



Enhancing Multi-objective Optimisation Through Machine Learning-Supported Multiphysics Simulation

Diego Botache¹ , Jens Decke¹ , Winfried Ripken² , Abhinay Dornipati³,
Franz Götz-Hahn¹ , Mohamed Aye³ , and Bernhard Sick¹ 

¹ Intelligent Embedded Systems, University of Kassel, Hessen, Germany
{diego.botache,jens.decke,franz.goetz.hahn,bsick}@uni-kassel.de

² Merantix Momentum GmbH, Berlin, Germany

³ Vehicle Systems and Fundamentals of Electrical Engineering, University of Kassel,
Hessen, Germany

{abhinay.dornipati,aye}@uni-kassel.de

<https://www.uni-kassel.de/eecs/ies>, <https://www.merantix-momentum.com/>,
<https://www.uni-kassel.de/eecs/fsg>

Abstract. This paper presents a methodological framework for training, self-optimising, and self-organising surrogate models to approximate and speed up multiobjective optimisation of technical systems based on multiphysics simulations. At the hand of two real-world datasets, we illustrate that surrogate models can be trained on relatively small amounts of data to approximate the underlying simulations accurately. Including explainable AI techniques allow for highlighting feature relevancy or dependencies and supporting the possible extension of the used datasets. One of the datasets was created for this paper and is made publicly available for the broader scientific community. Extensive experiments combine four machine learning and deep learning algorithms with an evolutionary optimisation algorithm. The performance of the combined training and optimisation pipeline is evaluated by verifying the generated Pareto-optimal results using the ground truth simulations. The results from our pipeline and a comprehensive evaluation strategy show the potential for efficiently acquiring solution candidates in multiobjective optimisation tasks by reducing the number of simulations and conserving a higher prediction accuracy, i.e., with a MAPE score under 5% for one of the presented use cases.

Keywords: Electric Motors · Multiobjective Optimisation · Surrogate-Modelling · Deep-Learning · Explainable Artificial Intelligence

1 Introduction

Multiphysics and multiscale simulations have become crucial for the computational modelling and analysis of multiple interacting physical phenomena in technical systems. These phenomena include mechanics, fluid dynamics, heat transfer, and electromagnetics for a wide variety of applications, such as aerospace engineering, biomedical engineering, and materials science, to name a few. Incorporating multiple physical phenomena into simulations is a powerful tool for engineers, enabling them to investigate various design alternatives and parameters and enhance the depth of their decision-making during design processes. Typically, this approach involves considering competing objectives simultaneously within multiobjective optimization tasks, thereby ensuring that the final design solutions strike an optimal balance across diverse performance criteria.

Acquiring optimal solutions presents a significant challenge due to the complex nature of numerical models and potential nonlinear dependencies among design parameters. Moreover, constrained solution spaces yield scarce feasible solutions and require the inclusion of advanced domain knowledge of the underlying physical problem. To address these issues, we propose using surrogate models, i.e. data-driven algorithms, as an alternative to running computationally expensive multiphysics simulations in combination with advanced optimisation techniques i.e. evolutionary algorithms. We demonstrate how surrogate models in multiphysics simulations reduce computational burdens, accelerating the design process and enabling broader exploration of design options.

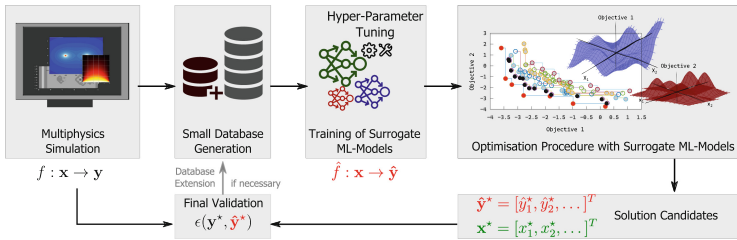


Fig. 1. Proposed strategy for training and self-optimising surrogate models using machine learning and deep learning techniques to tackle multiobjective optimisation problems in complex multiphysics simulations.

As shown in Fig. 1, our strategy can be summarised as follows: First, we identify the multiphysics system input and output space followed by a small data generation procedure. Next, we apply our pipeline for the training and evaluation of machine learning (ML) models, which are then used for the optimisation task of the input parameters of the multiphysics system. Finally, we validate the optimisation results by comparing the outputs of the ML models against the simulated values of selected solution candidates of the multiobjective optimisation problem.

The main contributions of our paper can be summarised as follows:

1. A real-world tabular dataset using multiphysics simulations, which is made publicly available
2. We streamline the efficient acquisition of solution candidates in multiobjective optimisation tasks using multiple surrogate models in combination with evolutionary algorithms
3. We validate the acquired results and derive insights about the relevance and dependencies of features in our real-world examples by using explainable AI techniques

The remainder of this article is structured as follows: Sect. 2 summarises the related work, followed by a detailed description of our combined training and optimisation pipeline in Sect. 3, Sect. 4 describes the underlying physical problems of the two real-world tabular datasets used in our approach. The results of our evaluation strategy for training and optimisation with an examination of the acquired results with explainable AI methods are presented in Sect. 5. Finally, Sect. 6 summarises the main conclusions and open issues for future work.

