors.

3.1.3. High-fidelity

HF dataset was previously generated [27], incurring no further computational cost. It consists of 58 RANS simulations with a fine grid comprising $15.6 \cdot 10^6$ elements. The GCI for this fine grid is around



(a) Distribution of sample points per fidelity level.



(b) Lift and pitching moment coefficients across the design space.

Fig. 3. Design variable distribution (AoA-Mach) and aerodynamic coefficient predictions for each fidelity level for the finite wing test case.

0.6%, indicating a very low discretization error and that the solution is nearly grid-independent. These HF simulations provide the most detailed resolution of transonic nonlinear features and the most accurate aerodynamic load predictions, serving as a benchmark for validating the MF-BayNet surrogate model. However, they are an order of magnitude more expensive than MF and millions of times more than LF. For fine-tuning, 7 simulations were identified as the minimum number of samples necessary to characterize the discrepancies between mid- and high-fidelity predictions, represented as red dots in Fig. 3. These samples were not selected arbitrarily; instead, they were deliberately chosen to ensure coverage of the most critical flow conditions, emphasizing regions where transonic shock formation, boundary-layer separation, and other nonlinear effects become more pronounced. This targeted selection captures the most challenging flow regimes with the fewest simulations, thereby minimizing the cost of HF runs while preserving accuracy. This selection was informed by the authors' prior knowledge of the relevant aerodynamic phenomena [27]. The remaining samples were used as a test set to validate the model, demonstrating that with a minimum number of simulations for fine-tuning, the model can achieve a relatively low error on the entire HF dataset.

3.1.4. Discussion

The inherent nonlinearity of aerodynamic behavior, particularly in the transonic regime, is a significant challenge when developing surrogate models. Fig. 3 provides a clear illustration of how the aerodynamic coefficients, C_L and C_M, vary nonlinearly across the M-AoA design space. The LF data, generated through XFoil, shows a more linear response, failing to capture the abrupt changes and complex phenomena like shock-wave formation and boundary-layer separation that occur at higher Mach numbers and AoA. This limitation necessitates the incorporation of higher-fidelity data to correct for these deficiencies. The MF RANS simulations, with their ability to better resolve the flow nonlinear characteristics, particularly near the critical Mach number, offer a significant improvement. However, it is the HF data that most accurately captures the sharp gradients and nonlinearities essential for predicting aerodynamic loads in the transonic regime. These differences across fidelity levels underscore the importance of a multi-fidelity approach. By integrating data from different fidelities, the MF-BayNet surrogate model more effectively captures and predicts the complex nonlinear dynamics of transonic aerodynamics.

Table 3
Number of samples used for training, validation, and testing across the three fidelity levels for the finite wing test case. HF test samples (51 in total) are used exclusively for final evaluation and are not included in training or validation.

Fidelity Level	Train Samples	Validation Samples	Test Samples
Low	437	188	-
Mid	33	15	-
High	5	2	51

3.2. Results

Here, the performance of the MF-BayNet surrogate model, three single-fidelity dataset and the CK method are evaluated by comparing their predictions of aerodynamic coefficients against an independent HF test dataset made of 51 CFD samples, which was excluded from both the HF model training and multi-fidelity training. Specifically, each single-fidelity dataset was partitioned into 70% for training and 30% for validation. The same partitioning was used to train MF-BayNet during TL process. Dataset sizes are shown in Table 3. Further, the MF-BayNet surrogate model uncertainty estimates are compared to CK method, highlighting its superior ability to capture and quantify prediction uncertainty. Results also include comparisons with BNNs trained on each dataset individually without TL, with model architectures optimized.

3.2.1. Comparison of models performance

Table 4 compares the performance of MF-BayNet with CK in predicting C_L and C_M. It also includes the results of the network trained individually on each dataset without utilizing TL, as well as results from the MF-BayNet framework where LF samples were excluded from training. For the latter, the model was first trained on MF samples and then TL was applied to fine-tune the model on HF samples. All results are averaged over a k-fold validation with a k value of 5.

The results demonstrate that MF-BayNet significantly outperforms all other models for both C_L and C_M. Specifically, MF-BayNet achieves

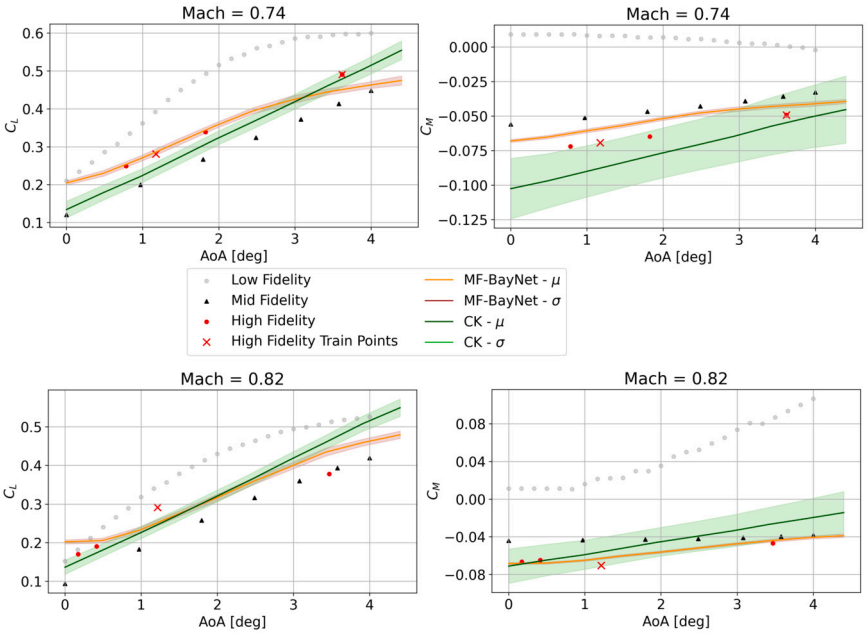


Fig. 4. Comparison of MF-BayNet and CK predictions for C_L and C_M at two Mach numbers ($M = 0.74$, left panel; $M = 0.82$, right panel) across varying AoA for the finite wing test case. Points from different fidelity datasets have been interpolated for 2D plot visualization.

Table 4
Performance metrics in percentage [%] for different models for the finite wing test case. Errors are calculated on HF test set.

Model Name	$\varepsilon_{\mu_{CL}}$	$\varepsilon_{\sigma_{CL}}$	$\varepsilon_{\mu_{CM}}$	$\varepsilon_{\sigma_{CM}}$	ε_{tot}
BNN LF	14.43	1.59	42.02	1.41	31.42
BNN MF	12.63	6.88	7.73	4.92	10.47
BNN HF	15.29	9.00	5.62	5.96	11.52
CK	7.53	2.98	7.70	9.46	7.61
MF-BayNet (MF+HF)	5.81	6.95	6.48	4.44	6.15
MF-BayNet (LF+MF+HF)	**4.24**	1.37	**5.50**	0.71	**4.91**

a $\varepsilon_{\mu_{CL}}$ of 4.2% and a $\varepsilon_{\mu_{CM}}$ of 5.5%, which are approximately half of those obtained with CK, which has errors of 7.5% and 7.7% for C_L and C_M respectively. Moreover, the standard deviations errors ($\varepsilon_{\sigma_{CL}}$ and $\varepsilon_{\sigma_{CM}}$) for MF-BayNet are also lower, indicating more consistent and reliable predictions. The BNN models, whether trained on LF, MF or HF datasets, show higher prediction errors and standard deviations, reinforcing the efficacy of the TL and fine-tuning processes utilized by MF-BayNet. Finally, these results confirm the added value of combining datasets across all fidelity levels; removing LF samples from the MF-BayNet training slightly increased prediction errors, indicating that lower fidelity data still contribute information that enhances generalization capability.

