Chinese Journal of Aeronautics, (2023), 36(2): 213–228

Chinese Society of Aeronautics and Astronautics
& Beihang University

Chinese Journal of Aeronautics

cja@buaa.edu.cn
www.sciencedirect.com

Surrogate role of machine learning in motor-drive optimization for more-electric aircraft applications

Yuan GAO [a], Benjamin CHEONG [b], Serhiy BOZHKO [a], Pat WHEELER [a], Chris GERADA [a], Tao YANG [a,*]

[a] *Power Electronics, Machines and Control Group, University of Nottingham, Nottingham NG7 2RD, UK*
[b] *School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798, Singapore*

Received 17 January 2022; revised 5 April 2022; accepted 10 June 2022
Available online 20 August 2022

KEYWORDS

Artificial Neural Network (ANN);
Design and Optimization;
Machine Learning (ML);
More-Electric Aircraft (MEA);
Motor drive;
Permanent Magnet Synchronous Motor (PMSM);
Search Algorithm;
Surrogate Algorithm

Abstract Motor drives form an essential part of the electric compressors, pumps, braking and actuation systems in the More-Electric Aircraft (MEA). In this paper, the application of Machine Learning (ML) in motor-drive design and optimization process is investigated. The general idea of using ML is to train surrogate models for the optimization. This training process is based on sample data collected from detailed simulation or experiment of motor drives. However, the Surrogate Role (SR) of ML may vary for different applications. This paper first introduces the principles of ML and then proposes two SRs (direct mapping approach and correction approach) of the ML in a motor-drive optimization process. Two different cases are given for the method comparison and validation of ML SRs. The first case is using the sample data from experiments to train the ML surrogate models. For the second case, the joint-simulation data is utilized for a multi-objective motor-drive optimization problem. It is found that both surrogate roles of ML can provide a good mapping model for the cases and in the second case, three feasible design schemes of ML are proposed and validated for the two SRs. Regarding the time consumption in optimizaiton, the proposed ML models can give one motor-drive design point up to 0.044 s while it takes more than 1.5 mins for the used simulation-based models.

1. Introduction

With the rapid development of data science and computing technology, the interests and application of Artificial Intelligence (AI) have been growing recently [1,2]. In most cases, AI relies on Machine Learning (ML) algorithms to perform desired specific AI tasks after a model training/test/validation process. Based on the collected data (from the original system) and the dedicated training, ML algorithms build mathematic mapping models which can make predictions effectively without the original system being explicitly executed/pro-

* Corresponding author.
E-mail address: tao.yang@nottingham.ac.uk (T. YANG).
Peer review under responsibility of Editorial Committee of CJA.

Production and hosting by Elsevier

https://doi.org/10.1016/j.cja.2022.08.011
1000-9361 © 2022 Chinese Society of Aeronautics and Astronautics. Production and hosting by Elsevier Ltd.

grammed [3]. For a motor drive design, this function of ML paves the way for effective estimation of system performance, which traditionally replies on the Finite-Element Analysis (FEA) of electromagnetic, mechanical, and thermal performance [4]. However, the FEA model simulation generally comes with heavy computation burden for computers and very time consuming. This is the main reason why ML algorithms have been explored in the design and optimization of electric motor systems to achieve good design but a much faster way [4].

Permanent Magnet Synchronous Motors (PMSMs) have been widely used in electrical vehicles, fans, drives, and compressors [5–7] due to its high power density and reliability. PMSM is also commonly selected for electromechanical actuators, electric compressors and braking systems in the More-Electric Aircraft (MEA) [8,9]. Fig. 1 shows the diagram of studied PMSM drive system, and an actuator example onboard MEA [10,11]. For flight control systems, the electromechanical actuator configuration is chosen to guarantee the maximum reliability and minimum mass [12]. To design the PMSM based drive systems, FEA and detailed simulation have become the main approach. However, because of its multi-physics complexity, designing an optimal motor drive using FEA and detailed simulation may take several hours or days of computations. To this end, nonlinear surrogate models, built by ML algorithms, are proposed to speed up the FEA-based motor-drive design and optimization process. Two different Surrogate Roles (SRs) of ML are introduced here to demonstrate alternative ways for the motor-drive design and optimization problem.

ML algorithms can be categorized into three main groups: supervised learning, unsupervised learning, and reinforcement learning [2,13,14]. According to the statistics in Ref.[14], most ML algorithms for the power electronic applications use the supervised learning of ML which accounts for 91% (444 journal papers from 1990 to 2020 were identified). More specifically, the most popular implementation of supervised learning is based on Artificial Neural Networks (ANNs), which share some principles with biological neurons in brains [13,15]. The main three classes of ANN are feedforward, convolutional, and recurrent networks [2,13]. Among those, the theory of feedforward ANN is simplest and it is a universal function approximator with strong generalization capability [16]. For this reason, this study will mainly investigate feedforward ANN in motor-drive design and optimization. It is found that, after training, the ANNs can smoothly provide the desired performance estimation (or dedicated correction factors) with negligible errors for the motor-drive design and optimization.

The applications of ML for the system-level design and optimization of a motor-drive are rarely reported. Most of literature only uses ML for design and optimization of individual electrical component, for example an electrical motor or an LC filter. In Ref.[4], the authors reviewed the recent developments of machine learning applications on the design optimization of passive components within power electronic systems. Though most of commonly used ML algorithms have been referred including ANN, Support Vector Machine (SVM), Extreme Learning Machine (ELM), convolutional neural network etc., the application of ANN in the system-level optimal design are not discussed. Ref.[17] reviewed AI/ML applications in electric machine drives: advances and trends. It is found that, nearly all the reviewed papers were using AI/ML techniques for the control and observers of motor drives, rather than the mass/loss design and optimization.

Ref. [18] used an ELM to train the measured experiment data of motor indirect flux linkage, and the predictions of this ELM showed a good match with both motor FEA and the experimental raw data. In Ref.[19], ANN has been utilized to build surrogate models for machine optimization, the model has the best accuracy versus training time tradeoff among several surrogate candidates but, most of the ANN technical details (e.g. sample collection, ANN training and test) are not presented. An adaptive network-based fuzzy inference system was presented in Ref.[20], this system acted as a surrogate for the motor FEA optimization. This approach may help accelerate the original FEA-based optimization process but, the computation time for the proposed method was still very long (around 24 h) in the case study.

To demonstrate the time-saving priority of ANN, the authors in Ref.[21] proposed using ANN to provide dedicated correction factors for the motor multi-objective optimization. After the training with 64 data samples, the ANN-analysis-hybrid model can predict the motor performance much faster than the traditional 2D FEA method (for one design point, 0.04 s compared to 1 min). However, the ANN-aided correction approach was used for motor optimization only, no system-level study was performed. Another interesting ANN application example for a subsystem can be found in Ref.[13], and reports fast and accurate ANNs to address the inductor design and optimization problems. There are four workflows introduced: ANN-based inductor model, ANN-based subcomponent model, classical model with ANN corrections, and ANN-based inductor optimization. Differently, inspired by the search (optimization) algorithms, this paper explicitly focuses on the input–output relation from the motor-drive optimization problem and proposes two surrogate roles of machine learning. For the SR method validation, experiment, full-analytical models, and FEA-joint simulation models are all considered in the two validation cases.

The main contributions of this paper can be summarized as: (A) Two kinds of ML SRs are proposed to provide general ideas for the ML applications on motor-drive optimization, which are for the system-level design and optimization, instead of the sub-system level. (B) The proposed SRs are both inde-

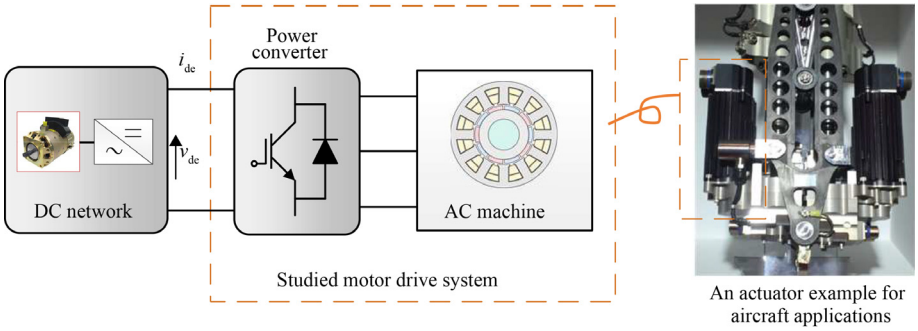


Fig. 1 System diagram.

pendent to the motor topologies and sizing models due to the constraint-free data-driven function of ML. Therefore, the potential applications should not be constrained only to MEA. (C) Using both SRs, two different cases are given with technical details to comprehensively introduce how to use experimental or simulation target data in the surrogate models of ML. In these cases, ML based surrogate models can all efficiently give predictions to ensure the high accuracy of performance estimation in the whole design space. (D) Comparisons with other surrogate methods (including function fitting, parameter optimization) are discussed in this paper.

This paper is organized as follows: Section 2 will introduce a simple category of commonly used algorithms and the principles of ANN; after that, the proposed surrogate roles of ANN in optimization will be illustrated and their differences with conventional optimization problems will be discussed in Section 3; Section 4 and Section 5 will give two cases to validate the proposed two SRs of ANN, where the first case demonstrates the priority of SRs over other surrogate methods and the second case further compares the performance of two SRs in the same optimization problem. After comparisons in Section 5, it is found that the second SR is a little superior than the first in this specific optimization case. Finally, this paper is concluded in Section 6.

2. Principle of machine learning

For an optimization problem, no matter which discipline, search algorithms are widely used to find the optimal solution. Commonly used search algorithms include Genetic Algorithm (GA) [22–24], particle swarm [25,26], tabu search [26,27], etc. Recently, on the other hand, an emerging trend is using ML algorithms to optimize motor system design [4,21]. The main motivation is that ML algorithms can effectively learn and then accurately predict the nonlinear and multi-modal characteristics in electromagnetic devices, which can be used to speed up the general optimization process. Before introducing the surrogate roles of ML, it is worth discussing the fundamental and functional differences between search and ML algorithms.

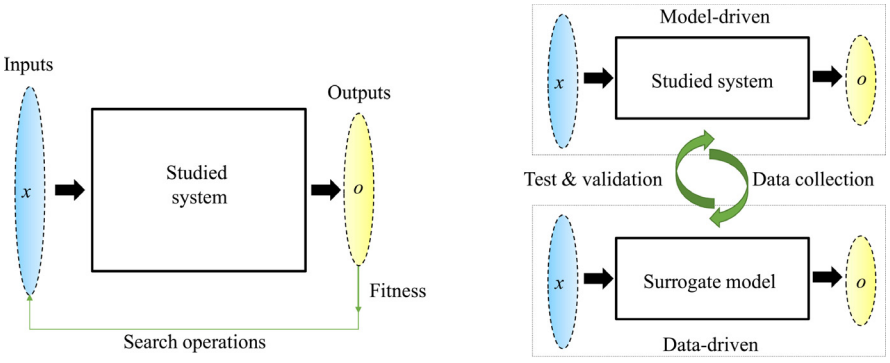
2.1. Search algorithm and surrogate algorithm

In Ref.[22], a novel category strategy for commonly used algorithms was proposed with two categories, i.e. search algorithm [24,26] and surrogate algorithm [15,28–30]. The general idea of these two categories are depicted in Fig. 2. Both algorithms are describing the relations of x-to-o but their logic directions are different: for search algorithms, the direction is from outputs to inputs because the fitness of outputs determine the input of next generation; in contrast, the logic direction of surrogate algorithm is "inside" the system because data in the algorithm operation is always flowing from inputs to outputs (x-to-o).

