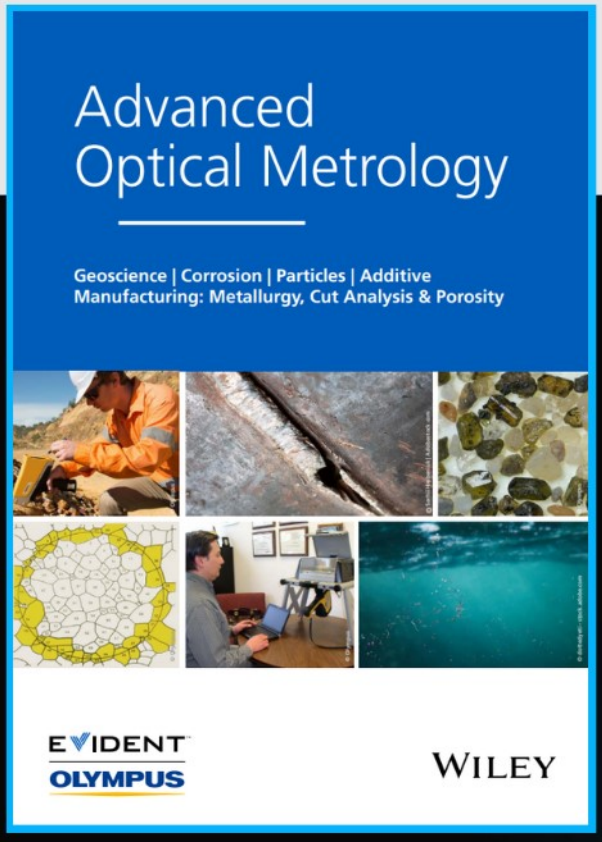
# Further investigation of convolutional neural networks applied in computational electromagnetism under physics-informed consideration

Ruohan Gong, Zuqi Tang

# 2nd Advanced Optical Metrology Compendium




**The latest eBook from Advanced Optical Metrology. Download for free.**

This compendium includes a collection of optical metrology papers, a repository of teaching materials, and instructions on how to publish scientific achievements.

With the aim of improving communication between fundamental research and industrial applications in the field of optical metrology we have collected and organized existing information and made it more accessible and useful for researchers and practitioners.

EVIDENT
OLYMPUS

WILEY

Received: 16 January 2022 | Revised: 27 February 2022 | Accepted: 28 February 2022

DOI: 10.1049/elp2.12183

IET Electric Power Applications

The Institution of Engineering and Technology WILEY

ORIGINAL RESEARCH

# Further investigation of convolutional neural networks applied in computational electromagnetism under physics-informed consideration

**Ruohan Gong** | **Zuqi Tang**

Univ. Lille, Arts et Metiers Institute of Technology, Centrale Lille, Junia, ULR 2697-L2EP, Lille, France

**Correspondence**

Zuqi Tang, Univ. Lille, Arts et Metiers Institute of Technology, Centrale Lille, Junia, ULR 2697-L2EP, L2EP - Bâtiment ESPRIT - 2e étage bureau S239, Avenue Henri Poincaré, Lille, 59655 Villeneuve d'Ascq, France.
Email: Zuqi.Tang@univ-lille.fr

**Abstract**

Convolutional neural networks (CNN) have shown great potentials and have been proven to be an effective tool for some image-based deep learning tasks in the field of computational electromagnetism (CEM). In this work, an energy-based physics-informed neural network (EPINN) is proposed for low-frequency electromagnetic computation. Two different physics-informed loss functions are designed. To help the network focus on the region of interest instead of computing the whole domain on average, the magnetic energy norm error loss function is proposed. Besides, the methodology of energy minimization is integrated into the CNN by introducing the magnetic energy error loss function. It is observed that the introduction of the physics-informed loss functions improved the accuracy of the network with the same architecture and database. Meanwhile, these changes also cause the network to be more sensitive to some hyperparameters and makes the training process oscillate or even diverge. To address this issue, the sensitivity of the network hyperparameters for both physics-informed loss functions are further investigated. Numerical experiments demonstrate that the proposed approaches have good accuracy and efficiency with fine-tuned hyperparameters. Furthermore, the post-test illustrates that the EPINN has excellent interpolation performance and can obtain good extrapolation results under certain restrictions.

**KEYWORDS**

computational electromagnetics, finite element analysis, learning (artificial intelligence), numerical analysis

## 1 | INTRODUCTION

Computational electromagnetism (CEM) is an important part of the electrical engineering domain. A precise field solution is necessary for the design and analysis of electromagnetic devices [1, 2]. This problem can be described by Maxwell's equations and calculated by numerical methods such as the finite element method (FEM). However, employing FEM for modelling and simulation about an electrical device can still be a computationally expensive process, especially when the studying object is geometrically complex or when there are some material non-linearities involved in the computation. The computational burden grows exponentially with the increase in degrees of freedom. To solve this problem, some surrogate approaches such as reduced-order modelling have been developed to accelerate the computation and reduce computation time [3, 4]. However, due to several limitations, these approaches are usually specific to a fixed problem and can only describe systems with a few parameters. One of the biggest limitations is that it is very difficult for these conventional surrogate models to handle geometric variables or some other variables that have significant nonlinearity, especially when these variables vary over a wide range. As a result, such surrogate models are limited to solving complex geometric feature parameterization problems in electromagnetic device design.