To better illustrate model performance, Figs. 4 show predictions of C_L and C_M at different Mach numbers. The left panels present predictions at $M = 0.74$, while the right panels show predictions at $M = 0.82$. For C_L predictions, the MF-BayNet surrogate model (orange line) closely follows the HF dataset (red dots), effectively leveraging knowledge acquired through transfer learning and fine-tuning. The MF-BayNet predictions exhibit narrow confidence intervals (brown shaded regions), indicating high predictive certainty. In contrast, the CK model (dark green line) struggles to capture the correct trend, especially at higher AoA and Mach numbers, and shows significantly larger confidence intervals (green shaded regions, particularly noticeable in the right panels). The nonlinear nature of the aerodynamic behavior is particularly evident at $M = 0.82$, where the C_L curve shows significant nonlinearity, especially at higher angles of attack. This nonlinearity arises from the formation of shock waves on the wing surface, which induce abrupt changes in the pressure distribution, leading to sharp variations in C_L. The MF-BayNet surrogate model successfully captures these complex behaviors, unlike the CK model, which struggles more in representing these nonlinear effects. Similarly, the C_M predictions reveal pronounced nonlinear trends as the aerodynamic center shifts due to interactions between shock waves and the boundary layer. These interactions complicate the aerodynamic response, causing rapid changes in C_M. The MF-BayNet surrogate model ability to closely match the HF data, along with its narrower confidence intervals, highlights its strength in modeling these challenging nonlinear phenomena. Similar considerations apply to Fig. 5, where the predictions of C_L and C_M are shown with fixed AoA values and varying Mach. It can be seen that the CK model occasionally produces predictions closer to the HF data points, but still fails to capture the more complex trends indicated by the MF and HF datasets. Instead, it continues to align more closely with the lower fidelity dataset trend. Examination of both figures reveals that the observed nonlinearities arise from the combined influence of Mach number and AoA variations, with the MF Baynet providing good accuracy over the entire parameter space, under both small and large nonlinearity ranges.

In summary, the single-fidelity model trained on LF data failed to capture the nonlinear transonic aerodynamics due to the oversimplified nature of the underlying data. Including MF data improved predictions by resolving more complex flow features, though discretization errors still constrained generalization. The HF-only model achieved low accuracy and poor extrapolation due to the limited number of samples. In contrast, the proposed multi-fidelity framework, which integrates LF, MF, and HF data through TL, achieved superior accuracy and generalization, outperforming both single-fidelity models and CK. While CK benefited from combining data sources, it underperformed in highly nonlinear regions and provided less reliable uncertainty estimates.

3.2.2. *Mid-fidelity dataset size impact*

Since MF data strikes a good balance - being significantly less computationally expensive than HF data, yet providing much richer information compared to LF data - we investigated the impact of MF sampling on model performance. To perform this sensitivity analysis, we first generated a denser set of MF points. The original sample consisted of 49 points arranged in a 7×7 grid, while the newly generated points formed a denser 13×13 grid of 169 points. Model performance was evaluated incrementally as the dataset size increased, using 5-fold cross-validation, with each fold divided into 70% training and 30% testing subsets.

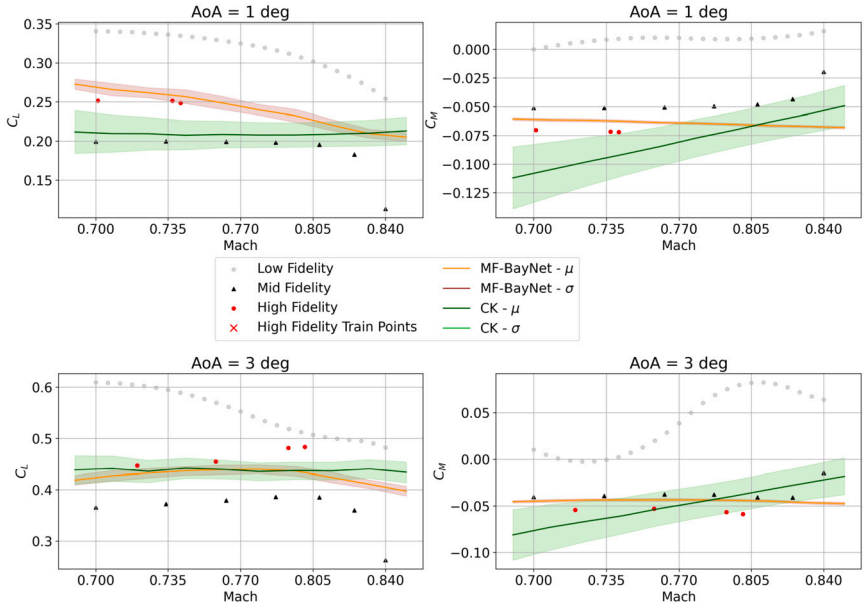


Fig. 5. Comparison of MF-BayNet and CK predictions for C_L and C_M at two AoA values ($AoA = 1.0$ deg, left panel; $AoA = 3.0$ deg, right panel) across varying Mach numbers for the finite wing test case. Points from different fidelity datasets have been interpolated for 2D plot visualization.

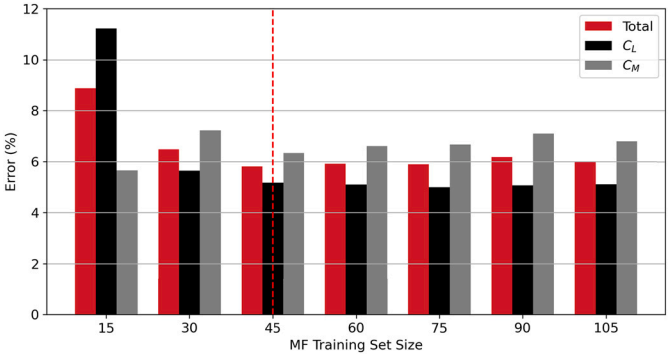


Fig. 6. Influence of MF training set size on model performance for the finite wing test case.

In general, increasing the number of MF samples leads to a reduction in prediction error. This behavior is expected, as more MF points allow the model to better capture the underlying physics. However, as shown in Fig. 6, the improvement quickly reaches a saturation point where further increases yield minimal or no additional benefit, reflecting the inherent limitations of the simplified representation provided by MF data compared to HF data. In our analysis, performance gains become negligible after approximately 50 MF samples.

3.2.3. Aleatoric uncertainty quantification

To assess the impact of aleatoric uncertainty on the predictions, we re-trained the baseline version of our MF-BayNet model—termed the "Vanilla MF-BayNet"—by systematically adding noise to each fidelity dataset separately. The training dataset was augmented following an augmentation factor value (AF) from 20% to 80%, by adding Gaussian noise with a standard deviation from 2% to 10% of the mean output label value, applied to either C_L (Fig. 7) or C_M (Fig. 8).

Results show that ε_μ generally decreases with increasing noise when added to LF and HF datasets for C_L, whereas the MF datasets exhibit an increasing trend in mean error. In contrast, for C_M, the mean error increases with noise in both LF and MF datasets, but adding noise to HF datasets tends to reduce the mean error, especially as AF increases. Across both C_L and C_M, ε_σ remains low and stable, indicating that noise primarily impacts the mean error rather than the variability of model predictions.