As shown in Fig. 2(a), the search (optimization) algorithm treats a studied system as a black box (no matter linear or nonlinear). This kind of algorithm is to calculate the fitness of all design candidates in the current generation and then determine new designs in the next generation according to the specific algorithm theories and search operations. Obviously, the search operations would differ for various search algorithms; for example, GA has three common operators: selection, crossover, and mutation [24,31]. Apart from the traditional velocity and position update operations, additional crossover operators can be introduced into the particle swarm algorithm for breeding high-quality exemplars in the particle evolution [32].

The other category, surrogate algorithm, is to build a mathematic model based on a set of samples (known as training data). As presented by Fig. 2(b), there are two parts in the algorithm operation: model-driven part and data-driven part. The first part is used to generate required data for the data-driven stage. In the first part, the outputs are generated throughout the studied system/model when given a combination of inputs. Based on that, the training data (inputs and outputs from the studied system or models) for the surrogate model can be collected as a set of input–output samples. Then, the second part is an off-line training process by only using the collected training data. The training parameters from various surrogate algorithms are different but, after training, the aim of surrogate model would be all replacing the original systems to smoothly and effectively perform specific tasks, without running the original system model. Therefore, different from the first part, the second part is a data-driven model-free process without using explicit instructions, relying on patterns and inference instead [3].

Most of ML algorithms are in the scope of surrogate algorithms, however, as mentioned before, this study mainly considers the applications of supervised learning, feedforward ANN more specifically. This ANN has been applied to various electrical engineering topics, from the optimal design for reliability of power electronic converters [15], to the diagnose of open-switch faults [33]. Noting that the SRs of ML presented here are independent with the ML techniques thus, the general methodology of using surrogate roles would be the same for other kinds of learning algorithms, such as convolutional neural network, ELM, etc.



(a) Search algorithm (b) Surrogate algorithm

Fig. 2 Search and surrogate algorithm category.

Based on the concept of surrogate algorithm, this paper further develops how to use surrogate/ML algorithms in the design and optimization of a motor-drive system, i.e. surrogate roles of ML in motor-drive system design. Generally, there are two roles of ML, one is providing correction factors for analytical models while the other role just leaves analytical models, directly mapping from design variables to system performance, see Section 3 for more details. Both surrogate roles are based on the theory of feedforward ANN, as discussed in the next subsection.

2.2. Fundamentals of feedforward ANN

Fig. 3 shows the schematic diagrams of a feedforward ANN and the mathematic operation in a certain neuron. Within an ANN, a neuron is a mathematical function that model the functioning of a biological neuron [13,16]. As shown in Fig. 3(a), a basic feedforward ANN consists of an input layer (x), one or more hidden layers (h), and an output layer (o), weights and biases are omitted for simplicity in Fig. 3(a). Each layer has one or several artificial neurons which receive and process signals from predecessor neurons, after that neurons deliver the processed data to the next layer. If only one hidden layer, the structure is named as shallow ANN. If ANNs have several hidden layers, they are usually qualified as deep learning ANNs.

In a Layer l (hidden layer or output layer), to calculate the output of a certain neuron n_i^l, the outputs of all the neurons p_j^{l-1} in Layer $l-1 (j = 1, 2, .., N_{l-1}, N_{l-1}$ denotes the neuron number of Layer $l-1$) are multiplied with given weights ω_{ij}^l and then the bias b_i^l is added. The result is further processed through an activation (propagation) function f_σ that usually takes the form of a sigmoid function, $f_\sigma(A) = 1/(1+e^{-A})$, to generate the output p_i^l, as shown in Fig. 3(b). This output then becomes one of the inputs for the next layer, $l+1$, and the same procedure is repeated to calculate the output of other neurons in layer l.

In Layer 1 (input layer), p_i^1 takes the form of inputs through the neuron n_i^1. On the other side, Layer L(output layer) typically uses the linear activation function to integrate signals of Layer $L-1$ for the desired output data p_i^L. In summary, the complete data flow of ANN can be described as follows:

Layer 1 (input layer):

$$p_i^1 = x_i, i = 1, 2, \ldots, N_1 \tag{1}$$

where x_i denotes the inputs of ANN.

Layers $l = 2, 3, \ldots, L-1$(hidden layers):

$$p_i^l = f_\sigma\left(\sum_{j=1}^{N_{l-1}} \omega_{ij}^l p_j^{l-1} + b_i^l\right), i = 1, 2, \ldots, N_l \tag{2}$$

Layer L(output layer):

$$o_i = \omega_i^L p_i^L, i = 1, 2, \ldots, N_L \tag{3}$$

where o_i represents the outputs of ANN. Therefore, every neuron is a node which determines the input–output relationship of the signal. ANN can be trained to be a nonlinear model using a proper data set. Such general nonlinear model can approximate any given input–output functions with arbitrary precision [16]. This strong generalization capability is used here to train surrogate models for fast and accurate mapping and prediction in the motor-drive design and optimization process.

The number of neurons in the input and output layers are dependent on samples and they are specific to the studied problem. The neuron numbers in the hidden layers are arbitrary and

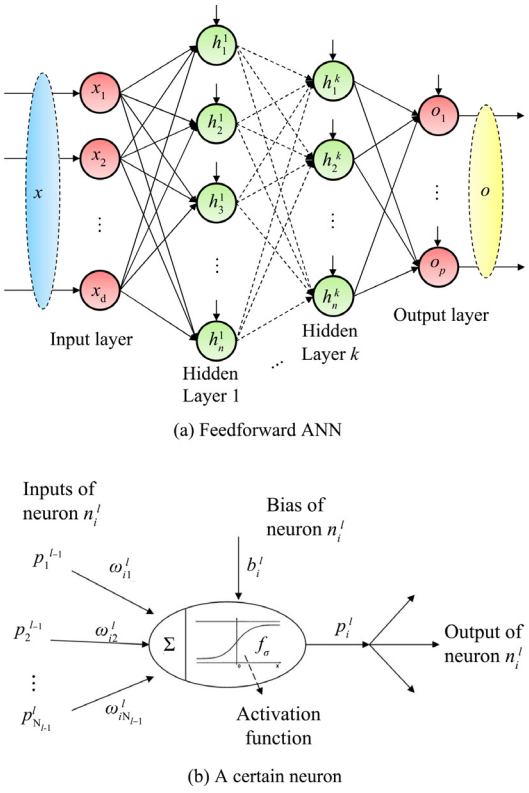


Fig. 3 Diagram of a feedforward ANN and a neuron in ANN.

can be set by an ANN algorithm developer. Therefore, neuron number selection for different ANN structures is flexible and this has been a persistent and hot topic in multi-discipline studies [34–36]. Obviously, the more hidden layers and neurons, the better learning capabilities of ANNs, which allows processing a more complex training dataset. However, over-fitting is prone to happen with ANNs featuring many artificial layers/neurons compared to the size and/or complexity of the dataset [2,13]. After a certain number of training epochs, the ANN may try to fit the noise of the training subset but, at the same time, resulted in increased error with the validation data set. Therefore, in this study, the ANNs with one-hidden-layer are used for the motor-drive system design.

3. Surrogate role of machine learning

In this section, two different ML surrogate roles for motor drive applications will be first introduced: one for the direct mapping from Design Variables (DVs) to the motor-drive performance while the other for mapping DVs to correction factors (for analytical motor-drive models). After that, the surrogate models will be used for optimized design.

3.1. Two different surrogate roles of ANN

Two different surrogate roles (SR_1 and SR_2) of ANN are studied in this paper, as depicted in Fig. 4. Both roles are describing how to perform the ML models in the motor-drive design and optimization. Inputs of SR_1 and SR_2 are the same, i.e. the DVs of motor-drive system. However, the outputs of ANN in different SRs are different.

ANN in SR_1 maps from variables to the target motor-drive performance data, as shown in Fig. 4(a). In this case, the

inputs of ANN are the design variables and the outputs are the motor-drive performance indices, which are usually the optimization objectives. Noting that, the target data can be either experimental or detailed simulation data, which should be accurate and fully trusted in the studied problem. Based on the collection of input–output samples, the ANN of SR_1 can be trained offline, i.e. updating the ANN weights ω and bias b according to the errors between raw outputs and current predictions. Then, the well-trained ANN can be used to perform the surrogate task in optimization: fast and accurately providing the performance index values when given a combination vector of all the input DVs.

Different from SR_1, ANN in SR_2 takes the analytical models of motor drives into account. Analytical models are based on analytical equations and feature closed-form analytical solutions which are simple and fast compared with FEA models. However, they are usually prone to lose the accuracy in an optimization problem due to their assumptions and simplifications, especially for a large design space [21,37]. As shown in Fig. 4(b), ANN in SR_2 is used to correct the analytical model estimations to follow the target data. This ANN would map from DVs to the dedicated correction factors, which are the ratios of target data on the corresponding analytical estimations. The same with SR_1, after the raw-data collection, the ANN of SR_2 can be trained offline; however, regarding the follow-up surrogate task of SR_2, the ANN-analysis-hybrid model should be used to replace the original models and efficiently provide the accurate prediction for a DV combination vector.

Therefore, two SRs represent different general functions of ANN models in motor-drive optimization but, they use the same offline training process after data collection. The ANN offline training usually divided all samples into three data sets: training data, validation data and test data. Training data is used to directly modify/update the weight and bias values in a Neural Network (NN); validation data is used for validating the stopping condition and the ranking of net candidates; testing data is used to obtain unbiased estimates of non-training data [38]. After training, the designed surrogate models can help perform the design and optimization problem very quickly. The next subsection will briefly discuss what the function of surrogate model is in optimization and the main differences with the traditional optimization approach.