2 Related Work

Coupling multiphysics simulations may compromise stability, accuracy, and robustness [13,17]. Utilizing machine learning (ML) and deep learning techniques to predict efficiently and approximate multiphysics processes typically requires large amounts of data for training and feature extraction [4]. While individual approaches to substituting simulations with ML techniques have been proposed, there is an absence of a general strategy for training, hyperparameter tuning, evaluating and comparing diverse data-driven models in multiphysics simulations. Evaluations are often customised for specific applications, and it is unclear whether the approach can be generalized to other problems, especially in cases with sparse data.

2.1 Surrogate Modelling

When dealing with tabular data, employing boosted trees or ensemble strategies for meta-modelling is a reliable benchmark for execution speed, computational efficiency, and accuracy [9]. In our research, we depict the multiphysics problem in tabular format, enabling direct training of surrogates for predicting individual output values of the associated physical system. This common approach allows hyperparameter tuning of each surrogate and gives insight into the models' performance for each system output. Besides fixed tabular or parameterised physical problems, integrating deep learning techniques with numerical simulation tools has shown the potential to reduce computational burden using more complex data. These techniques can predict multiple outputs of a physical system and extract in-depth features of the data. For example, convolutional auto-encoders have been used to model scalar transport equations coupled to Navier-Stokes equations [10]. Further application strategies in fluid flow applications [1,18],

heat transfer [14,25] and electromagnetics [20] show the potential for accurate modelling of multiphysics problems but also on predicting problem specific key performance indicators. Still, most of these approaches lack the explainability and interpretability of the trained or extracted features and require high amounts of data. Physics-informed neural networks [16,23,25] have been proposed as a powerful tool to obtain solutions for multiphysics simulations without access to ground-truth data. However, these models require substantial knowledge of the underlying system, making them less accessible to non-experts. Additionally, they can be computationally expensive and time-consuming to train, needing significant resources to converge to accurate solutions.

2.2 Multiphysics Optimisation and Data Extension

In many cases, the goal consists of optimising the design parameters within these technical systems, considering multiobjective criteria. Multiobjective optimisation requires finding trade-offs between potentially conflicting objectives [24], which quickly becomes a challenging task for automated optimisation algorithms. A recent study investigated this problem through the lens of diversity and showed that their approach increases diversity without sacrificing global performance [22]. In our work, we shift the decision to consider the different objectives of the user by presenting a range of Pareto-fronts of optimised designs. We additionally encourage diversity in our optimised results by carefully choosing combinations of surrogate models and optimisation algorithms.

Limited data also poses significant challenges in multiphysics optimisation, and only a few works highlight the challenges in specific use cases. An investigation integrates topology optimisation and generative models (e.g., generative adversarial networks) in a framework allowing the exploration of new design options, thus generating many designs starting from limited previous design data [19]. In a preliminary investigation of one of the use cases, novelty and anomaly detection algorithms were used in design optimisation tasks. The study found that these algorithms are effective in exploring the design space, but they have limitations when it comes to exploitation [7]. Thus, Deep active design optimization was introduced to address this disadvantage, combining deep active learning and design optimization [6].

3 Machine Learning Supported Optimisation Strategy

In this section, we present our pipeline for applying, explaining and evaluating ML surrogate models coupled with multiobjective optimisation strategies of technical systems. It streamlines the evaluation and experimentation process and presents a general validation and performance evaluation procedure. Our approach, depicted in Fig. 2, consists of three main blocks, which will be explained separately in the following subsections.

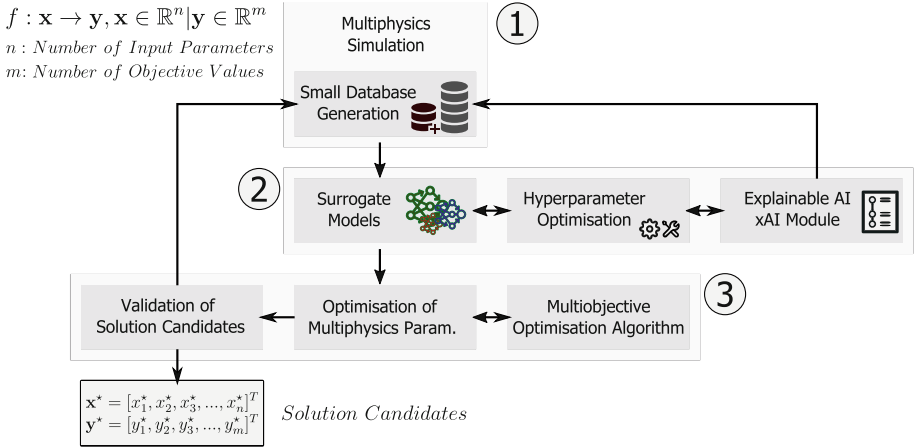


Fig. 2. Illustration of our proposed Pipeline for self-optimising surrogate models, which can be seamlessly integrated into the optimisation process of multiphysics problems and comprises three main blocks: data acquisition, surrogate model training, and multiobjective optimisation.