As the AF increases in MF datasets, the impact of noise on the mean error becomes more pronounced, while higher AF generally show decreased mean errors in HF datasets. This behavior can be explained by considering the robustness of HF data to noise and the sensitivity of MF data. HF datasets contain more accurate and detailed information, making the model less sensitive to noise and, in some cases, benefiting from noise through regularization, which improves the model generalization and reduces the mean error [2]. Adding noise to MF datasets results in a significant increase in mean error. This higher sensitivity to noise is due to the moderate levels of accuracy and detail in MF datasets, which are significantly degraded by the introduction of noise. The balance between the inherent detail and the added noise disrupts the model learning process, leading to increased errors. LF datasets, already less accurate and detailed, exhibit a mixed response to noise. In some cases, the noise further degrades data quality, increasing the mean error, while in others, it has minimal impact or even aids the model by introducing beneficial variability.

The relatively minor changes in the standard deviation of errors, despite varying noise levels, suggest that the model overall prediction uncertainty remains stable. This indicates that while noise impacts the accuracy (mean error) of predictions, it does not significantly affect the model confidence in its predictions. These observations underscore the importance of understanding and mitigating aleatoric uncertainty to improve predictive modeling performance, particularly by leveraging the robustness of HF data and addressing the sensitivity of MF data.

We also studied the effect of aleatoric uncertainty on model predictions when adding Gaussian noise to the entire LF dataset ($AF = 100\%$) (refer to Fig. 9). The noise standard deviation ranged from 2% to 10%, applied independently to the coefficients C_L and C_M. The graph depicts how the introduction of this noise influences the error associated with these coefficients separately. It is evident that as the noise standard deviation increases, the error in the predictions also increases. This trend is consistent for both C_L and C_M. This indicates that the predictive capability of the model is increasingly compromised as more noise is introduced. However, the standard deviation remains relatively stable for both coefficients, suggesting that while accuracy decreases, the consistency of the model predictions does not fluctuate significantly.

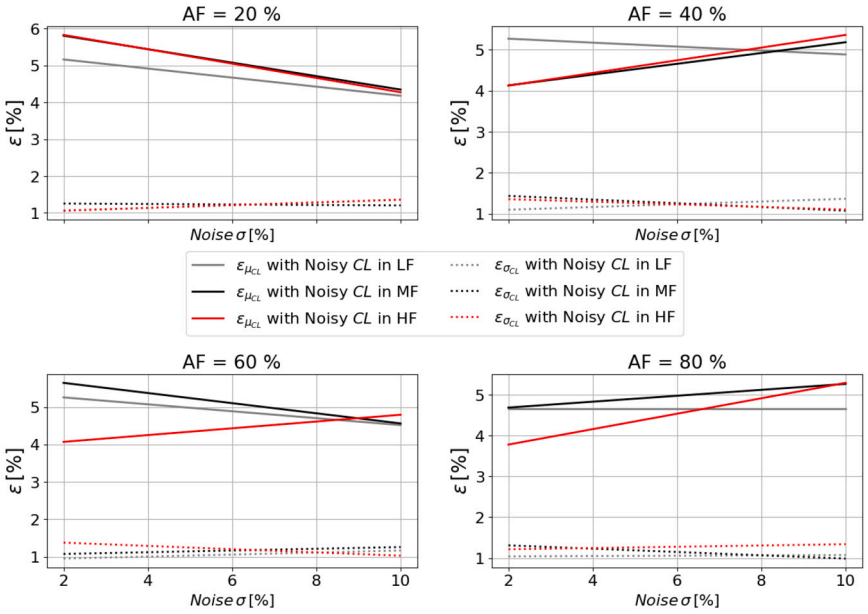


Fig. 7. Impact of aleatoric uncertainty on the model predictions, in terms of error on the standard deviation and mean value of the C_L. Training dataset augmented from 20% to 80% by adding Gaussian noise to the C_L of each dataset separately with a standard deviation ranging from 2% to 10%.

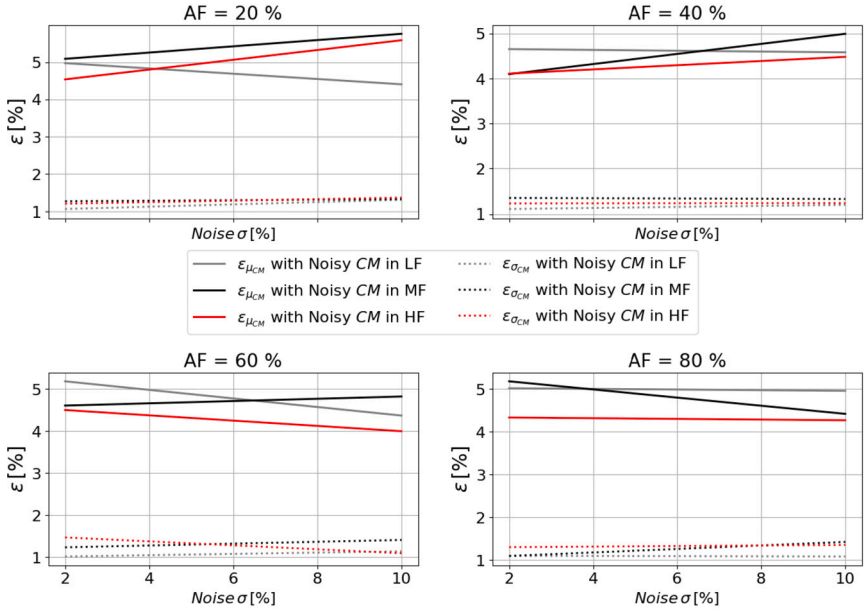


Fig. 8. Impact of aleatoric uncertainty on the model predictions, in terms of error on the standard deviation and mean value of the C_M. Training dataset augmented from 20% to 80% by adding Gaussian noise to the C_M of each dataset separately with a standard deviation ranging from 2% to 10%.

This behavior can be explained by the structure of our MF-BayNet surrogate model. Adding noise to the entire LF dataset increases the initial errors, which propagate through the subsequent training stages. The model learns from this noisier data initially, which degrades its baseline understanding and leads to higher errors that carry over even after TL processes. Moreover, the stable standard deviation suggests that the noise primarily affects the bias (mean error) rather than the variance of the predictions. This indicates that while the model accuracy decreases due to systematic errors introduced by the noise, its confidence in the consistency of predictions remains unaffected. Therefore, enhancing data quality or employing robust modeling techniques to mitigate the impact of noise can lead to better accuracy without compromising prediction consistency.

4. Full–configuration test case

This section presents a five-propeller electric vertical takeoff and landing (eVTOL) test case that illustrates the critical aerodynamic challenges associated with rotor-wing interactions in a real scenario. The test case combines two different fidelity datasets and it is inspired by the modeling exercise carried out by the authors in the EASA funded project MODEL-SI [15]. The vehicle under consideration has four longitudinally symmetric lift propellers mounted forward and aft of the wing, and one pusher propeller mounted at the tail for forward flight. Thanks to this configuration, the drone can switch between a so-called “helicopter mode” and an “airplane mode” depending on the use of the propellers. The fuselage is 4 m long, while the wingspan is 6 m with

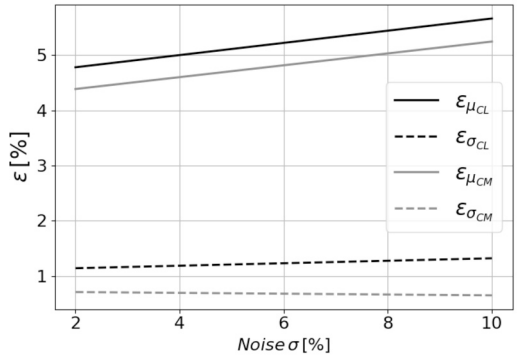


Fig. 9. Effect of aleatoric uncertainty on model predictions: Gaussian noise, with a standard deviation ranging from 2% to 10%, was independently added to the C_L and C_M of the entire LF dataset.