3.2. Surrogate model in optimization

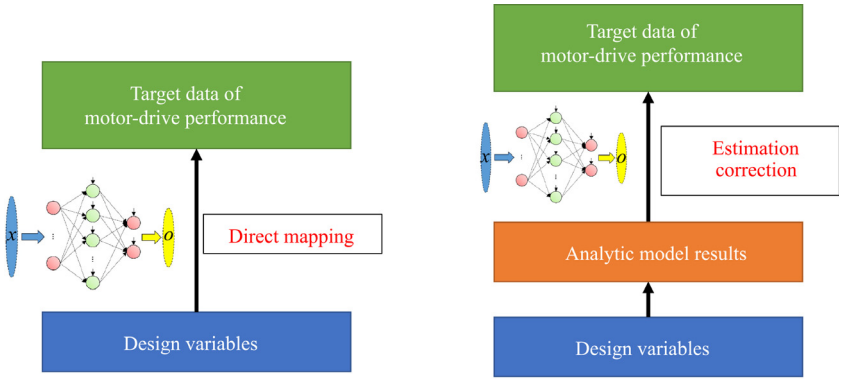
The design and optimization of motor-drive systems can generally be split into two parts: modelling and optimization. Modelling is to establish the relationship between the system design parameters and the system performance (for example power losses, mass, thermal). Based on the design models, search algorithms are generally utilized to search for a trade-off solution for the optimization problem, i.e. finding the best design point in a variable space subject to the constraints. In Ref.[39], this design and optimization process is named as optimization-based design, which consists of three parts (from outside to inside as shown in Fig. 5): optimization engine, objective function and detailed analysis.

Detailed analysis is a modelling process, which provides an understanding of the real-world system and models it to take in a set of input design parameters and, correspondingly, calculate and output values of interests. Based on these interests, objective function generates the objective values for optimization which are direct indices for the fitness calculation and iteration. Optimization engine refers to the function of optimization algorithms, it can provide insight into possible designs that are optimal as defined with a set of objective values. The optimization engine is depicted at the outer level because its computational algorithm is operated on the design variables and the objective function during all the optimization generations.

In this paper, ML algorithms will be applied to the optimization-based design problems for the aircraft motor drive systems (see Section 5). Based on the proposed SR_1 and SR_2 of ANN, the optimization-based design process (ML part) will not need to reply on the original time-consuming models (for example FEA) and search algorithms, but using the trained surrogate models and the exhaustive algorithm instead.

As shown in Fig. 5, first, the optimization engine would be just global exhaustive algorithms because, based on the ML surrogate models, the objective function and the detailed analysis only involve the mathematic models. These models can provide the desired outputs extremely fast when given an input combination. Obviously, after training, there is no need for extra complex and time-consuming derivations in this optimization process. This is due to the fact the original relations between inputs and outputs have been learned by the ML models, based on the collected and processed sample data.

Therefore, the main motivation for using ML is due to the ML algorithms representing some advantages over search algorithms [21,22]. After an ML model is trained, it can be used directly in any future study of the same optimization problem, instead of running the detailed system model all over again with search algorithms. Apart from the efficiency, other advantages of ML algorithms include: the trained ML models with



(a) SR_1, direct mapping approach (b) SR_2, correct approach

Fig. 4 Two kinds of Surrogate Role (SR).

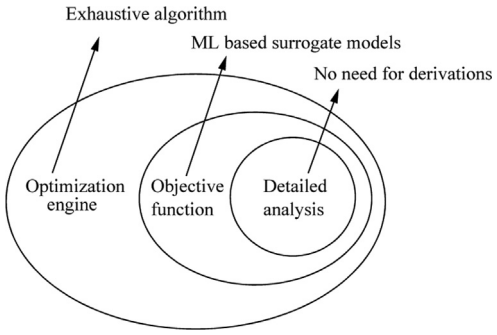


Fig. 5 Optimization-based design process using ML.

exhaustive algorithms have no risk of getting stuck in local optimum; In addition, within the original design space, the ML models can be also feasible for the evaluation and optimization in any sub design space, also fast and accurate. If using the FEA-based optimization with search algorithms, the FEA model needs to be run all over again for these new optimization problems because, in a subspace, the previous searched samples can be very little thus cannot be utilized to give the optimal design.

4. Case I

In this section, the first case will be given using the experimental data of PMSM losses under no-load condition in a motor-drive rig. The losses are divided into two parts: iron loss and bench mechanical loss, and the latter comprises bearing friction loss and windage loss. In practice, the analytical models have a few coefficients to define the total loss value, then this loss estimation is usually corrected by another correction factor to match the experimental results. All these ratios are usually determined by trial-and-error thus, there may be significant errors between analytical estimations and the experimental results.

Therefore, in this section, SR_1 and SR_2 are used to build ANN models to precisely predict the PMSM total losses with regards to the motor mechanical speed (ω_m). In addition, a search algorithm (Particle Swarm Optimization, PSO) is used as a comparative method in this section, which is to find the optimal coefficients of analytical models matching the experimental data. In the following contents, the analytical models and the motor rig will be first introduced; after that, the mapping results from PSO and ML algorithms will be compared.

4.1. Analytical models

PMSM usually shows three significant parts of their power losses: winding copper losses, iron losses within the magnetic circuit, and eddy current losses of Permanent Magnet (PM). The copper loss is proportional to the input current and winding resistance; in contrast, the other two are related to the machine fundamental frequency. As the PM (and rotor) eddy current losses are only problematic for high-speed machines, in this section, only iron loss is considered as the fundamental in the motor.

Under no-load conditions, the mechanical loss includes bearing friction loss and windage loss. A precise modelling of bearing friction and windage losses is complicated, and these losses are typically determined experimentally, using a dummy rotor without magnets. Therefore, mechanical losses are given here using empirical coefficients of friction and windage losses. The analytical models of iron loss and mechanical loss will be given separately, as below. The basic analytical model of sizing PMSM in from Ref.[40].

4.1.1. Iron loss

Under no-load condition, iron losses contribute the largest part of the total motor loss. A modified Steinmetz equation is commonly used analytically to calculate the iron losses [6,21]. The Steinmetz equation is with two different terms, hysteresis and eddy current loss. The specific iron loss ratio W_{ir}(W/kg) for a certain material is given as:

$$W_{ir}\left(f,\widehat{B}\right) = K_h f^{\alpha}\widehat{B}^{\beta} + K_e\left(K_{sf} f\widehat{B}\right)^2 \tag{4}$$

where $\widehat{B}$ is the peak amplitude of flux density and f is the motor electrical frequency, K_{sf} is the stacking factor of the lamination sheets which is empirically given as 1.5. K_h, K_e, α, and β are Steinmetz coefficients determined by fitting the loss data from manufactures for specific materials. K_{sf} is assumed as 1 in the analytical model of ML approach but, it is regarded as a variable in the coefficient optimization.

The assumption of using Eq. (4) is that the stator flux density is a sinusoidal wave. However, in fact, the flux densities in stator tooth tip and in the back-iron parts linking stator base are far from sinusoidal, even approximate linear relations in time domain. Moreover, analytical models traditionally use the densities of typical points to estimate stator losses for the sake of efficient computation which generates another difference between analytical and FEA model. Therefore, iron-loss estimation of using Eq. (4) at one or several typical points is prone to have large errors especially for different geometry and winding parameters. The iron material used in the test prototype machine is M235-35A steel whose Steinmetz coefficients can be found in Table 1.

4.1.2. Mechanical loss

Mechanical losses under no-load condition are estimated using empirical constants and mechanical loss relationship is given as:

$$P_{mec} = K_{fric}\omega_m + K_{wind}\omega_m^2 \tag{5}$$

where K_{fric} is the friction loss coefficient and K_{wind} is the windage loss coefficient. They can be calculated empirically to be $2\times10^{-3}\mathrm{W}\cdot\left(\mathrm{r}\cdot\mathrm{min}^{-1}\right)^{-1}$ and $3.3\times10^{-6}\mathrm{W}\cdot\left(\mathrm{r}\cdot\mathrm{min}^{-1}\right)^{-2}$, respectively, in this case study. However, these coefficients may not work perfectly with regards to the variation of mechanical speed. Thus, they can be adjusted according to practical data obtained from experiments. As mentioned, this study utilizes an optimization process to find the best design of these two coefficients (as well as K_{sf}) in order to match the experimental data, see Section 4.3.

4.2. Test rig

Fig. 6 shows the photo of the test bench of motor drive system. The 12-slot-10-pole (12s10p) prototype machine parameters are given in Table 2.

To measure the fundamental iron losses, the machine shaft is allowed to rotate freely, i.e. decoupled from any external loads. In the test motor drive system, the machine is fed through a power converter and it is speed controlled by applying several speed set points. For each speed, the electrical

machine input power is measured by means of an N4L PPA2530 precision power analyzer connected at the machine terminals. As stator current is close to zero (i.e. Joule losses are negligible), the measured input power consists mostly of fundamental iron loss. The measurements (P_{meas}) are plotted against mechanical speed are shown in Fig. 7 together with the estimations of iron losses from Eq. (4) in Section 4.1.1. Obviously, the measurements are significantly larger than the predicted values. The reason is that mechanical losses in Eq. (5) were not initially considered in the analytical model. Besides, core material machining has a huge effect on the actual machine's specific core loss. In order to improve the loss predictions, estimations of the mechanical loss are added by Eq. (5) and a correction factor, 1.5, is added to the fundamental iron loss calculation in Eq. (4). This correction factor is an empirical value, determined by trial-and-error. After the correction, the analytical predictions are much closer to the experimental data.

4.3. Coefficient optimization method

In this section, a search algorithm PSO is used to further improve the analytical predictions. There are three coefficients introduced in Eqs. (4) and (5): K_{sf}, K_{fric}, and K_{wind}, which are directly regarded as the DVs. All other parameters are assumed fixed in this optimization, for example parameters of M235-35A steel and motor geometry. The predicted total motor loss P_{tot} is just the summation of iron loss P_{iron} and mechanical loss P_{mec} using 3 DVs. There are 10 data points from experiment and the objective of the optimization is chosen as the Root Mean Square Error (RMSE) of the analytical predictions and the corresponding experimental data set. The diagram of the coefficient optimization problem is shown in Fig. 8.

This study utilizes a toolbox of PSO in MATLAB [41] to do the factor & coefficient optimization, i.e. minimize the RMSE between P_{tot} and P_{meas}. The initial ranges of three variables in optimization are given in Table 3 based on the empirical values. Swarm size is set as 20. In order to get the global optimum, derivative-free method is utilized to find minimum of unconstrained multi-variable function after the particle swarm algorithm terminates.

After 113 iterations with calling RMSE calculation for 2345 times, the optimal design of K_{sf}, K_{fric}, and K_{wind} is obtained (Table 3) with the minimum RMSE given as 0.1715 W. Compared with the empirical approach (RMSE = 1.5217 W), PSO of 3 variables gives a much better solution of PMSM loss correction.

4.4. Proposed surrogate-role model using ANN

Based on the proposed SR_1 and SR_2 in Section 3, two different ANN models are established in this case: one maps from ω_m to

Table 1 Parameters of M135-35A steel for PMSM stator.

Parameter	Value
Density (kg/m^3)	7600
Lamination thickness (mm)	0.35
K_h	0.0081294
α	1.208357
K_e	3.442366
β	1.78619

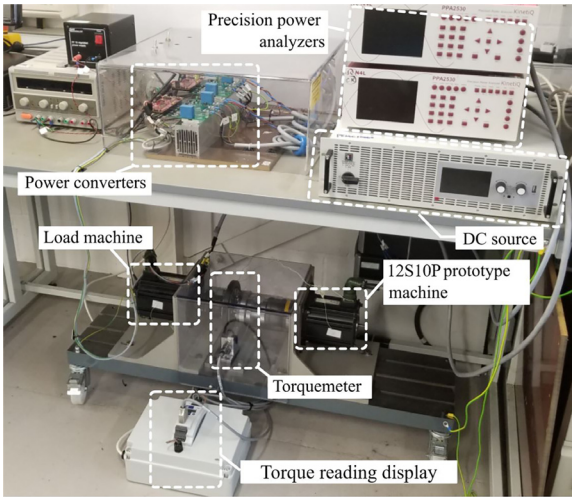


Fig. 6 Experimental test bench employed for power loss measurement.