3.1 Data Acquisition

The first block (1), shown in Fig. 2, consists of the data acquisition module, which, in the initial phase, is a manual process. Here, a small database of simulation results from available numerical models is gathered, functioning as a starting point of the optimisation pipeline. This step is crucial for the success of the optimisation process and poses specific requirements and considerations. On the one hand, the initial database must cover many feasible input parameter combinations to train the surrogate models efficiently. On the other hand, the generation process itself relies on slow and complex numerical models, implicitly restricting the size of the database. Naturally, the quality of this first selection of data impacts the accuracy and reliability of the subsequently trained machine-learning models and the outcome of the optimisation process as a whole. In the context of the outlined use cases, the supplied tabular data comprises singular simulation instances. It can be extended to non-parametric or tabular approaches like pictures or videos, but this is not the scope of this paper. Consequently, instances characterized by incomplete simulation results or missing data will be systematically filtered from the database. In addition, depending on the type of surrogate model, some data pre-processing is essential and can be defined by the pipeline user.

3.2 Surrogate Models

The extreme gradient boosting (XGB) [2] serve as our baseline surrogate model and we train single models for the prediction of individual output values. In contrast, deep learning techniques are used for predicting multiple target values,

therefore they can consider relationships in the output space. The training of the surrogate models is shown in Fig. 2 block (2). We employ hyperparameter optimisation strategies to ensure robust training of the models. Since we assume that the databases are limited and usually distributed in a specific range of parameter values it is still possible to generate solution candidates outside that range at the end of the proposed pipeline, but this could affect the prediction performance of the models. Therefore, to identify possible unexplored areas that should be generated in the first block, Fig. 2 (1), it is crucial to understand the dependencies present in the data. To achieve this, our approach incorporates explainable artificial intelligence (xAI) techniques that provide useful insights into the data will be explained in detail in the following.

3.3 Interpretable Surrogate Modelling (xAI Module)

By analysing the results obtained from selected xAI methods, we can identify features or parameters in the data that highly influence the outputs of the physical systems. Additional methods can also provide information about the distribution of the parameter values, showing areas of the data that are not fully explored, and therefore extend the database accordingly. We used feature relevances from the XGB surrogates to understand the impact of input features on target values. Furthermore, we extended our approach using partial dependence plots [26] to identify whether the relationship between relevant features and target values is linear, non-linear, monotonic, or more complex. These plots are a powerful technique to show the marginal effects of one or two variables on the predicted outcome of a machine learning model while holding all other variables constant. Additional xAI techniques can be included in the pipeline but will not be further considered in this study.

3.4 Multiobjective Optimisation and Validation

The third block (3) of our approach leverages the pre-trained machine learning models to perform the multiobjective optimisation task, including evolutionary algorithms. The optimisation algorithm used in the following use cases is a non-dominated sorting genetic algorithm (NSGA). Genetic algorithms have proven to be a robust and reliable design optimisation method [3]. Therefore, we integrated the NSGA-2 into our optimisation pipeline [5]. The results provided by the evolutionary algorithm will be evaluated using Pareto frontiers, and the final selection of solution candidates will correspond to the points at the Pareto front. A validation step is required to validate the prediction performance of the surrogate models within the selected solution candidates. After the validation step, we propose using performance indicators [12, 15] to compare the simulated Pareto frontiers from multiple experiments with different surrogate models.

4 Experimental Design

In the following use cases, we explain the optimisation task and the corresponding target values to consider when training the surrogate models. We train a

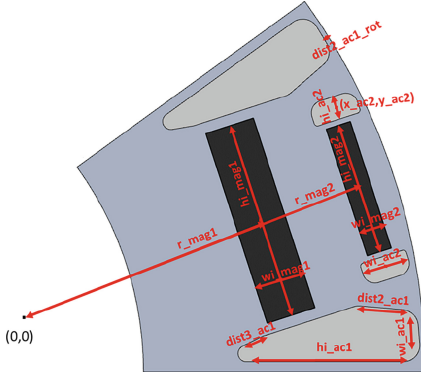


Fig. 3. Parameterised 2D rotor segment of the baseline machine with black fields representing the magnet components and grey areas for the air cavities. (Color figure online)

Table 1. Dimensions, parameters, constraints, and performance characteristics

Dimension or Parameter	Value
Stator outer \varnothing	230 [mm]
Rotor outer \varnothing	152 [mm]
Airgap Length	1 [mm]
Magnet Remanence at 120 °C	1.18 [T]
Constraints	Value
Maximum Current	636.34 [A]
Maximum Voltage	270 [V]
Performance Characteristics	Value
J_M @1000 rpm	280.5 [Nm]
J_{Φ} @1000 rpm	3.76 [kW]
J_m	2.78 [kg]

single XGB Regressor for each target value considering the mean squared error (MSE) and optimise each model’s hyperparameters independently using a combined cross-validation and Bayesian optimisation strategy. The XGB as a baseline provides valuable insights into the underlying problem and data dependencies. Further, we employ ensemble strategies combining multiple scikit-learn regressors at the decision level [11, 21]. First, we conduct a random search to identify promising regressors. Next, we form an ensemble of these regressors using a weighted average, with weights optimized via gradient-based methods. The ensemble is trained with cross-validation, and the best one is selected based on validation scores. The final chosen ensemble is trained on the entire training dataset. This approach balances exploring the hyperparameter space and exploiting the most promising models. Finally, we use two deep-learning methods, MLP and CNN, for the regression task. We use both models to estimate all target values, and we tune the hyperparameters of each model using a combined cross-validation and Bayesian optimisation strategy.