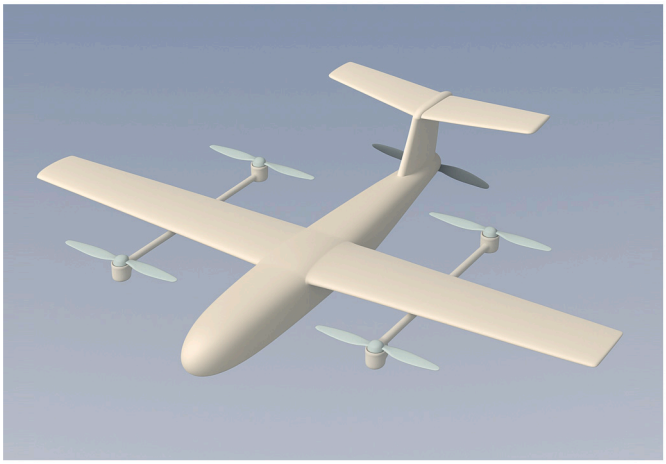


Fig. 10. Impression of eVTOL configuration.

a mean chord of 1 m of the NACA0012 profile. An impression of the configuration can be obtained from Fig. 10.

The primary aerodynamic challenge is the complex interaction between the propellers, wing, and fuselage. Forward propellers generate rotor wakes that flow over the wing and impinge directly on the aft propellers, significantly altering the local flowfield. This perturbation changes the lift distribution, adds extra wing loading, and increases drag. Also, the rear propellers suffer a decrease in thrust when operating in the wake of the front rotors for the same power input, resulting in spatially varying performance that is difficult to predict with lower fidelity methods alone. Once the wing itself generates lift, its near-field flow curvature affects the inflow conditions of the forward rotors. Depending on how far forward of the wing the propellers are located, the swirling flow can reduce rotor efficiency if the incoming flow is distorted or partially blocked. These phenomena occur over a wide flight envelope, from vertical takeoff and forward flight to the transition between helicopter and airplane mode, making the problem highly nonlinear.

To capture the most important physical parameters that affect the rotor-wing aerodynamics of the eVTOL vehicle, the inputs include AoA, freestream velocity U_∞, and rotational speeds of five propellers: pusher (PP), front right (FR), front left (FL), rear right (RR), and rear left (RL). The multi-fidelity framework predicts two outputs for each of the five propellers: thrust coefficient (K_T) and torque coefficient (K_Q). Specifically, the model produces $K_{T_{\text{PP}}}, K_{Q_{\text{PP}}}$ for the pusher propeller and $K_{T_{\text{FR}}}, K_{Q_{\text{FR}}}, K_{T_{\text{FL}}}, K_{Q_{\text{FL}}}, K_{T_{\text{RR}}}, K_{Q_{\text{RR}}}, K_{T_{\text{RL}}}, K_{Q_{\text{RL}}}$ for the four lift propellers.

The MF-BayNet surrogate model was trained using the TL framework, sequentially incorporating LF and MF datasets. The optimized hyperparameters used in this test case are reported in Appendix B.2.

Table 5
Summary of datasets used for training the MF-BayNet surrogate model.

Fidelity Level	Number of Samples	Simulation Approach
Low	2,000	BEM
Mid	250	VPM

4.1. Multi-fidelity datasets

The strongly coupled and nonlinear flow physics in this eVTOL configuration provide another representative test case for the surrogate aerodynamic model presented. Although HF simulations (e.g., RANS-based CFD) can offer greater accuracy, they become prohibitively expensive—and often difficult to converge—when exploring wide flight envelopes. Indeed, in this study, fully converged CFD data could not be generated due to excessive computational costs. By contrast, lower-fidelity simulations are more affordable yet struggle to capture the pronounced rotor–wing interactions and wake impingement effects. Consequently, a multi-fidelity approach provides a more balanced solution: it integrates fast, lower-fidelity analyses with higher-order corrections from MF data, ensuring that complex flow interactions are accurately captured without incurring prohibitive computing time.

To generate samples, we covered a wide range of operating conditions: AoA varies from -180 to 180 deg, U_∞ from 0 to 40 m/s, and each propeller rotational speed up to ±4000 RPM, depending on its position and direction of rotation. These ranges were chosen based on aerodynamic and flight mechanics considerations to ensure that the dataset covers realistic maneuvers and bounds of the flight envelope.

Latin Hypercube Sampling (LHS) strategy was used to generate the LF dataset, resulting in approximately 2000 data points covering the broad ranges described above. These LF points were derived from an analytical aerodynamic function, allowing efficient data generation over a wide parameter space. In contrast, a limited set of MF samples were obtained from more accurate but computationally expensive simulations, resulting in 250 points distributed like in Fig. 11. This smaller MF set refines and corrects the trends observed in the LF data. Table 5 schematically illustrates how both LF and MF data form the training dataset.

4.1.1. Low-fidelity

The LF aerodynamic model is inspired to the work by Davoudi [11] in which Momentum Theory and Blade Element Method (BEM) are exploited to assess induced velocity and thrust generated by a moving rotor. Momentum theory provides a global analysis of flow through an actuator disk, deriving thrust and power relationships from force and flow balances. BEM refines this approach by discretizing the rotor blade into spanwise segments, each analyzed independently using 2D sectional aerodynamics. These models are most commonly used for aeroacoustic studies [43] to develop flight mechanics models of drones, capturing variations in AoA, chord, and twist along the blade span at low computational cost. In essence, the induced velocity model is inspired by Peter's model [44] and represents a simple yet sufficiently accurate approach for most engineering applications, providing a valuable starting point for rapid iterative analysis in preliminary design.

4.1.2. Mid-fidelity

For the MF aerodynamic simulations, we employed the DUST framework [59]. It is designed to analyze the aerodynamics of non-conventional aircraft configurations at a level of detail that bridges low- and high-fidelity approaches by implementing a Vortex Particle Method (VPM). Compared to purely potential-based tools or blade-element methods, DUST offers superior accuracy in capturing three-dimensional effects while remaining considerably less expensive than full RANS-based simulations. In the model setup, care was taken to model rotors and propeller blades with the accurate chord and twist distributions. The results shown in this paper were obtained with the option called

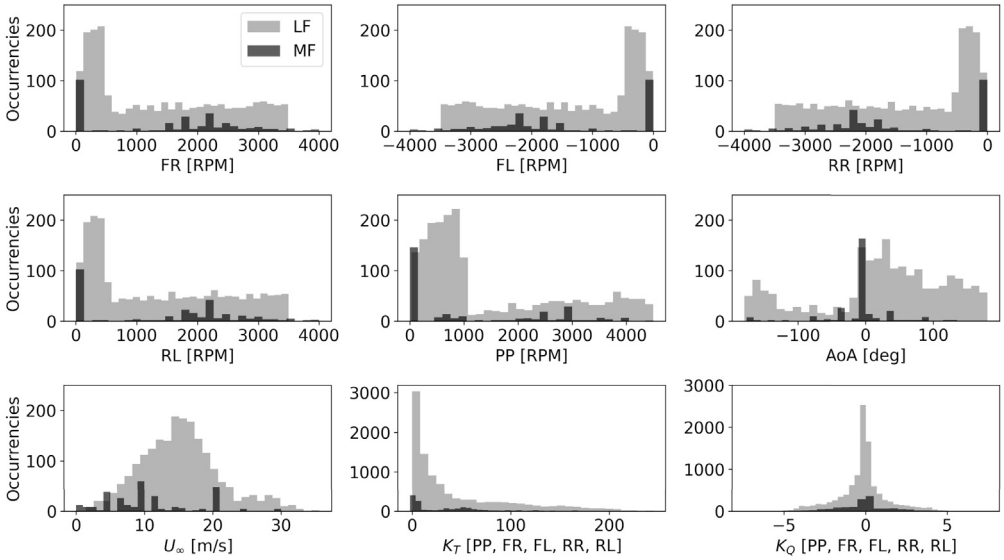


Fig. 11. Histograms of LF and MF datasets distribution across the design space. K_T and K_Q values are flatten for all propellers PP, FR, FL, RR, RL.