Table 2 Main parameters of 12s10p prototype machine.

Parameter	Value
Rated speed (r/min)	1500
Rated output power (W)	1300
Rated phase current (A)	9.33
No. of turns per coil	30
Stack length (mm)	54
Stator outer radius (mm)	61

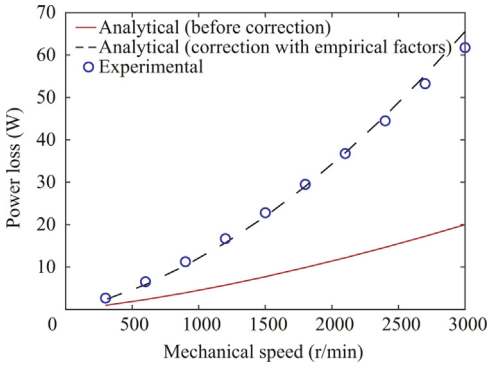


Fig. 7 Power loss measurement against mechanical speed (before and after correction).

P_{meas}, and the other maps form ω_m to the iron loss correction factor. In particular, the ANN of SR_1 represents the following relation:

$$y = F(x) \Longleftrightarrow (P_{meas}) = F_{1,SR_1}(\omega_m) \tag{6}$$

which means this ANN is learning the relation from the DV ω_m to the target data P_{meas}, replacing all the experimental and analytical models of motor total losses (under no-load condition). Regarding the ANN of SR_2, it represents another relation:

$$y = F(x) \Longleftrightarrow (k_{c,ml}) = F_{1,SR_2}(\omega_m) \tag{7}$$

where $k_{c,ml}$ is the iron loss correction factor linking P_{iron} to measured machine loss P_{meas} thus, it can be directly computed by

P_{meas}/P_{iron}. After getting loss data from experiment in some sample conditions (ω_m), P_{iron} can be given at same conditions by Eq. (4) based on the accurate estimation of $\widehat{B}$[6]. Then, the target $k_{c,ml}$ can be obtained by the above simple division. Lastly, the ML model is built by training the feedforward ANN of Eq. (7).

Since there are only 10 data points with 1-element-input and 1-element-output, two ANNs are both simply using one-hidden-layer structure with 3 hidden neurons. They can be trained using the train command, which is a part of MATLAB's Deep Learning Toolbox [42]. For the training parameters, learning rate is set as 0.1, training goal (RMSE) is set as 0.001, and the maximum epoch number is 500. Both ANN can be trained in few seconds on a standard computer. Since the training starts from random values, the training was tried 10 times and pick up the best one for each ANN. The final RMSE results of both PSO and SR methods are shown in Fig. 9.

4.5. Discussion of Case I

Generally, seen from Fig. 9, the proposed two ANN SR models are better than the PSO, and the empirical correction approach performs the worst in this case. SR_1 ANN model is better than SR_2 model but the advantage is not obvious. Here are main reasons of this conclusion: first of all, the empirical approach only uses one factor, 1.5, to correct iron loss data, which is hard to balance the loss predictions under different speed conditions; then, for the PSO, the main challenge is to choose the ranges for 3 DVs, i.e. the size of design space of optimization, which would directly determine the optimization results; lastly, ANN has a strong generalization capability with no constraint or design space for ANN inputs/outputs, even for the 3-hidden-neuron structure, there are many parameters (i.e. weights and bias, see Section 2.2) to be trained for the target mapping. In addition, it is noted that the RMSE of ANN approach can be further improved if the number of hidden-layer neurons is increased; therefore, compared with empirical factor and PSO approaches, the proposed ANN approach should be the best way to give the total loss predictions.

Another candidate of surrogate model is using the fitting function, which is feasible in this case. However, the mapping performance of fitting is always limited by the pre-defined functions (e.g. polynomial, exponential). In addition, it would become extremely hard to choose the functions for high-dimensional data sets. In contrast, the ANN model has no limit for the number of elements in inputs and outputs because it is based on a nonparametric regression model. User does not need to specify the relationship between the predictors (input) and responses (output) since ANN can learn them automatically by using the training parameters.

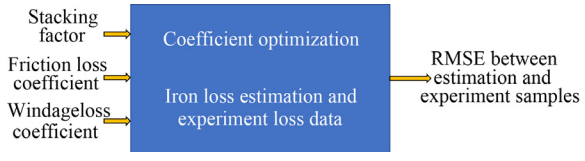


Fig. 8 Diagram of coefficient optimization.

Table 3 Correction factors in ACF and RMSE results in NN approach.

Variable	Empirical	Optimization ranges	Optimal design
K_{sf}	1.5	[1, 3]	1.675
$K_{fric}\left[\mathrm{mW}\cdot\left(\mathrm{r}\cdot\mathrm{min}^{-1}\right)^{-1}\right]$	2	[1, 4]	3.52
$K_{wind}\left[\mu\mathrm{W}\cdot\left(\mathrm{r}\cdot\mathrm{min}^{-1}\right)^{-2}\right]$	3.3	[2, 5]	2

5. Case II

This section will discuss the case of motor-drive multi-objective optimization which uses the detailed simulation data as the target data for ANN training. In the studied motor drive system, there are two main parts: a two-level VSI (DC-AC) and a PMSM, as shown in Fig. 10. It is assumed the input DC interface is connected to a DC bus within an aircraft electrical power system and remains unchanged. Noted that, though this case is for aircraft actuation applications, the proposed ANN aided approach is independent with the drive's operation scenario. The scenarios' parameters can be fixed in the optimization or, they can be also set as design variables. No matter in which situation, the steps of the proposed ANN aided PMSM optimization methodology would not be different. The motor drive here is used for an actuator onboard MEA. The design space of optimization would be different for different motor-drive application scenarios, either actuation, fan, or compressor. And, the design space should be predefined before using the ANN aided design and optimization method.

For the control of such a drive system, the PI controller of a two-level inverter provides the converter modulators with appropriate output voltage vector references, to regulate the motor output torque and speed [40,43]. For field-oriented vector control, the sinusoidal *abc* current and voltage components are transformed into a rotating *dq* frame. This enables the zero steady state error tracking of the current references with simple PI compensator [40,43–45]. With the conventional cascaded control scheme employed, motor currents are controlled in the inner control loop, and the speed control is in the outer loop. In addition, to ensure that converter voltage limits are not exceeded, a field-weakening strategy in Ref.[44] is adopted in this motor-drive system.

Optimization of motor-drive system involves in two objectives: minimization of total mass and total power loss of the drive system. For each design point, the PMSM model is fed by converter currents operating at a certain condition where both DC supply and motor output reference should be assumed (for example, motor speed, motor output torque). Then, power losses of two subsystems (converter and PMSM) are computed separately. Regarding the mass estimation, converter mass is assumed fixed in this study while the motor mass will be different against the changes of Design Variables (DVs).

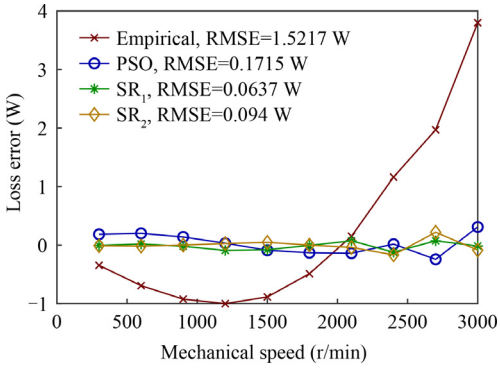


Fig. 9 Loss error comparison using four different methods (Case I).

In the analytical approach, some key parameters of the PMSM (such as motor torque, *dq* inductance, etc.) should be provided to the converter analytical model for the further evaluation of the converter's performance [40,46]. Therefore, in the detailed simulation, PMSM FEA model should be first exercised then the drive system can be designed and run with all necessary parameters. In order to pursue high accuracy of the detailed simulation, a joint simulation scheme of motor drive systems is proposed in this case. This scheme can combine the PMSM FEA model (using MotorCAD) with the drive simulation system (using PLECs) under the operation of MATLAB script codes.

5.1. Joint-simulation for data collection

Conventionally, the motor-drive system is mainly simulated on a system-level platform (for example, MATLAB/Simulink, PLECs), which does not consider the geometrical parameters of components. In contrast, the proposed joint simulation takes PMSM FEA model into account thus it can regard the detailed PMSM geometries as the design parameters for a motor-drive optimization problem. In this case study, the PMSM FEA model is utilized here to generate the accurate motor key parameters (which are feedforward to PLECs for the system-level simulation) and performance (for example motor mass, power losses). Lastly, since the key role of SR_2 is to establish an ANN-based surrogate model of performance correction linking the analytical model to the detailed simulation model, the data from both analytical model and simulation model are required for the ANN training. To this end, the proposed joint simulation scheme integrates all the required models using MATLAB codes, which can collect the desired data efficiently and automatically.

The diagram of the proposed joint simulation scheme is depicted in Fig. 11. First, the used motor-drive analytical model (built on the MATLAB) is systematically introduced in Ref.[40]. The resulting model is a non-iterative and high dimensional sizing model which costs less than 0.1 s to run on a standard PC, and its accuracy under certain conditions was verified by both simulation and experimental tests. For each design point, the analytical model can be run based on the inputs of parameters (for example, DC voltage, motor current, geometry, materials) and the performance of both converter and PMSM can be estimated. Then, PMSM FEA model is exercised based on the same motor input parameters, and the motor performance index values can be obtained by FEA. At the same time, some key parameters of PMSM, which are required in the motor-drive system-level simulation, can be extracted from the FEA model and then feedforward to the motor-drive system on PLECs. These parameters include Torque reference (T), dq-axis inductance (L_d, L_q), flux linkage established by magnets (flux), DC cable resistance (R) and the motor inertia. Moreover, for the converter subsystem, the basic operation input parameters (i.e. input power, motor electrical frequency, DC voltage, and switching frequency) are also the same with the MATLAB analytical model. Finally, run the entire drive system and the converter power loss data can be obtained. It is noted that, to run the PLECs model (together with other models) using MATLAB codes and collect the training data easily, this PLECs model is embedded into Simulink platform in this study.

As discussed in Section 3.1, analytical design methods pursue computational efficiency based on simplifications and assumptions, thus they are prone to lose the estimation accuracy compared with detailed simulation models. Most of factors that could generate power loss are not considered in analytical methods (for example, non-uniformed flux distributions in stator tooth or yoke). Moreover, analytical models often use linearization techniques to simplify the loss calculation, however practically, the power losses within a PMSM drive also include losses due to the non-linear characteristics.