4.1 Use Case 1: Motor Dataset

The first use case consists in optimising the performance of a topology of an electric motor. The objectives of the optimisation problem correspond to maximising the mean **Torque** J_M while minimising the **Total Loss** J_{Φ} and the **Magnet Mass** J_m . Therefore, the optimisation objectives are conflicting in nature. The baseline machine for optimization is described by its dimensions, parameters, constraints, and performance characteristics in Table 1. For this work, only the rotor topology of the baseline machine—a 60-slot 10-pole internal permanent magnet synchronous machine—is parameterised with 15 geometric parameters. These parameters govern the size and position of the magnets and air cavities,

which significantly impact all the target values.¹ As shown in Fig. 3, two lateral magnets are embedded into the rotor iron lamination. In addition, two air cavities are designed in the rotor topology to guide the magnetic flux from the magnets into the air gap at the stator-rotor interface.

Efficient sampling of high-dimensional spaces is crucial to deliver enough information about the design space and, thus, train ML surrogate models. This is usually constrained by the computational resources with the numerical models, and exploring every possible combination of parameters becomes unfeasible. In this case, we apply Latin Hypercube Design (LHD) sampling, which limits the number of simulations and enables systematic and efficient exploration across the parameter space of the motor topology. The COMSOL 6.0 uncertainty quantification module generates 691 sample points based on this algorithm, i.e., motor design variations. The training and test datasets consist of 552 and 139 designs, respectively.

4.2 Use Case 2: U-Bend Dataset

The second use case, from fluid dynamics, addresses energy dissipation caused by flow deflections in technical systems. Typical applications include large-scale piping grids or cooling channels of gas turbine blades. Figure 4 depicts the U-bend, which is described by 28 design parameters consisting of six boundary points (green) and 16 curve parameters (red), offering considerable flexibility in its design and potential for optimisation. Two target values—**Pressure Loss** J_p and **Cooling Power** J_T —are evaluated using computational fluid dynamics. The Navier-Stokes equations determine the velocities of the fluid and the pressure. In addition, the energy equation is used to consider the heat conduction and the convective heat transfer between fluid and solid using a multi-physics solver. Target J_T is inversely defined, meaning a lower value signifies higher cooling power. However, these two objectives inherently conflict, as no single design can optimise both simultaneously, presenting a classic multi-objective optimization challenge.

The dataset is publicly available [8], and the inclined reader is referred to the relevant paper for a detailed description. We utilize a subset of the data, focusing solely on the parameter representation of the dataset to maintain comparability. In contrast to the Motor case, the authors used identical and independent random sampling to generate the dataset. The training set comprises 800 designs, and the test set includes 200.

¹ Please refer to <https://github.com/dbotache/> for a detailed description of the motor dataset use case.

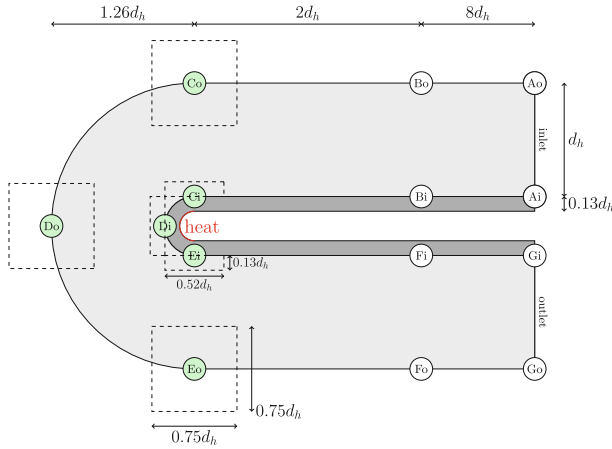


Fig. 4. Parameterised geometry with boundary points (green) and curve parameters (red). The Figure is adapted and rotated from [7] (Color figure online)

5 Results

We illustrate the versatility of our pipeline using the presented use cases and structure our results as follows: Subsect. 5.1 showcases the performance of the involved ML-surrogate models. Subsection 5.2 provides valuable insights into the regression task’s outcomes. Results obtained with an evolutionary optimisation strategy are presented in Subsect. 5.3. Moreover, in Subsect. 5.4, we validate the acquired solution candidates by comparing the predictions of the surrogate models against the numerical simulations and finally present a performance criterion for selecting the final results.

5.1 Prediction Performance of Surrogate Models

Surrogate models are evaluated using the mean absolute percentage error (MAPE) on the test dataset (20%). This metric allows evaluation and comparison of the models’ performance regardless of the target’s scale, as seen in the top half of Fig. 5. Here, MAPE values below 1.9% for the **Torque** (J_m), **Total Loss** (J_Φ) and **Magnet Mass** (J_m) indicate good prediction performance on the Motor dataset. The U-Bend dataset poses a more difficult problem, evident by the significantly higher MAPE scores using any surrogate model for predicting the target values **Pressure Loss** (J_P) and **Cooling Power** (J_T). The bottom part of Fig. 5 shows notable residual values for specific data points, particularly in predicting J_T for the U-Bend dataset. These outliers likely contribute to the elevated MAPE values observed for the same target variable, given the sensitivity of this metric to extreme values.