Table 6
Number of samples used for training, validation, and testing across the two fidelity levels for the full–configuration test case.

Fidelity Level	Train Samples	Validation Samples	Test Samples
Low	1,400	600	-
Mid	605	130	130

Table 7
Averaged thrust and torque coefficients percentage errors [%] for different models for the full–configuration test case. Errors are calculated respect to MF test set.

Model Name	ε_{K_T}	ε_{K_Q}	ε_{tot}	σ_{tot}
CK	3.93	4.98	4.50	0.26
MF-BayNet	2.99	3.12	3.06	0.87

"nonlinear lifting line" in the user manual [50]. We selected between 30 and 40 lifting lines per blade, the aerodynamic coefficients were given by look-up tables obtained from CFD for a number of blade sections. The maximum number of vortex particles was set to 100 million. The wing surface is discretized using approximately 3,000 panels for each side.

4.2. Results

The eVTOL test case presents a more challenging aerodynamic scenario than the finite wing test case, due to strong rotor-rotor and rotor-wing interactions. Such interactions introduce significant nonlinearities, wake impingement effects, and asymmetric flowfields, thereby increasing the difficulty of accurately capturing the physics across the design space. Consequently, this test case provides a high-dimensional benchmark for evaluating the generalizability of DF models.

Following the previous test case approach, the dataset is partitioned into training, validation, and test sets as detailed in Table 6. The MF test set, which is completely excluded from the training process, is used to compare the prediction accuracy of both MF-BayNet and CK. The comparison focuses on both mean prediction error and uncertainty quantification performance, with particular attention to the ability of each model to capture the variability introduced by complex aerodynamic couplings.

Table 7 shows the results for K_T and K_Q for the two different models. For K_T, the two models have comparable performance (3.9% vs. 2.9%), indicating that both methods adequately capture the basic rotor thrust variations under different flight conditions. However, for K_Q, MF-BayNet achieves a lower average error (3.1%) compared to CK (~5%). The torque coefficient is more sensitive to local flow features (e.g., rotor wake distortion or wing-propeller flow interference), and MF-BayNet's TL structure appears to be better able to leverage the MF data to capture these interaction effects.

Fig. 12 illustrates how the two DF approaches predict the K_T and K_Q values of the pusher propeller (PP) under "airplane mode" conditions, namely at $AoA = -5.0$ deg, $U_\infty = 10$ m/s, with the lift propellers off (0 RPM). Compared to CK, MF-BayNet follows the reference data more accurately, especially at medium to high RPM values, and captures the shift in aerodynamic behavior. In contrast, the CK prediction deviates significantly from the MF samples at higher speeds, reflecting a less effective DF for strong nonlinear rotor-propeller interactions. This highlights the benefit of the MF-BayNet framework, which better exploits the MF data to refine and correct the predictions in challenging flow regimes.

5. Conclusions

We developed a multi-fidelity framework using Bayesian neural networks and transfer learning (MF-BayNet) to enhance the accuracy of aerodynamic load prediction in the transonic regime over a three-dimensional wing and to reduce the required HF sample for training the surrogate model. Our primary objective was to leverage the strengths of BNNs, notably their ability to quantify uncertainty, alongside the efficiency and robustness of TL to improve predictive performance across datasets of different accuracy. The resulting surrogate model effectively integrates data of varying fidelities, harnessing the strengths of each fidelity level to produce superior predictive performance and robust uncertainty quantification.

The results obtained on two different test cases–a transonic wing and a full–configuration eVTOL vehicle–demonstrate that the MF-BayNet surrogate model not only surpasses traditional models trained on single-fidelity datasets but also outperforms CK method in predicting integrated aerodynamic loads. These results demonstrate the effectiveness of our hybrid approach in achieving superior accuracy and reliability, especially in complex aerodynamic scenarios where nonlinearity plays a critical role.

Our study also revealed that the impact of aleatoric uncertainty on model predictions varies significantly with the fidelity level of the

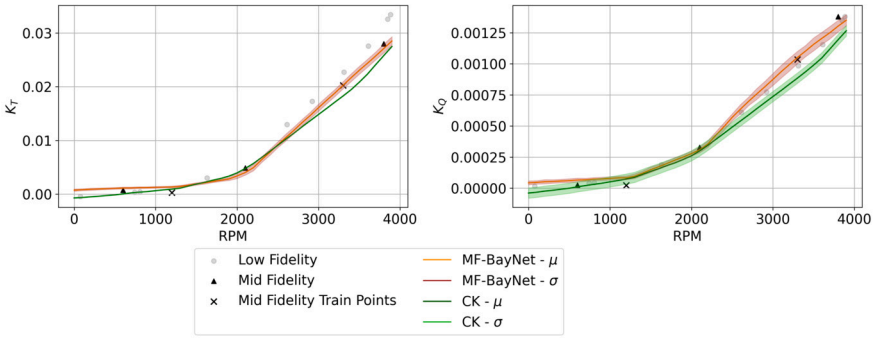


Fig. 12. Comparison of MF-BayNet and CK predictions in Airplane Mode for K_T and K_Q using ($AoA = -5.0$ deg, $U_\infty = 10$ m/s, and all the lift propellers set to 0 RPM) across varying RPM values for the pusher propeller (PP) in the full–configuration test case. Points from different fidelity datasets have been interpolated for 2D plot visualization.

datasets to which noise is added. Adding noise generally increases the mean error with noisy MF datasets while often decreasing the mean error with noisy HF datasets, particularly as the augmentation factor increases. Noisy LF datasets show a mixed response, with slight increases or decreases in mean error. The standard deviation of the error remains relatively stable across different noise levels, indicating that noise primarily affects the accuracy of predictions rather than the variability. These findings highlight the robustness of HF data to noise and the sensitivity of MF data, suggesting that while noise can regularize and improve model performance with HF data, it can degrade performance with MF data.

Additionally, we observed that adding noise to the entire LF dataset increases the initial errors, which propagates through the subsequent training stages, degrading overall accuracy. This is because the model, which first trains on the LF data, carries forward these systematic errors introduced by the noise even after TL processes. Consequently, while the mean error rises due to these systematic errors, the prediction consistency, as indicated by the stable standard deviation, remains unaffected. Therefore, understanding and addressing these subtleties is extremely important for enhancing predictive performance and mitigating the effects of aleatoric uncertainty.

Future work will explore the application of this framework to other relevant scenarios, in order to highlight the versatility and robustness of the model. The incorporation of more diverse and complex datasets will be a key focus to challenge and refine model performance. We plan to explore advanced data fusion techniques, including multi-fidelity Monte Carlo and Deep Gaussian Processes, to evaluate and compare the strengths and limitations of the MF-BayNet framework across various test cases.

The open-source implementation of the framework enables the research community to access, modify, and build upon our work, driving further advancements in the framework and wider applicability.[1]

CRediT authorship contribution statement

Andrea Vaiuso: Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Formal analysis, Data curation, Conceptualization. **Gabriele Immordino:** Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Marcello Righi:** Writing – review & editing, Supervision, Project administration, Funding acquisition. **Andrea Da Ronch:** Writing – review & editing, Supervision.