To this end, the proposed SR_2 of ML in this study is to use an ANN to bridge the gap between analytical models and the detailed simulations (FEA + drive circuit system). The ANN-aided method in SR_2 will not only correct the analytical results with high accuracy but also leave the complicate simulations of motor drive in the optimization. Though the analytical basic models are just simple equations, the dedicated ANN-based correction model can accurately and efficiently link them to the joint simulation model in a design space. Besides, the proposed ML SR_1 provides an alternative way of generating the target motor-drive performance data when given an input combination, without considering the analytical models and the correction. The next subsections will first introduce the SR_1 and SR_2 functions using different ANN designs and then discuss how to comprehensively obtain the surrogate models in this case.

5.2. Surrogate roles in optimization and research thinking

In this section, three different schemes of ANN design would be proposed one by one for the mapping of two optimization objectives (power loss and mass). The first scheme is using one ANN mapping two objectives thus there would be two outputs in this ANN design. For the second scheme, two objectives are separated for the ANN training; Therefore, the ANNs only have one neuron in the output layer. In the third scheme, two objectives will be integrated into one before training the ANN. Based on these three ANN schemes, the research thinking of ANN training and validation will be introduced and discussed afterwards.

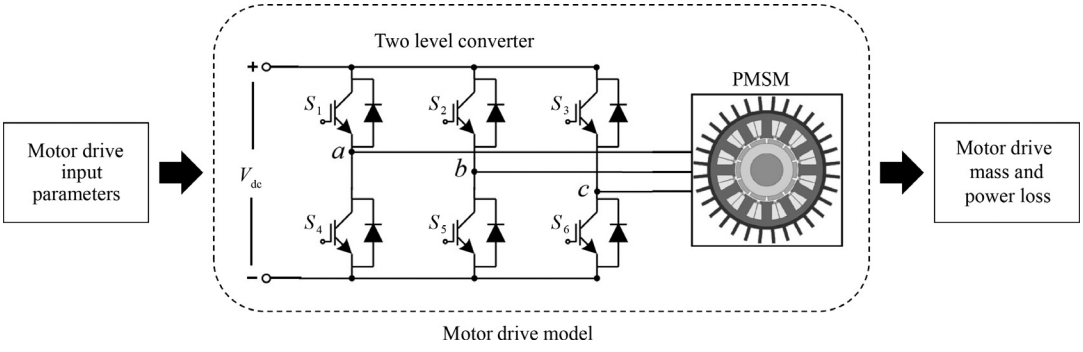


Fig. 10 Diagram of motor-drive case study.

5.2.1. ANN design scheme with two outputs, Scheme 1

Three DVs are studied in this case: switching frequency of inverter (DV1, f_{sw}, [10, 20] kHz), PMSM airgap height (DV2, h_{ag}, [1, 3] mm) and PMSM wire diameter (DV3, d_{wire}, [0.6, 0.7] mm). Other parameters in this case can be found in Table 4. In Scheme 1, two objectives are put into one ANN as the outputs for both SR_1 and SR_2. Based on the proposed SR_1 and SR_2 in Section 3, two different ANN models are established in this scheme: one maps from three DVs to the total motor-drive power losses P_{drive} and mass M_{drive}, and the other maps from three DVs to the loss/mass correction factors. In particular, the ANN of SR_1 in Scheme 1 represents the following relation:

$$y = F(x) \Longleftrightarrow (P_{drive}, M_{drive}) = F_{SR_1}^{Scheme1}(f_{sw}, h_{ag}, d_{wire}) \tag{8}$$

which means this ANN is learning the relation from the three DVs to the target data P_{drive} and M_{drive}, replacing all the simulation models of motor-drive performance. Regarding the ANN of SR_2, it represents another relation:

$$y = F(x) \Longleftrightarrow (k_{c,loss}, k_{c,mass}) = F_{SR_2}^{Scheme1}(f_{sw}, h_{ag}, d_{wire}) \tag{9}$$

where $k_{c,loss}$ is the motor-drive loss correction factor and $k_{c,mass}$ is the total mass correction factor. They are linking the analytical model estimations ($P_{a,drive}$ and $M_{a,drive}$) to the target data of joint simulation, P_{drive} and M_{drive}. The correction factor $k_{c,loss}$ is given by $P_{drive}/P_{a,drive}$, and $k_{c,mass}$ is given by $M_{drive}/M_{a,drive}$.

5.2.2. ANN design scheme with one output, Scheme 2

The second ANN design scheme decouples the two objectives by using two separated neural nets, i.e. one ANN is for mapping the power loss, the other for the mass mapping. This design scheme trains the ANNs only with one output thus it can get rid of the linkage between two objectives during the training. The ANN of SR_1 in Scheme 1 represents the following relation:

$$y = F(x) \Longleftrightarrow \begin{cases} P_{drive} = F_{SR_1,P}^{Scheme2}(f_{sw}, h_{ag}, d_{wire}) \\ M_{drive} = F_{SR_1,M}^{Scheme2}(f_{sw}, h_{ag}, d_{wire}) \end{cases} \tag{10}$$

which means one ANN is learning the relation $F_{SR_1,P}^{Scheme2}$ from the three DVs to the first objective P_{drive}, the other ANN is learning the relation $F_{SR_1,M}^{Scheme2}$ from DVs to the second objective M_{drive}. Regarding the SR_2 in Scheme 2, it represents the correction approach using the following two ANNs:

$$y = F(x) \Longleftrightarrow \begin{cases} k_{c,loss} = F_{SR_1,P}^{Scheme2}(f_{sw}, h_{ag}, d_{wire}) \\ k_{c,mass} = F_{SR_1,M}^{Scheme2}(f_{sw}, h_{ag}, d_{wire}) \end{cases} \tag{11}$$

Therefore, all the inputs/outputs in Scheme 2 are the same with the Scheme 1 but using one more ANN for each SR. Obviously, both Scheme 1 and Scheme 2 are focusing on two objectives of the motor-drive optimization problem. In contrast, the next design scheme (Scheme 3) will use an integrated objective for the ANN training, which leaves two outputs and only needs one element in the ANN output layer.

5.2.3. ANN design scheme mapping the integrated objective, Scheme 3

Since this case is studying on a multi-objective optimization problem, a Pareto front will be usually generated after optimization. In order to further give the best design point, an integrated index r of two objectives is utilized to select one particular point for the Pareto front in objective space, the criterion for this decision-making solution is the minimal distance from ideal objectives [21,23,47]:

$$\text{Solution} \equiv \min(r_i) \tag{12}$$

where

$$r_i = \sqrt{\lambda_{MA}\left(\frac{M_i}{M_{max}}\right)^2 + \left(\frac{P_i}{P_{max}}\right)^2} \tag{13}$$

where M_i is the mass of the i^{th} solution; M_{max} is the maximal mass of all designs; P_i is power losses of the i^{th} solution; P_{max} is the maximal power losses of all designs; λ_{MA} is the weight of mass objective.

λ_{MA} is usually set as 1 which means two objectives have the same priority. But, in practice, different objectives can be prioritized by changing the weight value in Eq. (13). Obviously, the best design obtained from Eqs. (12) and (13) will differ with regards to λ_{MA}. In particular, when λ_{MA} increases, power loss of the best design will climb up while the mass will

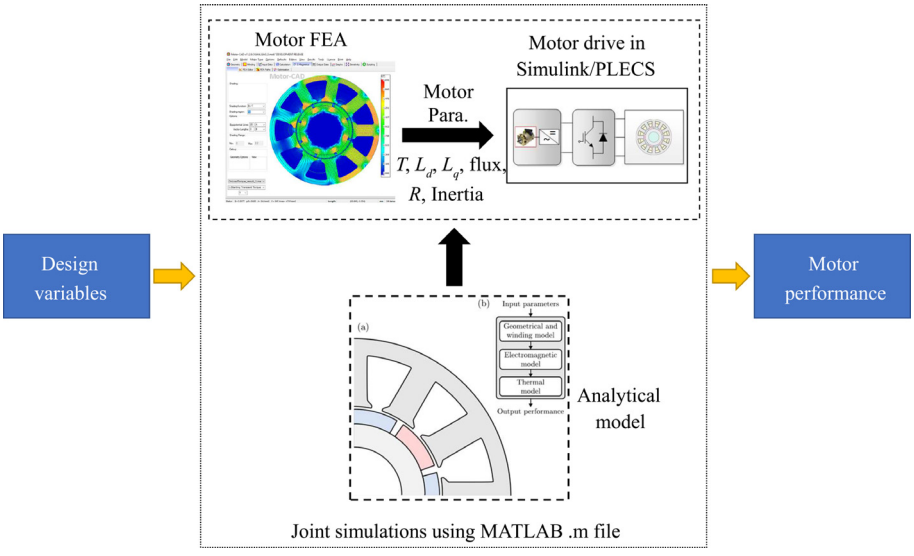


Fig. 11 Joint simulation scheme of motor drive system.

decrease. For the MEA application, λ_{MA} is usually set larger than 1 since the mass objective has a higher priority. In this case study, λ_{MA} is pre-assumed as 3 according to the finding of Ref.[21].

Based on Eqs. (12) and (13), the third scheme of ANN is designed to map from DVs directly to the r value, which leaves two objectives in the ANN training. The motivation of this design is that, the best design point of optimization is directly determined by r value thus this ANN design scheme can also serve as the basis of optimization. Obviously, after the sample data collection, the objective data should be pre-processed using Eqs. (12) and (13) to give r value, which is required by the ANN training in Scheme 3. Therefore, the ANN of SR_1 in Scheme 3 represents the following relation:

$$y = F(x) \Longleftrightarrow (r_{\text{drive}}) = F_{SR_1}^{\text{Scheme3}}(f_{sw}, h_{ag}, d_{\text{wire}}) \tag{14}$$

which means this ANN is learning the relation from the three DVs to the processed target data r_{drive}, by using the samples collected from the joint simulation models. On the other hand, the ANN of SR_2 represents the relation:

$$y = F(x) \Longleftrightarrow (k_{c,r}) = F_{SR_2}^{\text{Scheme3}}(f_{sw}, h_{ag}, d_{\text{wire}}) \tag{15}$$

where $k_{c,r}$ is the motor-drive correction factor of r value. $k_{c,r}$ is calculated by $r_{\text{drive}}/r_{a,\text{drive}}$, where $r_{a,\text{drive}}$ denotes the r value estimation of motor-drive analytical models.

As discussed above, in the proposed three ANN schemes, the first two are mapping from design variables to the two (or more) objectives while the third one is mapping the integrated objective, r. And the unique difference between the first two schemes is just whether using only one ANN to do the mapping or not. Regarding the studied motor-drive optimization problem, three variables ($f_{sw}, h_{ag}, d_{\text{wire}}$) are considered here and three ANN schemes (six ANN models due to two SRs) are all using r value to represent the fitness of design points and to determine the final best design. However, as given by above equations, the first two ANN schemes should output the objective values first and then use Eq. (13) to obtain r values; In contrast, the third scheme directly generates r values based on variable values without using the original two (or more) objectives.