Another potential reason for the higher prediction error in the U-Bend case is the greater dimensionality of the input space, which has 13 dimensions more

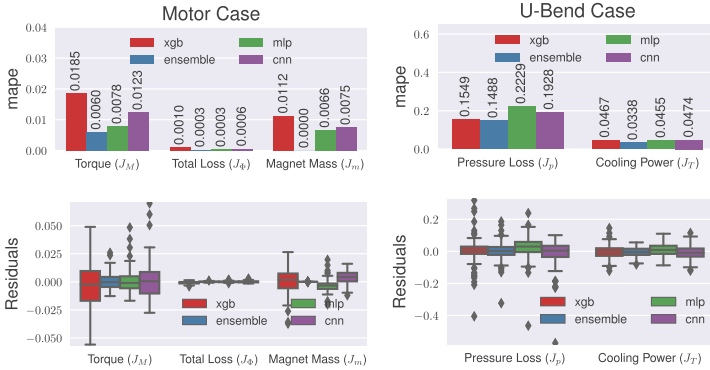


Fig. 5. Assessment of surrogate model performance using MAPE scores and residual analysis on the test sets for each use case.

than the Motor case with 15 design parameters. MSE loss for training and hyperparameter optimisation leads to a severe penalty for outliers. Consequently, the optimisation process becomes more challenging in regions featuring low target values, as they have less importance when squared. Moreover, the U-Bend case features an entirely turbulent flow, where minor parameter adjustments can trigger vortex formation and introduce discontinuities in the objective function. The cases differ fundamentally in their physics: the Maxwell equation is a first-order linear partial differential equation, whereas the Navier-Stokes equation is a second-order nonlinear system with five solution variables, potentially accounting for the variance in prediction accuracy across our two use cases.

5.2 Identifying Critical Features and Relevant Dependencies

We start considering in Fig. 6 the feature importances of each XGB model trained independently to predict a single output value in each of the presented use cases. In the Motor case, we can observe notable feature importances for the design parameters wi_mag1 , wi_mag2 and hi_mag1 concerning the output value **Magnet Mass** (J_m). It was anticipated that these parameters would strongly correlate with the magnet’s mass since it is a function of its size. Regarding the output value **Total Loss** (J_ϕ) the high importance value for the parameter wi_ac1 can be explained by the air cavities influence for the magnetic flux from the magnets to the air gap, which could affect the overall loss and efficiency. Finally, the most relevant feature for the target **Torque** (J_M) is given by the parameter hi_mag1 . This correlates with the knowledge that the magnet components influence the intensity of the magnetic flux and, therefore, the torque.

The partial dependency plots presented in Fig. 7 underscore the identified correlations using the feature importances. The curves show mainly linear dependencies for the Motor case with a high influence of the width parameters of the magnets and a low influence of the air cavity parameters on the **Magnet Mass** J_m . Conversely, air cavity-related parameters primarily influence the

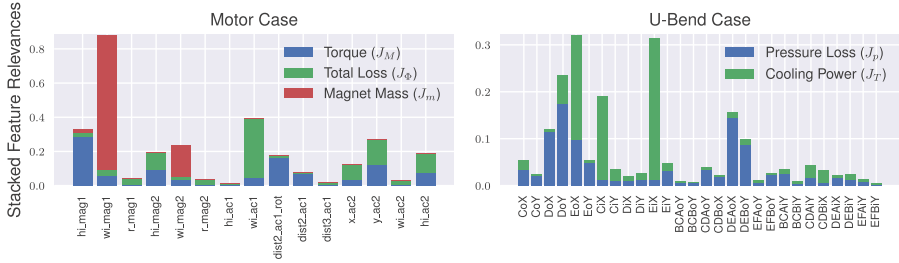


Fig. 6. Analysis of feature importance. The plots show the stacked bars of feature important values obtained from the XGBoost models for each target value and use case.

Total Loss J_{ϕ} and we can notice an extremely low influence of air cavity parameters for the target value J_m .

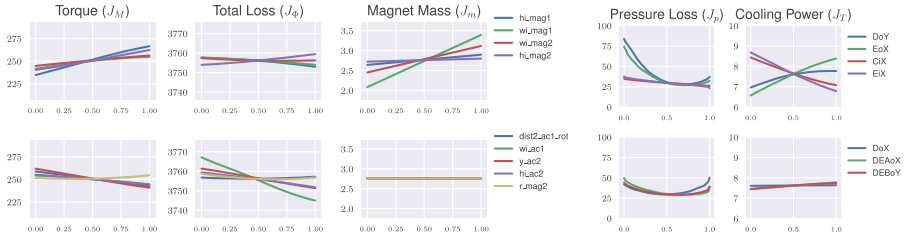


Fig. 7. Partial dependency plots for selected features and target values for the Motor and the U-Bend datasets.