[1] https://github.com/andrea-vaiuso/MF-Baynet.

Table A.8
CPU hours for different simulation runs for the dataset creation process of the finite wing test case.

Dataset	Run time	
	(Full dataset)	(1 run)
Panel Method (LF)	0.3475 (625 runs)	0.00056
RANS - Coarse Grid (MF)	2,352 (49 runs)	48
RANS - Fine Grid (HF)	29,000 (58 runs)	500

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Gabriele Immordino reports financial support was provided by ZHAW University of Applied Sciences. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

The authors acknowledge the use of the IRIDIS High Performance Computing Facility, and associated support services at the University of Southampton, in the completion of this work. The project has been supported by Digitalization Initiative of the Zurich Higher Education Institutions (DIZH) grant 9710.Z.12.P.0003.05 from Zurich University of Applied Sciences (ZHAW).

Appendix A. Computational cost analysis

A.1. Finite wing test case

Tables A.8 and A.9 provide a comparative analysis of the computational costs associated with dataset creation and model performance.

Table A.8 highlights the substantial disparity in computational effort required across different fidelity levels. The HF dataset requires approximately 500 CPU hours per sample, making it over 10 times more expensive than the MF dataset and nearly a million times more computationally intensive than the LF dataset. This underscores the impracticality of relying exclusively on HF simulations for aerodynamic load predictions, reinforcing the necessity of multi-fidelity modeling. By integrating lower-fidelity data, the MF-BayNet framework effectively mitigates the reliance on computationally expensive HF simulations while maintaining a high level of predictive accuracy.

Table A.9 presents the GPU hours required for training and prediction across different models using an NVIDIA Quadro P2000 GPU, with execution times averaged over one dataset fold and results calculated using 5-fold cross-validation (k = 5). For single-fidelity datasets, the average training times were approximately 41 minutes for the LF model,

Table A.9
GPU hours for training and prediction costs for the proposed models for the finite wing test case.

Model	Training	Prediction
BNN LF	0.68	7.0e-6
BNN MF	0.58	7.0e-6
BNN HF	0.08	7.0e-6
MF-BayNet	0.97	7.0e-6
CK	0.08	1.4e-6

Table A.10
CPU hours for different simulation runs for the dataset creation process of the full–configuration test case.

Dataset	Run time	
	(Full dataset)	(1 run)
BEM (LF)	0.2 (2000 runs)	0.0001
DUST (MF)	7,800 (60 runs)	130

Table A.11
GPU hours for training and prediction costs for the proposed models for the full–configuration test case.

Model	Training	Prediction
MF-BayNet	1	7.7e-6
CK	0.08	1.4e-6

35 minutes for the MF model, and less than 5 minutes for the HF model. The MF-BayNet required approximately 58 minutes for training. In comparison, CK took about 5 minutes to fit from LF data using HF points as an additional feature to perform CK. Prediction times across all BNN models were consistent, averaging around 0.025 seconds to compute both mean and standard deviation values through MCS with 100 predictions. Conversely, the CK model demonstrated faster prediction times, requiring only about 0.005 seconds to generate mean and standard deviation predictions due to the simpler architecture and structure.

A.2. Full–configuration test case

In Table A.10 we report the total CPU hours required to generate the LF and MF datasets for the eVTOL test case. As shown, the analytic function (LF) approach is extremely efficient, requiring only 0.2 CPU hours for the entire dataset of 2000 runs (i.e., 0.0001 CPU hours per run). In contrast, the DUST (MF) simulations, while providing higher accuracy due to a more sophisticated modeling of the rotor-wing interaction, are significantly more expensive, amounting to 130 CPU hours per run or a total of 7,800 CPU hours for 60 runs.

Table A.11 shows the GPU hours required for training and prediction for different models using the same procedure as for the previous test case. MF-BayNet required approximately 60 minutes for training, while CK completed training in about 5 minutes. For prediction, MF-BayNet consistently took around 0.025 seconds per sample using MCS with 100 forward passes, whereas CK required only about 0.005 seconds.

Appendix B. Optimized network architectures

B.1. Finite wing test case

This section outlines the architecture used for the finite wing test case after the Bayesian hyperparameter optimization. The design parameters for the optimization are chosen accordingly to the multi-fidelity framework, and are described in Section 2.3. The final model is composed by 114,194 trainable weights for the pre-trained model with LF samples, 57,026 trainable weights for the pre-trained model with the first TL phase using MF samples, and 354 trainable weights during the last TL phase using HF samples. The model needed about 55 minutes to converge on an NVIDIA Quadro P2000 using a TL approach with specific training settings. For the first TL phase with MF samples, the model was trained using a batch size of 32 and a learning rate of 0.001. The maximum number of epochs was set to 8000, setting an early stopping criterion with a patience of 500. During the final TL phase with HF samples, the batch size was reduced to 1 to better capture HF details. This training followed a similar structure, with a maximum of 8000 epochs and an early stopping patience of 500. However, only one layer was unfrozen for fine-tuning, ensuring that the model retained the knowledge learned in previous stages while adapting to the HF dataset. The optimized hyperparameters for the four models developed are summarized in Table B.12.

Table B.12
Optimized hyperparameters of the Low-Fidelity (LF), Mid-Fidelity (MF), High-Fidelity (HF) BNN models, and the Transfer Learning (TL) MF-BayNet models for the finite wing test case.

Model	Hyperparameter	Best value
LF BNN	N_{layers}	6
	N_{units}	[176, 176, 144, 176, 176] + [2] output
	lr	0.0016
	μ_{prior}	0
	σ_{prior}	0.0126
MF BNN	N_{layers}	3
	N_{units}	[160, 80] + [2] output
	lr	0.0014
	μ_{prior}	0
	σ_{prior}	0.0526
HF BNN	N_{layers}	4
	N_{units}	[128, 160, 176] + [2] output
	lr	0.0014
	μ_{prior}	0
	σ_{prior}	0.0596
MF-BayNet (MF+HF)	N_{layers}	3
	N_{units}	[160, 80] + [2] output
	$lr(MF \rightarrow HF)$	0.001
	μ_{prior}	0
	σ_{prior}	0.0526
	N_{frz} (MF → HF)	2
MF-BayNet (LF+MF+HF)	N_{layers}	6
	N_{units}	[176, 176, 144, 176, 176] + [2] output
	$lr(LF)$	0.0016
	$lr(LF \rightarrow MF)$	0.001
	$lr(MF \rightarrow HF)$	0.001
	μ_{prior}	0
	σ_{prior}	0.0126
	N_{frz} (LF → MF)	3
	N_{frz} (MF → HF)	5

B.2. Full–configuration test case

This section outlines the architecture used for the full–configuration test case after the Bayesian hyperparameter optimization. The overall procedure closely follows what was done for the finite wing test case described in Appendix B.1, with the main difference being the larger number of layers and trainable weights required to capture the increased complexity of the full–configuration problem.

As summarized in Table B.13, the final network architecture for the full–configuration test case consists of six layers, for a total of 174,932 trainable weights. The model is first trained on the LF dataset using a batch size of 32, a learning rate of 0.0016, a maximum of 5000 epochs, and an early stopping criterion with a patience of 300. Subsequently, the model transitions to the MF dataset through TL. In this phase, three out of the six layers are kept frozen, while the remaining layers are fine–

Table B.13
Optimized hyperparameters of the MF-BayNet model for the full–configuration test case.