Table 4 Parameters and variables of motor-drive case study.

Parameter	Value
Rated speed (r/min)	1500
Electrical frequency (Hz)	125
Input RMS current (A)	10.162
No. of turns per phase	80
Axial length (mm)	54
Depth of tooth tip (mm)	1.26
Depth of tooth base (mm)	19.67
Tooth tang angle (°)	38.38
Tooth base faction	0.6033
PM height (mm)	4.4
Coil density (kg/m^3)	8960
Coil conductivity (S)	5.7×10^7
Rotor radius (mm)	28.6
Yoke Depth (mm)	6.88
Tooth tip fraction (%)	78.83
PM fraction (%)	88.19
PM density (kg/m^3)	7500
PM relative permeability	1.033
Shaft radius (mm)	17.5
Shaft density (kg/m^3)	7820
Design variable	**Range**
DV1: Switching frequency (kHz)	[10, 20]
DV2: Airgap height (mm)	[1, 3]
DV3: Wire diameter (mm)	[0.6,0.7]

5.2.4. Research thinking of ANN training and validation

In order to implement the training of three design schemes (totally six ANNs), a small number of sample data should be collected from the joint simulation. After that, the surrogate ANN can be efficiently trained by these samples. Regarding the neuron number in the hidden layer of six ANNs, it can be set by an algorithm developer. However, it is all simply set as 6 in this motor-drive optimization study. The reason is that there are only 27 sample points in ANN training thus good regression performance can be easily obtained with 6 hidden neurons, which benefits from the good global generalization capability of ANN. All surrogate models were trained using the same train command. For their training parameters, learning rate is set as 0.1, training goal (RMSE) is set as 0.00001, and the maximum epoch number is 1000. The ANNs can be trained in few seconds on a standard computer. Their validation performance will be given in Section 5.3.

Furthermore, the trained ANNs will be tested by the new data collected from another round of joint simulation for the ANN model validation. The new round of joint simulation will generate new data for ANN validation. If the ANN predictions are very close to the joint-simulation data at these new samples, the validation is successful (for example, the relative error of all samples are smaller than 1%). Otherwise, the validation fails, and the ANN should be re-trained. The diagram of the motor-drive performance validation is depicted in Fig. 12. Details of the four steps in this approach are discussed as follows:

Step 1. The joint simulation establishment.As discussed above, MATLAB analytical model is used to feedforward parameters to two different simulation platforms (Motor-CAD for PMSM loss/mass analysis, PLECs for converter loss analysis). Noting that most of the parameters delivered to MotorCAD (geometries, windings, input current etc.) do not need to be delivered to PLECs (for example PM height, slot fraction, motor axial length), because the MotorCAD model is the detailed FEA model of PMSM while the PLECs is a high-level simulation model of the motor-drive circuit.

Step 2. The data collection via the established joint simulation.At the beginning, a design space with one or several variables should be assumed for a motor-drive multi-objective optimization problem. Then, sample at least two values (i.e. upper/lower boundary) for each design variable. For example, in this case, there are 3 design variables and 3 values are sampled for each variable during the first round of joint simulation. Therefore, there are totally $3^3 = 27$ design points. Since 3 different values are sampled for each variable, this sampling scheme is denoted as "sweep3". For each design point, the joint simulation should be run for one time and the corresponding power loss results of both analytical and simulation models can be obtained. Therefore, the second step of this case is mainly running the joint simulation based on a sampling scheme in the design space and collected the detailed power loss data. The "sweep3" sampling is further illustrated in Fig. 13.

Step 3. ANN training part after getting the processed training data from the joint-simulation models.The power losses of different components are considered in the joint simulation: the PM loss, stator (York and Tooth) iron loss, and copper loss in PMSM, as well as the conduction loss, switching loss in the converter devices (i.e. six IGBTs and six Diodes). That means, in SR_2, all the analytical models are needed and all the loss values should be collected from the joint simulation (with regards to design variables). However, in the ANN training stage of SR_1 and SR_2, all these power losses were summated up to generate the whole power loss of the motor drive. Regarding the mass of motor drive, as mentioned, the inverter mass is assumed fixed (as 5 kg) and the motor mass will change with regards to the variables.

Step 4. Validate/test the trained ANN using another round of joint simulation.In this step, it is suggested using new data as much as possible thus the sampling scheme should be different from that in Step 2. In this case study, 4 different values are sampled for each variable (named as "sweep4", shown in Fig. 13) and the second round of joint simulation is then exercised. Under this sampling scheme, for each design point, the loss/mass/r estimation from the analytical model need to be corrected by the corresponding correction factor (given by SR_2 ANN); after that, the corrected loss/mass/r value should be compared with the collected simulation data to see whether the error is acceptable. For the other ANN SR_1, the predictions can be directly compared with the collected new data. If the error at every design point is very small (for example, relative error smaller than 1%), the validation is successful. Finally, the validated ANNs can be used to do the motor-drive power loss optimization using the exhausted algorithm, which will be discussed in the next subsection.

5.3. Validation results of surrogate roles in three schemes

Fig. 13 shows the overall scheme of sweep3 and sweep4. The first and the last values of each DV are the boundary values of their ranges while the other sampled values are evenly distributed in the middle. For example, in sweep4, a_1 and a_4 are the lower and upper boundaries of DV1, and the distance of adjacent values equals $(a_4-a_1)/3$. Obviously, in sweep4, most of 64 sample points are not included in sweep3 sampling which provides the training data (27 samples) for ANN. One the other hand, Fig. 13 also depicts the series number of DV combinations for each sample value of DV3. In sweep3, the series numbers are given as 1,2,...,9 for every value of DV3. And in sweep4, when a value of DV3 is confirmed, there will be 4×4 designs whose series numbers are marked as 1,2,...,16, which serves as the basis of demonstration in the following validation results.

After training and test, the relative errors of the abovementioned six ANNs using the sweep4 design points are shown in Figs. 14–16. In these figures, the x axis all represents the series number (1 to 16) of motor-drive design points and every solid/dotted line gives the relative prediction errors of 16 design points with a certain value of DV3. Since both Scheme 1 and Scheme 2 consider two objectives as the ANN outputs, they are compared in Figs. 14 and 15 with two subfigures (one subfigure for loss prediction, the other for mass prediction). And in Figs. 14&15, all Scheme 1 results are depicted using solid lines while dotted lines are used for Scheme 2. Based on Scheme 3, the integrated objective (r value) is considered as the unique ANN output and two ANN models from SR_1 and SR_2 are compared in the Fig. 16.

As shown in Figs. 14&15, generally, the validation performance of Scheme 2 is a little better than the Scheme 1, no matter for which surrogate role, SR_1 and SR_2. The main reason is that, Scheme 2 decouples the linkage between two objectives by training two independent ANNs thus, Scheme 2 can get a better training and prediction performance. For example, regarding the mass prediction shown in Fig. 14(b), all 64 absolute errors of Scheme 2 are smaller than 0.05%; In contrast, half of the absolute errors in Scheme 1 are larger than 0.05% and the "$D_{\text{wire}} = 0.667$ mm" results are larger than 0.06%. However, it should be noted that, the prediction difference between Scheme 1 (solid lines) and Scheme 2 (dotted lines) is not much in this case study, especially for the SR_2 results in Fig. 15. The biggest relative RMSE difference is only 0.062%, as shown in Fig. 14(a). This is mainly due to the good learning capacity of ANN and the small amount of training data set. Both Scheme 1 and Scheme 2 can easily and effectively learn the designed relations with negligible errors.

On the other hand, compare Fig. 14 with Fig. 15, it is found that the SR_2 models perform better than the SR_1 models, no matter for which ANN design scheme. Regarding the mass prediction (right-side subfigures), the relative RMSE of SR_1 models is 20 times bigger than SR_2 models in Scheme 2, and 30 times bigger than SR_2 models in Scheme 1. Therefore, in this case, using ANN predictions to correct the analytical models (SR_2 approach) makes the ANN training easier than that of direct mapping approach (SR_1). It is important to notice that, the SR_2 approach is not always superior than the SR_1, the performance of trained model depends on the characteristics of training data and should be case-specific. Namely, which surrogate role is better is mainly determined by the characteristics of ANN input–output data. After the correction process in SR_2, the data may become easier to train an ANN but, in some cases, the direct mapping approach (SR_1) can be a better choice if the original motor-drive performance data shows an easier relationship with regards to DVs. In the Case I, the loss prediction performance of SR_1 is a little better than SR_2, see Section 4.

Finally, Fig. 16 gives the validation results of Scheme 3. The results from the SR_1 ANN model are depicted by solid lines while dotted lines are used for the error results of SR_2 ANN model. Obviously, SR_1 approach performs better than the SR_2 approach (with a RMSE difference of -0.05%). This conclusion is the same with the comparisons in Scheme 1 and Scheme 2 because all three ANN design schemes are use the same training data (from sweep3) and the same validation data (from sweep4).

5.4. Optimization results

After ANN training and validation, for the multi-objective optimization of drive system, a large amount of data should be sampled for the global optimization framework. As summarized in Table 5, 11 values of f_{sw}, 11 values of h_{ag}, and 11 values of D_{wire} were sampled. The total sample number is 1331. Besides, the maximum and minimum values of power loss and mass from analytical models and FEA are also given in Table 5. Among that, the maximum objective values should be used in Eq.(13) to derive the r values for training and optimization.

Based on the proposed two surrogate roles (Section 3) and three design schemes (Section 5.2), six ANNs were trained separately and their validation results have been given in the last subsection. After that, the optimization using six ANNs were exercised one by one using the small sample steps. No matter for the direct mapping approach (SR_1) or the ANN-aided cor-

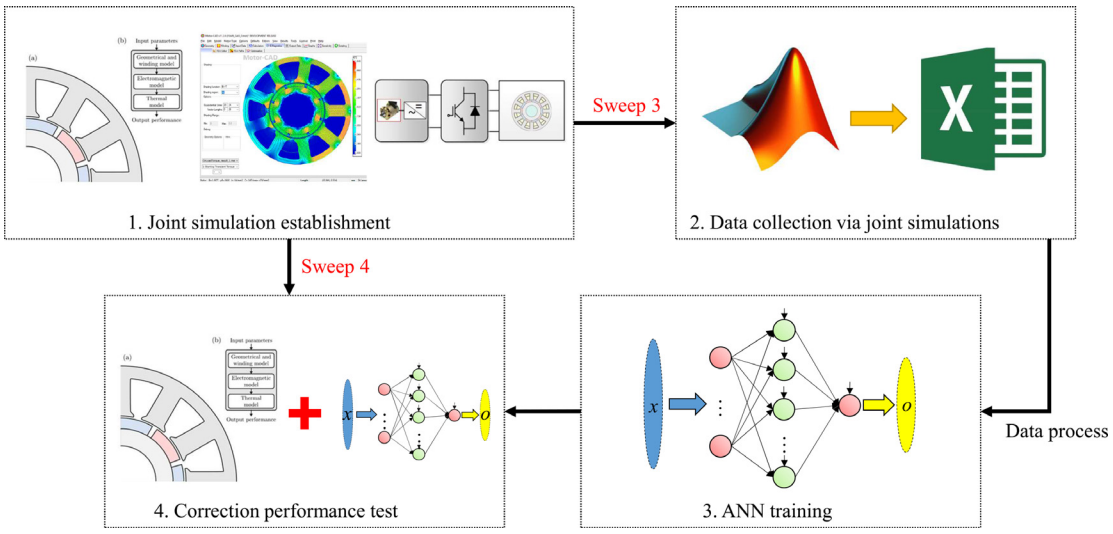


Fig. 12 Research thinking of ANN training and validation.