In the U-Bend case, the parameter DoY is crucial for **Pressure Loss** (J_P), influencing the channel width at the outer layer's deflection point. This is expected, since altering the channel's width affects the local Reynolds number, i.e. the turbulence and vortex formation that contribute to J_P . Although EoX also impacts J_P , its effect is less pronounced. Outer curve parameters notably affect J_P , as optimal flow alignment can prevent vortex formation, whereas too small parameters may cause excessive deflection and stall. For **Cooling Power** J_T , parameters EoX, CiX, and EiX are pivotal with the highest importance values. These parameters are especially relevant to the inner layer and directly impact the heated surface area, where a thinner surface lowers heat conduction resistance. Given the high Reynolds numbers, the dominant heat transfer mechanism from the solid to the fluid is convective, making a thin solid layer beneficial. Narrowing the channel—mainly managed by outer parameters—enhances cooling performance by promoting efficient flow contact.

5.3 Evaluation of Solution Candidates in the Multiobjective Optimisation Task

The optimisation results acquired with the four surrogate models in combination with the evolutionary algorithm are presented in Fig. 8. With our presented EA optimisation strategy, we can evaluate around eight million points using the surrogate models instead of the numerical simulations. The red dots represent the data points in the original database used to train and test the surrogate models. The coloured (Motor case left) and blue (U-Bend case right) dots correspond to the predictions of each surrogate model for the final population (2,000 samples) of points present in the last iteration of the evolutionary algorithm. The colour of the points (left) represents the value of the **Magnet Mass** J_m . The Pareto frontiers with dark red points are based on the model’s prediction. In the Motor case (left), we can notice that the frontiers given by the prediction from the MLP and CNN surrogates are far from the original database. In contrast, the results with the XGB and Ensemble surrogate models deliver solution candidates which only outperform the database in regions of higher torque values ($J_M > 250$).

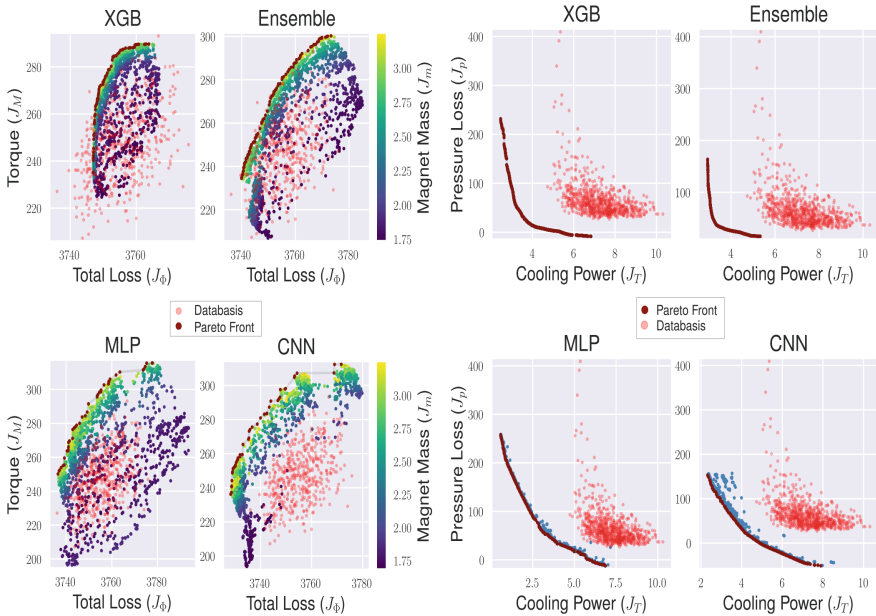


Fig. 8. Optimisation Results for the Motor Case (left) and the U-Bend dataset (right) with EA in a combination of four different surrogate models

In the U-Bend case, in Fig. 8 (right), evolutionary optimization produces Pareto frontiers superior to those in the initial database for each surrogate model, shown as dark red dots. However, a critical observation is that three

of the four models predict physically unrealistic negative pressure loss values, suggesting a lack of understanding of the physical principles involved. Recalculating the design candidates from these frontiers will further assess the models' performance.

5.4 Validation of Solution Candidates in the Multiobjective Optimisation

In the last stage of our evaluation strategy, we compare the predictions of the surrogate models, i.e. the selected solutions at the Pareto frontiers, against the true values using the original physical numerical models. The first indicator for the surrogate models' quality is reflected in Table 2 through the simulation rate. A higher simulation rate reveals the models' ability to capture parameter interrelationships and generate many plausible solutions.