Model	Hyperparameter	Best value
MF-BayNet	N_{layers}	6
	N_{units}	[224, 144, 160, 112, 96] + [10] output
	$lr(LF \rightarrow MF)$	0.0081
	μ_{prior}	0
	σ_{prior}	0.0351
	$N_{frz}(LF \rightarrow MF)$	3

tuned with a batch size of 16, a learning rate of 0.0081, a maximum of 8000 epochs, and an early stopping patience of 500. Freezing part of the model ensures that knowledge gathered from the LF training is preserved, while the unfrozen layers adapt to the higher–fidelity data.

Data availability

Data will be made available on request.

References

[1] Loïc Brevault, Mathieu Balesdent, Ali Hebbal, Overview of Gaussian process based multi-fidelity techniques with variable relationship between fidelities, application to aerospace systems, Aerosp. Sci. Technol. 107 (2020) 106339.
[2] Peter Bühlmann, Sara Van De Geer, Statistics for High-Dimensional Data: Methods, Theory and Applications, Springer Science & Business Media, 2011.
[3] Russel E. Caflisch, Monte Carlo and quasi-Monte Carlo methods, Acta Numer. 7 (1998) 1–49.
[4] Martin A. Carrizales, Gaétan Dussart, Vilius Portapas, Alessandro Pontillo, Mudassir Lone, Verification of a low fidelity fast simulation framework through rans simulations, CEAS Aeronaut. J. 11 (2020) 161–176.
[5] Lucy R. Chai, Uncertainty estimation in Bayesian neural networks and links to interpretability, Master's Thesis, Massachusetts Institute of Technology, 2018.
[6] Souvik Chakraborty, Transfer learning based multi-fidelity physics informed deep neural network, J. Comput. Phys. 426 (2021) 109942.
[7] Tianqi Chen, Emily Fox, Carlos Guestrin, Stochastic gradient Hamiltonian Monte Carlo, in: International Conference on Machine Learning, PMLR, 2014, pp. 1683–1691.
[8] Adrien Crovato, Hugo S. Almeida, Gareth Vio, Gustavo H. Silva, Alex P. Prado, Carlos Breviglieri, Huseyin Guner, Pedro H. Cabral, Romain Boman, Vincent E. Terrapon, et al., Effect of levels of fidelity on steady aerodynamic and static aeroelastic computations, Aerospace 7 (4) (2020) 42.
[9] A. Da Ronch, M. Ghoreyshi, K.J. Badcock, On the generation of flight dynamics aerodynamic tables by computational fluid dynamics, Prog. Aerosp. Sci. 47 (8) (2011) 597–620.
[10] Andrea Da Ronch, Jernej Drofelnik, Michel P.C. van Rooij, Johan C. Kok, Marco Panzeri, Arne Voß, Aerodynamic and aeroelastic uncertainty quantification of NATO STO AVT-251 unmanned combat aerial vehicle, Aerosp. Sci. Technol. 91 (2019) 627–639.
[11] Behdad Davoudi, Karthikeyan Duraisamy, A hybrid blade element momentum model for flight simulation of rotary wing unmanned aerial vehicles, in: AIAA Aviation 2019 Forum, 2019, p. 2823.
[12] Stefan Depeweg, José Miguel Hernández-Lobato, Finale Doshi-Velez, Steffen Udluft, Uncertainty decomposition in Bayesian neural networks with latent variables, arXiv preprint, arXiv:1706.08495, 2017.
[13] Armen Der, Kiureghian and ove Ditlevsen. Aleatory or epistemic? Does it matter?, Struct. Saf. 31 (2) (2009) 105–112.
[14] Earl Dowell, Reduced-order modeling: a personal journey, Nonlinear Dyn. 111 (11) (2023) 9699–9720.
[15] EASA, Model-si (digital transformation - case studies for aviation safety standards - modelling and simulation), https://www.easa.europa.eu/en/research-projects/model-si-digital-transformation-case-studies-aviation-safety-standards-modelling, 2024.
[16] Thomas D. Economon, Francisco Palacios, Sean R. Copeland, Trent W. Lukaczyk, Juan J. Alonso, Su2: an open-source suite for multiphysics simulation and design, AIAA J. 54 (3) (2016) 828–846.
[17] Hamidreza Eivazi, Hadi Veisi, Mohammad Hossein Naderi, Vahid Esfahanian, Deep neural networks for nonlinear model order reduction of unsteady flows, Phys. Fluids 32 (10) (2020).
[18] M. Giselle Fernández-Godino, Review of multi-fidelity models, arXiv preprint, arXiv:1609.07196, 2016.
[19] Alexander I.J. Forrester, András Sóbester, Andy J. Keane, Multi-fidelity optimization via surrogate modelling, Proc. Royal Soc. A, Math. Phys. Eng. Sci. 463 (2088) (2007) 3251–3269.
[20] Alex Graves, Practical variational inference for neural networks, Adv. Neural Inf. Process. Syst. 24 (2011).
[21] Mengwu Guo, Andrea Manzoni, Maurice Amendt, Paolo Conti, Jan S. Hesthaven, Multi-fidelity regression using artificial neural networks: efficient approximation of parameter-dependent output quantities, Comput. Methods Appl. Mech. Eng. 389 (2022) 114378.
[22] Zhong-Hua Han, Stefan Görtz, Hierarchical Kriging model for variable-fidelity surrogate modeling, AIAA J. 50 (9) (2012) 1885–1896.
[23] Lei He, Weiqi Qian, Tun Zhao, Qing Wang, Multi-fidelity aerodynamic data fusion with a deep neural network modeling method, Entropy 22 (9) (2020) 1022.
[24] Jennifer Heeg, Overview of the aeroelastic prediction workshop, in: 51st AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, 2013, p. 783.
[25] H.B. Helmbold, Der unverwundene ellipsen flugel als tragende flanche, Jahrbuch (1942) I111–I113.
[26] Dengshan Huang, Rui Bai, Shuai Zhao, Pengfei Wen, Shengyue Wang, Shaowei Chen, Bayesian neural network based method of remaining useful life prediction and uncertainty quantification for aircraft engine, in: 2020 IEEE International Conference on Prognostics and Health Management (ICPHM), IEEE, 2020, pp. 1–8.
[27] Gabriele Immordino, Andrea Da Ronch, Marcello Righi, Steady-state transonic flowfield prediction via deep-learning framework, AIAA J. (2024) 1–17.
[28] Baptiste Kerleguer, Claire Cannamela, Josselin Garnier, A Bayesian neural network approach to multi-fidelity surrogate modeling, Int. J. Uncertain. Quantificat. 14 (1) (2024).
[29] Taehyoun Kim, Moeljo Hong, Kumar G. Bhatia, Gautam SenGupta, Aeroelastic model reduction for affordable computational fluid dynamics-based flutter analysis, AIAA J. 43 (12) (2005) 2487–2495.
[30] Toni Lassila, Andrea Manzoni, Alfio Quarteroni, Gianluigi Rozza, Model order reduction in fluid dynamics: challenges and perspectives, in: Reduced Order Methods for Modeling and Computational Reduction, 2014, pp. 235–273.
[31] Dani Levin, Kai Kruger Bastos, Earl H. Dowell, Convolution and Volterra series approach to reduced-order modeling of unsteady aerodynamic loads, AIAA J. 60 (3) (2022) 1663–1678.
[32] Zengcong Li, Shu Zhang, Hongqing Li, Kuo Tian, Zhizhong Cheng, Yan Chen, Bo Wang, On-line transfer learning for multi-fidelity data fusion with ensemble of deep neural networks, Adv. Eng. Inform. 53 (2022) 101689.
[33] Peng Liao, Wei Song, Peng Du, Hang Zhao, Multi-fidelity convolutional neural network surrogate model for aerodynamic optimization based on transfer learning, Phys. Fluids 33 (12) (2021).
[34] Quan Lin, Jiexiang Hu, Qi Zhou, Parallel multi-objective Bayesian optimization approaches based on multi-fidelity surrogate modeling, Aerosp. Sci. Technol. 143 (2023) 108725.
[35] Yixin Liu, Shishi Chen, Fenggang Wang, Fenfen Xiong, Sequential optimization using multi-level cokriging and extended expected improvement criterion, Struct. Multidiscip. Optim. 58 (2018) 1155–1173.
[36] David J. Lucia, Philip S. Beran, Walter A. Silva, Reduced-order modeling: new approaches for computational physics, Prog. Aerosp. Sci. 40 (1–2) (2004) 51–117.
[37] David J.C. MacKay, A practical Bayesian framework for backpropagation networks, Neural Comput. 4 (3) (05 1992) 448–472.
[38] Andrea Mannarino, Earl H. Dowell, Reduced-order models for computational-fluid-dynamics-based nonlinear aeroelastic problems, AIAA J. 53 (9) (2015) 2671–2685.
[39] Xuhui Meng, Hessam Babaee, George Em Karniadakis, Multi-fidelity Bayesian neural networks: algorithms and applications, J. Comput. Phys. 438 (2021) 110361.
[40] Hiroyuki Morino, Hitoshi Yamaguchi, Takayasu Kumano, Shinkyu Jeong, Shigeru Obayashi, Efficient aeroelastic analysis using unstructured cfd method and reduced-order unsteady aerodynamic model, in: 50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference 17th AIAA/ASME/AHS Adaptive Structures Conference 11th AIAA No, 2009, p. 2326.
[41] Jayant Mukhopadhaya, Brian T. Whitehead, John F. Quindlen, Juan J. Alonso, Andrew W. Cary, Multi-fidelity modeling of probabilistic aerodynamic databases for use in aerospace engineering, Int. J. Uncertain. Quantificat. 10 (5) (2020).
[42] Neal Radford, Bayesian learning via stochastic dynamics, Adv. Neural Inf. Process. Syst. 5 (1992).
[43] Bernardo Pacini, Anil Yildirim, Behdad Davoudi, Joaquim R. Martins, Karthikeyan Duraisamy, Towards efficient aerodynamic and aeroacoustic optimization for urban air mobility vehicle design, in: AIAA Aviation 2021 Forum, 2021, p. 3026.
[44] David A. Peters, David Doug Boyd, Cheng Jian He, Finite-state induced-flow model for rotors in hover and forward flight, J. Am. Helicopter Soc. 34 (4) (1989) 5–17.
[45] Vinh Pham, Maxim Tyan, Tuan Anh Nguyen, Jae-Woo Lee, Extended hierarchical Kriging method for aerodynamic model generation incorporating multiple low-fidelity datasets, Aerospace 11 (1) (2023) 6.
[46] Herbert Rakotonirina, Paul Honeine, Olivier Atteia, Antonin Van Exem, A generative deep neural network as an alternative to co-Kriging, Available at SSRN 4725658, 2024.
[47] EASA Artificial Intelligence Roadmap, Guidance for level 1 and 2 machine learning applications, 2024.
[48] Sal Rodriguez, Applied Computational Fluid Dynamics and Turbulence Modeling: Practical Tools, Tips and Techniques, Springer Nature, 2019.
[49] Francesco Romor, Marco Tezzele, Markus Mrosek, Carsten Othmer, Gianluigi Rozza, Multi-fidelity data fusion through parameter space reduction with applications to