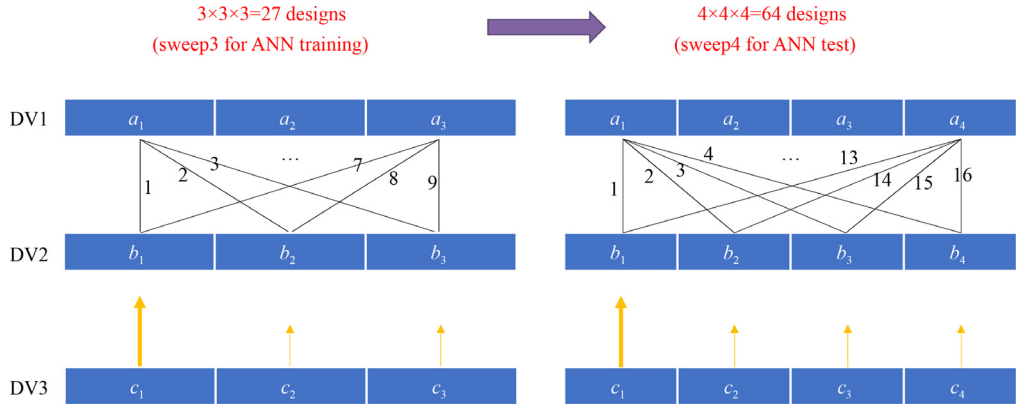


Fig. 13 Design variable combinations for sweep3 and sweep4.

rection approach (SR_2), the optimization can all be finished in a short time (see the last column in Table 6) because both analytical models and trained ANNs only contain simple math functions which are running very fast. However, since the direct mapping approach does not need to run the motor-drive analytical models, it is faster than the correction approach. After the optimization, the best design from all six ANN-based methods are summarized in Table 6, together with their time cost of doing optimization on a standard computer. Besides, the optimal design was used as the inputs to simulate the detailed joint-simulation (one design point costs more than 1.5 min), and the total motor-drive loss and mass (target data) were also obtained and given in Table 6. It would cost a very long time if only using the joint simulations (conventional approach) to sample the 1331 design points.

It is found that, all six models can result into the same optimal design point. Though they are predicting different values of power loss, mass, and r, their differences and the errors with the joint-simulation data all remain in a very low level. The absolute relative errors of power-loss prediction are no bigger than 0.2% and for the mass prediction, the errors are smaller than 0.071%. The target r value of this optimal design is 1.844541, calculated by Eq. (13); Therefore, the relative r errors are within [−0.0273%, 0.0011%], extremely close to the target data. That is why all six models can generate the same best design point. In addition, seen from Table 6, the two conclusions of the model comparison in the validation (Section 5.3) also fit the results after optimization: Scheme 2 performs better than Scheme 1 and SR_2 models give a better prediction than SR_1.

Finally, choose the scheme/role candidate (Scheme 3, SR_2) which has the closest r value (optimal) with the target data to output the optimization results for this problem. All the 1331 samples and the best design point are plotted in Fig. 17. There is a clear Pareto front in this 2-dimensional objective space, which shows the trade-off between two objectives. As mentioned, all six models can result into the same optimal design point and very close predictions. Therefore, the optimization results in the objective space from other five models should be extremely closed to that in Fig. 17.

5.5. Discussion of Case II

Seen from above discussions of feasible ANN design schemes and their optimization results, the proposed two SRs can be easily applied into three different ANN schemes in this multi-objective motor-drive optimization problem. More importantly, all ANNs are efficient and can provide accurate specific estimations for the optimization. Therefore, compared with the conventional FEA-based methods, the proposed ML SRs of motor drive are time-saving to do the optimization. As mentioned in Section 3.2, another advantage of the ANN-

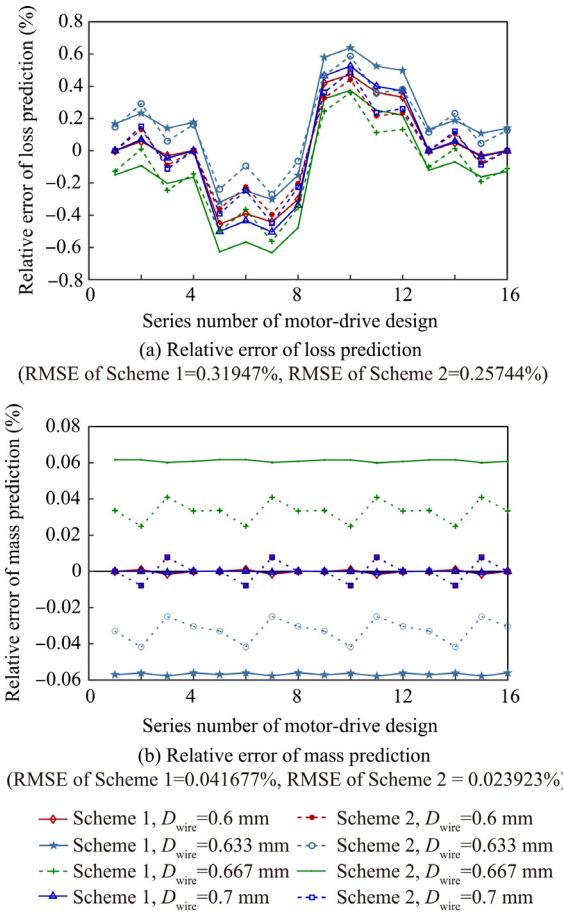


Fig. 14 Validation results of SR_1 models in Scheme 1 and Scheme 2.

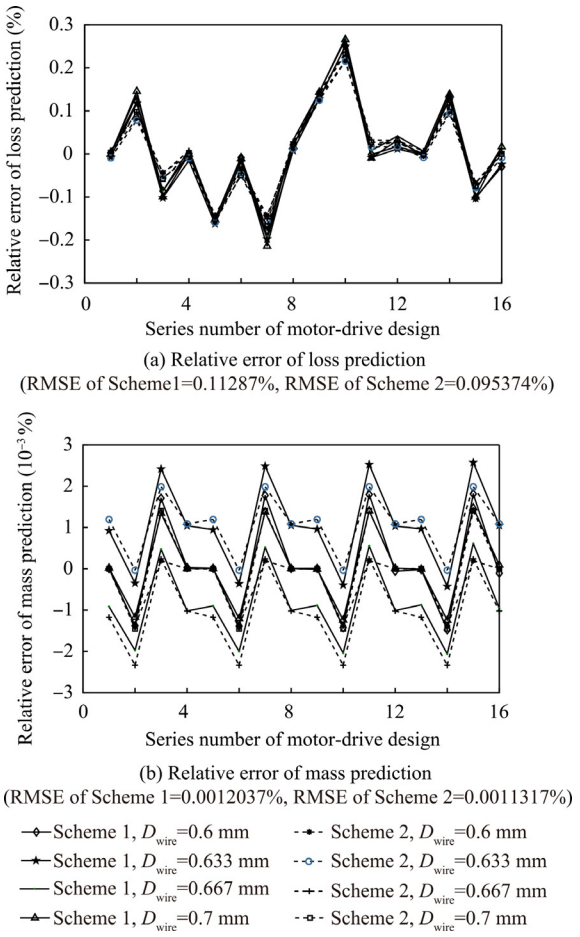


Fig. 15 Validation results of SR_2 models in Scheme 1 and Scheme 2.

aided approach is that it has no risk of getting stuck in local optimum due to the utilization of exhaustive algorithm. For the FEA-based optimization using search algorithms, they usually need to sample hundreds even thousands of design points and the search algorithm may also bring the risks of local optimum. Lastly, the six trained ANN models can all be feasible for the optimization in any potential design subspaces (the original space can be found in Table 5), also quickly and accurate, without training new ANNs. This fact does not fit with the conventional search methods, there may be few of previously searched samples in a subspace thus they cannot be used to determine the optimal design.

However, there is one limitation of the proposed ANN aided optimization approach: the trained ANN(s) may not work well for the variable ranges out of the original design space. Namely, if enlarge the design space, the prediction performance of the trained ANN(s) cannot be guaranteed. It is because the raw data collected will not cover the whole design space. For the design points out of range, the FEA model (or experiment) is still needed to train new ANN(s), which can then provide the accurate estimations for the whole space. Therefore, it is suggested the motor-drive variables and their ranges should be well selected before using the ANN aided approach.

6. Conclusions

This paper considered two surrogate roles (SRs) of ANN for motor-drive optimization problems: SR direct mapping approach (SR_1) and SR correction approach (SR_2). Two surrogate roles are defined after introducing the fundamentals and surrogate functions of ANN. In addition, technical details of training ANNs are given in two case studies, including sample collection, data process, ANN training, validation of trained ANNs, etc. In the first case, the trained SR models are discussed, and they show the priority of accuracy and feasibility over other surrogate candidates. In the second case, three design schemes of ANN are proposed for the motor-

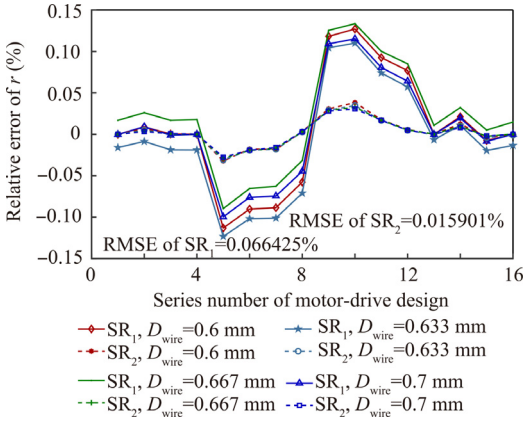


Fig. 16 Validation results of two ANNs in Scheme 3.

Table 5 Design space of motor drive system.

	f_{sw}(kHz)	h_{ag}(mm)	D_{wire}(mm)	$P_{a,drive}$(W)	P_{drive}(W)	$M_{a,drive}$(kg)	M_{drive}(kg)
Range	[10,20]	[1,3]	[0.6,0.7]	[112.0836, 166.1161]	[96.4362, 145.4770]	[8.9557, 9.3021]	[9.0498, 9.4073]
Sampling step	1	0.2	0.01				

Table 6 Best design point of six ANN approaches.