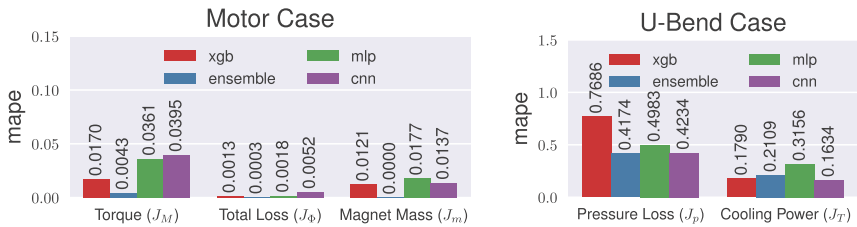


Fig. 9. MAPE of the final validation of the Motor and U-Bend Use Cases

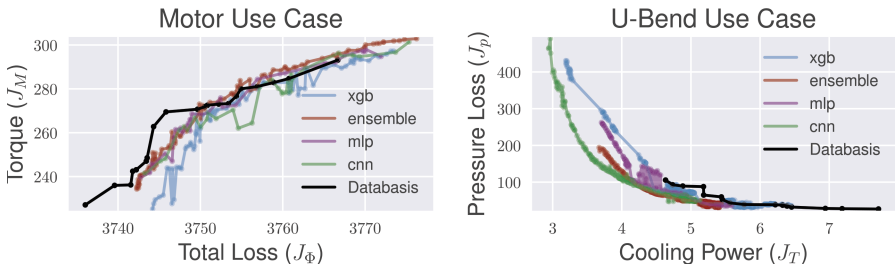


Fig. 10. Pareto frontiers of the final validation of the Motor and U-Bend Use Cases

In the Motor case, we achieve a 100% simulation rate with the ensemble model, and due to the complexity of the U-Bend case, only up to 78% of the solution candidates provided likewise with the ensemble were plausible. Next, in Fig. 9, we validate the solution candidates using the MAPE score. Higher MAPE scores for the U-Bend case reflect higher complexity in the design space and more challenging physical principles. However, the deviation of model predictions in

areas of interest in the optimisation task is mainly affected by the lower presence of samples in the training database. If the simulation rate is extremely low, an additional training cycle of the surrogate models using the simulated data points, i.e., plausible samples from the Pareto frontiers, should be considered. These additional data for training can provide additional information about regions of lower density and closer to the Pareto frontiers to the surrogate models.

Table 2. Performance indicators of validated solutions candidates concerning the original database (DB) for both use cases using Generational Distance (GD), GD-Plus (GD+) [15] and the Hypervolume Indicator (HV) [12].

	Motor Case				U-Bend Case			
	Sim. Rate	GD	GD+	HV↑	Sim. Rate	GD	GD+	HV↑
Database (DB)	–	–	–	1.753	–	–	–	0.973
DB + XGB	98.91%	0.059	0.012	1.766	69.2%	0.063	0.033	1.077
DB + Ens.	100%	0.048	0.018	1.795	78.3%	0.062	0.009	1.112
DB + MLP	80.43%	0.038	0.010	1.774	20.9%	0.111	0.078	1.091
DB + CNN	91.81%	0.041	0.012	1.770	53.6%	0.128	0.064	1.157

The numerical validation and final results are shown in Fig. 10. The optimisation task was successful in both cases. In the Motor case, we appreciate better solutions candidates at higher Torque values as appreciated in the predictions in Subsect. 5.3. Particularly in the U-Bend case, the simulation results are closer to the database and reflect higher deviations from the predicted values in Fig. 8 on the right. However, the obtained solutions candidates outperformed the original database Pareto frontier in all regions, concerning the two objective values J_T and J_P . These results emphasise the versatility of our strategy and set the focus on the quality of the final solution candidates, which do not depend on the prediction accuracy of the surrogate models.

In our experiments, four surrogate strategies were compared using Pareto frontiers, but combining more surrogates and optimization strategies could make qualitative comparisons unmanageable. Using performance indicators as shown in Table 2 allows a quantitative comparison of the validated Pareto frontiers and the filtering of only the most significant results. In the Motor case, we see higher performance of the results with the ensemble strategy with a Hypervolume (HV) of 1.795 and in the U-Bend case, we observe the best results with the CNN surrogate and an HV of 1.157. The values of the HV are calculated using normalized target values and reference points of the maximal values present in the original databases.

6 Conclusion

Our investigation underscores the efficacy of employing diverse machine learning surrogate models across heterogeneous engineering use cases. In particular,

our analysis reveals that although the Motor dataset exhibits commendable performance with MAPE values below 1.9%, the U-Bend dataset poses greater challenges, evident in higher MAPE values. At the hand of a feature importance analysis and partial dependency plots, we determined key parameters significantly influencing output values within both datasets. Validating the solution candidates against validated values using numerical simulations confirms the efficacy of the surrogate models. Despite higher complexity in the U-Bend case with some deviations attributed to the scarcity of training data in certain regions, our approach achieves efficient results. It showcases the versatility of our pipeline in addressing diverse optimization tasks and the evaluation strategy can be applied in a general way in future work to multiple use cases. Future investigations will focus on iteratively refining the optimization cycle and expanding the dataset to enhance model performance, the robustness of our methodology and its applicability in addressing complex engineering problems.

Acknowledgement. We kindly acknowledge funding by the Federal Ministry for Economic Affairs and Climate Action (BMWK) within the project "KITE: KI-basierte Topologieoptimierung elektrischer Maschinen" (#19121034B). We extend our special thanks to Thomas Menne³ for simulating solution candidates and providing feedback on the Motor case.