automotive engineering, Int. J. Numer. Methods Biomed. Eng. 124 (23) (2023) 5293–5311.
[50] Alberto Savino, Dust user manual, https://public.gitlab.polimi.it/DAER/dust/-/blob/master/doc/DUST_user_manual.pdf, 2024.
[51] Antonio Segalini, P. Henrik Alfredsson, A simplified vortex model of propeller and wind-turbine wakes, J. Fluid Mech. 725 (2013) 91–116.
[52] Aman Sharma, A. Gandhi, Ajay Kumar, Exploring multi-fidelity Bayesian neural network for nuclear data evaluation, Proc. DAE Symp. Nucl. Phys. 66 (2022) 1218.
[53] Muskan Sharma, Priyanka Kushwaha, Pragati Kumari, Pushpanjali Kumari, Richa Yadav, Machine learning techniques in data fusion: a review, in: International Conference on Communication and Intelligent Systems, Springer, 2022, pp. 391–405.
[54] Yiren Shen, Jacob T. Needels, Juan J. Alonso, Vortexnet: a graph neural network-based multi-fidelity surrogate model for field predictions, in: AIAA SciTech 2025 Forum, 2025, p. 0494.
[55] Jasper Snoek, Hugo Larochelle, Ryan P. Adams, Practical Bayesian optimization of machine learning algorithms, in: F. Pereira, C.J. Burges, L. Bottou, K.Q. Weinberger (Eds.), Advances in Neural Information Processing Systems, vol. 25, Curran Associates, Inc., 2012.
[56] Sören Richard Stahlschmidt, Benjamin Ulfenborg, Jane Synnergren, Multimodal deep learning for biomedical data fusion: a review, Brief. Bioinform. 23 (2) (2022) bbab569.
[57] Jeffrey D. Taylor, Douglas F. Hunsaker, Low-fidelity method for rapid aerostructural optimisation and design-space exploration of planar wings, Aeronaut. J. 125 (1289) (2021) 1209–1230.
[58] Dustin Tran, Mike Dusenberry, Mark Van Der Wilk, Danijar Hafner, Bayesian layers: a module for neural network uncertainty, Adv. Neural Inf. Process. Syst. 32 (2019).
[59] Matteo Tugnoli, Davide Montagnani, Monica Syal, Giovanni Droandi, Alex Zanotti, Mid-fidelity approach to aerodynamic simulations of unconventional vtol aircraft configurations, Aerosp. Sci. Technol. 115 (2021) 106804.
[60] Zhicun Wang, Zhichao Zhang, D.H. Lee, P.C. Chen, Danny Liu, Marc Mignolet, Flutter analysis with structural uncertainty by using cfd-based aerodynamic rom, in: 49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, 16th AIAA/ASME/AHS Adaptive Structures Conference, 10th AIAA Non-Deterministic Approaches Conference, 9th AIAA Gossamer Spacecraft Forum, 4th AIAA Multidisciplinary Design Optimization Specialists Conference, 2008, p. 2197.
[61] Marcus Wiegand, Andriy Prots, Marcus Meyer, Robin Schmidt, Matthias Voigt, Ronald Mailach, Robust design optimization of a compressor rotor using recursive cokriging based multi-fidelity uncertainty quantification and multi-fidelity optimization, J. Turbomach. 147 (6) (2025).
[62] Daniel N. Wilke, Multifidelity surrogate models: a new data fusion perspective, arXiv preprint, arXiv:2404.14456, 2024.
[63] Bofeng Xu, Tongguang Wang, Yue Yuan, Zhenzhou Zhao, Haoming Liu, A simplified free vortex wake model of wind turbines for axial steady conditions, Appl. Sci. 8 (6) (2018) 866.
[64] Wataru Yamazaki, Dimitri J. Mavriplis, Derivative-enhanced variable fidelity surrogate modeling for aerodynamic functions, AIAA J. 51 (1) (2013) 126–137.
[65] Yi Zhang, Dapeng Zhang, Haoyu Jiang, Review of challenges and opportunities in turbulence modeling: a comparative analysis of data-driven machine learning approaches, J. Mar. Sci. Eng. 11 (7) (2023) 1440.
[66] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, Qing He, A comprehensive survey on transfer learning, Proc. IEEE 109 (1) (2020) 43–76.