Scheme	SR	f_{sw}(kHz)	h_{ag}(mm)	D_{wire}(mm)	Loss prediction (W)	P_{drive}, target data (W)	Mass prediction (kg)	M_{drive}, target data (kg)	r	Time cost (s)
1	SR_1	10	2.4	0.62	107.952468	107.7415	9.168831	9.175265	1.844038	0.051
	SR_2	10	2.4	0.62	107.631148	107.7415	9.175445	9.175265	1.844267	58.04
2	SR_1	10	2.4	0.62	107.859951	107.7415	9.172392	9.175265	1.844383	0.061
	SR_2	10	2.4	0.62	107.686897	107.7415	9.175370	9.175265	1.844407	43.18
3	SR_1	10	2.4	0.62		107.7415		9.175265	1.844120	0.03
	SR_2	10	2.4	0.62		107.7415		9.175265	1.844561	42.08

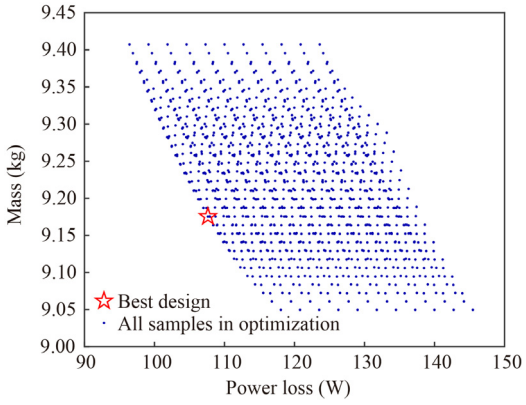


Fig. 17 Optimization results in the two-objective space, Scheme 3, SR_2.

drive multi-objective optimization problem and two SRs were tried for every design scheme. After the ANN training and validation analysis, it is found that both SRs can be easily applied into three different ANN schemes in this motor-drive optimization; six different ANN models are all efficient and accurate, and they can predict the same optimal design point though their ANN designs are totally different. Generally, SR_2 represents a small superior over SR_1 in this optimization case but noting that, the performance of SR depends on the characteristics of collected training data thus should be case-specific.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This project has received funding from the Clean Sky 2 Joint Undertaking under the European Union's Horizon 2020 research and Innovation Programme No. 807081.

References

1. Sutton RS, Barto AG. Reinforcement learning: An introduction. Cambridge: MIT Press; 2018.
2. Skansi S. Introduction to deep learning: From logical calculus to artificial intelligence. Berlin: Springer; 2018.
3. Bishop CM, Nasrabadi NM. *Pattern recognition and machine learning*, Vol. 4. Berlin: Springer; 2006.
4. Li Y, Lei G, Bramerdorfer G, et al. Machine learning for design optimization of electromagnetic devices: Recent developments and future directions. *Appl Sci* 2021;**11**(4):1627.
5. Bramerdorfer G, Tapia JA, Pyrhönen JJ, et al. Modern electrical machine design optimization: Techniques, trends, and best practices. *IEEE Trans Industr Electron* 2018;**65**(10):7672–84.
6. Cheong B, Giangrande P, Zhang X, et al. Fast and accurate model for optimization-based design of fractional-slot surface PM machines. *2019 22nd international conference on electrical machines and systems (ICEMS)*. 2019. p. 1–6.
7. Zhang Z, Huang J, Jiang Y, et al. Overview and analysis of PM starter/generator for aircraft electrical power systems. *CES Trans Electr Mach Syst* 2017;**1**(2):117–31.
8. Gerada C, Galea M, Kladas A. Electrical machines for aerospace applications. *2015 IEEE workshop on electrical machines design, control and diagnosis (WEMDCD)*; 2015. p. 79–84.
9. Giangrande P, Galassini A, Papadopoulos S, et al. Considerations on the development of an electric drive for a secondary flight control electromechanical actuator. *IEEE Trans Ind Appl* 2019;**55**(4):3544–54.
10. Morvan H. The IAT: Enabling aerospace research and innovation for the aerospace sector in the UK; 2015. [cited 2022 Jan 1]. Available from: https://blogs.nottingham.ac.uk/aerospace/2015/08/17/the-iat-enabling-aerospace-research-and-innovation-for-the-aeropsace-sector-in-the-uk/.
11. Giangrande P, Madonna V, Sala G, et al. Design and testing of PMSM for aerospace EMA applications. *IECON 2018-44th annual conference of the IEEE industrial electronics society* 2018;2038–43.
12. Christmann M, Seemann S, Janker P. Innovative approaches to electromechanical flight control actuators and systems. *Recent Adv Aerospace Actuat Syst Compon* 2010;5–7.
13. Guillod T, Papamanolis P, Kolar JW. Artificial neural network (ANN) based fast and accurate inductor modeling and design. *IEEE Open J Power Electron* 2020;**1**:284–99.

14. Zhao S, Blaabjerg F, Wang H. An overview of artificial intelligence applications for power electronics. *IEEE Trans Power Electron* 2020;**36**(4):4633–58.
15. Dragičević T, Wheeler P, Blaabjerg F. Artificial intelligence aided automated design for reliability of power electronic systems. *IEEE Trans Power Electron* 2018;**34**(8):7161–71.
16. Hornik K, Stinchcombe M, White H. Multilayer feedforward networks are universal approximators. *Neural Networks* 1989;**2**(5):359–66.
17. Zhang S. Artificial intelligence in electric machine drives: Advances and trends. arXiv preprint:211005403, 2021.
18. Xu Y, Zhang Z, Yu L, et al. Indirect measurement and extreme learning machine based modelling for flux linkage of doubly salient electromagnetic machine. *IET Electr Power Appl* 2018;**12**(5):643–50.
19. Bramerdorfer G, Zăvoianu AC. Surrogate-based multi-objective optimization of electrical machine designs facilitating tolerance analysis. *IEEE Trans Magn* 2017;**53**(8):1–11.
20. Krasopoulos CT, Beniakar ME, Kladas AG. Robust optimization of high-speed PM motor design. *IEEE Trans Magn* 2017;**53**(6):1–4.
21. Gao Y, Yang T, Bozhko S, et al. Neural network aided PMSM multi-objective design and optimization for more-electric aircraft applications. *Chin J Aeronaut* 2021;**35**(10):233–46.
22. Gao Y, Yang T, Bozhko S, et al. Filter design and optimization of electromechanical actuation systems using search and surrogate algorithms for more-electric aircraft applications. *IEEE Trans Transport Electrif* 2020;**6**(4):1434–47.
23. Gao Y, Yang T, Dragičević T, et al. Optimal filter design for power converters regulated by FCS-MPC in the MEA. *IEEE Trans Power Electron* 2020;**36**(3):3258–68.
24. Sareni B, Regnier J, Roboam X. *Recombination and self-adaptation in multi-objective genetic algorithms. International conference on artificial evolution (Evolution Artificielle)*. Berlin: Springer; 2003. p. 115–26.
25. Sarigiannidis AG, Beniakar ME, Kakosimos PE, et al. Fault tolerant design of fractional slot winding permanent magnet aerospace actuator. *IEEE Trans Transport Electrif* 2016;**2**(3):380–90.
26. Burke EK, Burke EK, Kendall G, et al. *Search methodologies: Introductory tutorials in optimization and decision support techniques*. Berlin: Springer; 2014.
27. Sarkheyli A, Bagheri A, Ghorbani-Vaghei B, et al. Using an effective tabu search in interactive resources scheduling problem for LEO satellites missions. *Aerosp Sci Technol* 2013;**29**(1):287–95.
28. Dragičević T, Novak M. Weighting factor design in model predictive control of power electronic converters: An artificial neural network approach. *IEEE Trans Industr Electron* 2018;**66**(11):8870–80.
29. Li Z, Outbib R, Giurgea S, et al. Online implementation of SVM based fault diagnosis strategy for PEMFC systems. *Appl Energy* 2016;**164**:284–93.
30. Liu J, Li Q, Chen W, et al. A fast fault diagnosis method of the PEMFC system based on extreme learning machine and Dempster-Shafer evidence theory. *IEEE Trans Transport Electrif* 2018;**5**(1):271–84.
31. Seada H, Deb K. U-NSGA-III: A unified evolutionary algorithm for single, multiple, and many-objective optimization[Internet]. 2014. Available from: https://www.egr.msu.edu/kdeb/papers/c2014022.pdf.
32. Chen Y, Li L, Xiao J, et al. Particle swarm optimizer with crossover operation. *Eng Appl Artif Intell* 2018;**70**:159–69.
33. Li Z, Gao Y, Zhang X, et al. A model-data-hybrid-driven diagnosis method for open-switch faults in power converters. *IEEE Trans Power Electron* 2021;**36**(5):4965–70.
34. Norouzi R, Kosari A, Sabour MH. Real time estimation of impaired aircraft flight envelope using feedforward neural networks. *Aerosp Sci Technol* 2019;**90**:434–51.
35. Thirumalainambi R, Bardina J. Training data requirement for a neural network to predict aerodynamic coefficients. *Independent component analyses, wavelets, and neural networks. SPIE* 2003;92–103.
36. Xu Z, Gao Y, Wang X, et al. Surrogate thermal model for power electronic modules using artificial neural network. *IECON 2019-45th annual conference of the. IEEE industrial electronics society* 2019;3160–5.
37. Gao Y, Yang T, Wang X, et al. Machine learning based correction model in PMSM power loss estimation for more-electric aircraft applications. *2020 23rd international conference on electrical machines and systems (ICEMS)*. 2020. p. 1940–4.
38. Ganesh S. What's the role of weights and bias in a neural network? [Internet]. 2020. [cited 2022 Jan 1]. Available from: https://towardsdatascience.com/whats-the-role-of-weights-and-bias-in-a-neural-network-4cf7e9888a0f.
39. Sudhoff SD. *Power magnetic devices: A multi-objective design approach*. New York: John Wiley & Sons; 2021.
40. Cheong B. Advanced modelling and evolutionary multiobjective optimization of motor drive for aerospace applications[dissertation]. Nottingham: University of Nottingham; 2019.
41. MathWorks. Support documentation, particleswarm[Internet]. 2021. [cited 2022 Jan 1]. Available from: https://uk.mathworks.com/help/gads/particleswarm.html.
42. MathWorks. Support documentation, trainbr[Internet]. 2021. [cited 2022 Jan 1]. Available from: https://uk.mathworks.com/help/deeplearning/ref/trainbr.html.
43. Cheong B, Giangrande P, Zhang X, et al. Fast and accurate multi-physics model for optimization-based design of VSBBC. *IECON 2019-45th annual conference of the. IEEE industrial electronics society* 2019;1762–7.
44. Lin PY, Lai YS. Voltage control technique for the extension of DC-link voltage utilization of finite-speed SPMSM drives. *IEEE Trans Ind Electron* 2011;**59**(9):3392–402.
45. Lipo TA. *Introduction to AC machine design*. New York: John Wiley & Sons; 2017.
46. Cheong B, Giangrande P, Zhang X, et al. System-Level motor drive modelling for optimization-based designs. *2019 21st European conference on power electronics and applications (EPE'19 ECCE Europe)*. 2019. p. P-1.
47. Ounis H, Sareni B, Roboam X, et al. Multi-level integrated optimal design for power systems of more electric aircraft. *Mathe Comput Simul* 2016;**130**:223–35.