References

1. Bhatnagar, S., Afshar, Y., Pan, S., et al.: Prediction of aerodynamic flow fields using convolutional neural networks. *Comput. Mech.* (2019)
2. Chen, T., Guestrin, C.: XGBoost: a scalable tree boosting system. In: International Conference on Knowledge Discovery and Data Mining (SIGKDD). ACM, San Francisco California USA (2016)
3. Choi, M., Choi, G., Bramerdorfer, G., Marth, E.: Systematic development of a multi-objective design optimization process based on a surrogate-assisted evolutionary algorithm for electric machine applications. *Energies* **16**, 392 (2023)
4. De Bézenac, E., Pajot, A., Gallinari, P.: Deep learning for physical processes: incorporating prior scientific knowledge. *J. Stat. Mech. Theory Exper.* (2019)
5. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multi objective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* (2002)
6. Decke, J., Gruhl, C., Rauch, L., Sick, B.: DADO – Low-cost query strategies for deep active design optimization. In: 2023 International Conference on Machine Learning and Applications (ICMLA), pp. 1611–1618. IEEE (2023)
7. Decke, J., Schmeißing, J., Botache, D., Bieshaar, M., Sick, B., Gruhl, C.: NDNET: a unified framework for anomaly and novelty detection. In: Schulz, M., Trinitis, C., Papadopoulou, N., Pionteck, T. (eds.) *Architecture of Computing Systems: 35th International Conference, ARCS 2022, Heilbronn, Germany, September 13–15, 2022, Proceedings*, pp. 197–210. Springer International Publishing, Cham (2022). https://doi.org/10.1007/978-3-031-21867-5_13
8. Decke, J., Wünsch, O., Sick, B.: Dataset of a parameterized U-bend flow for deep learning applications. *Data Brief* 109477 (2023)
9. Derouiche, K., Garois, S., Champaney, V., et al.: Data-driven modeling for multiphysics parametrized problems-application to induction hardening process. *Metals* (2021)

10. El Haber, G., Viquerat, J., Larcher, A., et al.: Deep learning model to assist multiphysics conjugate problems. *Phys. Fluids* (2022)
11. Feurer, M., Klein, A., Eggenberger, K., et al.: Efficient and robust automated machine learning. In: *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 2962–2970 (2015)
12. Fonseca, C., Paquete, L., Lopez-Ibanez, M.: An improved dimension-sweep algorithm for the hypervolume indicator. In: *International Conference on Evolutionary Computation*, IEEE, Vancouver, BC, Canada (2006)
13. Groen, D., Zasada, S.J., Coveney, P.V.: Survey of multiscale and multiphysics applications and communities. *Comput. Sci. Eng.* **16**, 34–43 (2014)
14. Hachem, E., Ghraieb, H., Viquerat, J., et al.: Deep reinforcement learning for the control of conjugate heat transfer. *J. Comput. Phys.*, 110317 (2021)
15. Ishibuchi, H., Masuda, H., Tanigaki, Y., Nojima, Y.: Modified distance calculation in generational distance and inverted generational distance. In: Gaspar-Cunha, A., Henggeler Antunes, C., Coello, C.C. (eds.) *Evolutionary Multi-Criterion Optimization: 8th International Conference, EMO 2015, Guimarães, Portugal, March 29 –April 1, 2015. Proceedings, Part II*, pp. 110–125. Springer International Publishing, Cham (2015). https://doi.org/10.1007/978-3-319-15892-1_8
16. Karniadakis, G.E., Kevrekidis, I.G., Lu, L., et al.: Physics-informed machine learning. *Nature Rev. Phys.* **3**, 422–440 (2021)
17. Keyes, D.E., McInnes, L.C., Woodward, C., et al.: Multiphysics simulations: challenges and opportunities. *Int. J. High Performance Comput. Appl.* pp. 4–83 (2013)
18. Morimoto, M., Fukami, K., Zhang, K., et al.: Convolutional neural networks for fluid flow analysis: toward effective metamodeling and low dimensionalization. *Theor. Comput. Fluid Dyn.* (2021)
19. Oh, S., Jung, Y., Kim, S., et al.: Deep generative design: integration of topology optimization and generative models. *J. Mech. Design* (2019)
20. Parekh, V., Flore, D., Schöps, S.: Variational autoencoder-based metamodeling for multi-objective topology optimization of electrical machines. *IEEE Trans. Magnetics*, 1–4 (2022)
21. Pedregosa, F., Varoquaux, G., Gramfort, A., et al.: Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.*, 2825–2830 (2011)
22. Pierrot, T., Richard, G., Beguir, K., Cully, A.: Multi-objective quality diversity optimization. In: *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 139–147 (2022)
23. Raissi, M., Perdikaris, P., Karniadakis, G.E.: Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.*, 686–707 (2019)
24. Sener, O., Koltun, V.: Multi-task learning as multi-objective optimization. In: *Advances in Neural Information Processing Systems (NeurIPS)* (2018)
25. Sharma, P., Chung, W.T., Akoush, B., Ihme, M.: A review of physics-informed machine learning in fluid mechanics. *Energies*, 2343 (2023)
26. Zhao, Q., Hastie, T.: Causal interpretations of black-box models. *J. Bus. Econ. Stat.* (2021)