On the other hand, with the development of the graphics processing unit hardware and algorithm efficiency, deep learning (DL) is poised to be a very powerful tool that can

significantly increase our ability to conduct scientific research [5] and has demonstrated good potentials for the application in the field of computational electromagnetics [6]. DL has been successfully applied in the design and optimization of electrical devices such as motors [7], transformers [8, 9] and antennas [10], with satisfactory results. Among the many DL networks, in particular, the convolutional neural network (CNN), which is a type of network that automatically detects important features without any human supervision, has been widely used and some state-of-the-art performances acheived. Bayesian CNN is adopted to test the magnetic field computation of electromagnetic systems, as recently reported in [6]. CNN has also been applied to fault detection and diagnosis in induction motors based on electromagnetic signals [11, 12]. In addition, CNN has been applied to topology optimization and efficiency evaluation [13]. In [14], a deep residual convolutional and recurrent network is adopted to estimate the temperature inside the motor. Although DL has achieved many satisfactory results, the requirement for big data has greatly limited its further application in electrical engineering, where the data generally comes from complex experiments or expensive computations. It does not make too much sense for the DL application if its performance heavily relies on a large number of training samples because it costs lots of experiments or computation just to prepare these datasets, let alone the training and tuning of the DL network itself. To solve this problem, in our previous research, we have investigated the feasibility and efficiency of CNN U-net to the magneto-thermal coupling problem [15]. It is observed that due to its excellent feature extraction capability, the CNN U-net can be effectively trained with a small dataset. Besides, to further improve the efficiency of DL, we optimize the sample selection strategy using the greedy algorithm [16]. Take the FEM computation results as the ground truth, the CNN can be applied in the training-prediction manner: parts of FEM results can be adopted as the training data to train the DL network by minimizing the loss via gradient descent optimization algorithms, such as Adam [17] and Adadelta [18] until the loss is smaller than an expected threshold. Some other FEM results can be utilized to validate the feasibility and accuracy of the trained network. Once the DL network is effectively trained and verified, rest of the samples can be predicted by the effectively trained DL model without further FEM computation considering the complex physical context implicit in the system.

So far DL has yielded some good performances in CEM, especially in some image-based tasks. Nevertheless, several new issues have arisen along with the application of DL. One of the limitations of the conventional image-based CNN model is the lack of physical constraints. In most practical applications of CNN, the constructed network can be thought of overlooking any possible underlying mathematical model expressing physical laws, and thus produces a model-ignorant algorithm because the network training process does not require any knowledge of the underlying mathematical model. During the DL network training procedure, loss functions are needed to indicate and monitor the iterative convergence process of the network, which is crucial for the network learning capacity. Generally, the two most widely used loss functions in the field of DL and computer vision are mean absolute error (MAE) and mean square error (MSE). But these two loss functions only describe the difference between the model output and the ground truth from a pixel perspective, ignoring some implicit relationships including physical laws. When we are dealing with some traditional computer vision tasks, such as distinguishing apples from pears [19], diagnosing damaged tissues in medicine images [20] or traffic recognition systems for autonomous vehicles on the highway [21], the research objects are generally images without hidden laws that can be formulated, in these cases the general loss function MAE or MSE can usually meet our needs.

Whereas, when we face some problems involving physical systems, including electromagnetic field problems, our goal is often not simply classification or identification, but probably some physical quantities or laws that are related to the appearance, such as the magnetic energy implied behind the magnetic field distribution. At this point, uniformly evaluating the output from a pixel perspective by using traditional MAE and MSE alone may give a fine result in terms of image pixels after the network converges, but it may not necessarily achieve a good prediction for the specific physical quantity we desired. In addition, the magnetic flux density obtained from FEM is based on the law of minimizing the error in the energy sense, so the value at one certain pixel does not make too much sense; the difference between the ground truth and the results obtained by DL should be estimated in a global energy sense, for instance, the magnetic energy norm.

The physics-informed neural network (PINN), as introduced in [22, 23], is an effective method to handle supervised learning tasks while respecting any given law of physics described by general nonlinear partial differential equations (PDEs). By penalizing deviations from the target value and properly weighting any given data, the physical rules and any other constraints can be seamlessly integrated into the loss function of the PINN. What's more, if some features of the PDE solutions are known a priori, it is also possible to encode them in the network architectures [24]. This allows the PINN to take advantage of some prior knowledge related to a certain physical system that has been overlooked in a conventional DL network, hence improving the performance and generalization of the DL network on this certain physical scenario.

The approach of embedding physical knowledge into the PINN is very flexible, we can either design a specialized architecture that implicitly enforces knowledge or impose such constraints in a soft manner by appropriately penalizing the loss function of PINN or a hybrid approach that mixed these two. This flexibility allows researchers to develop different variants of PINN to satisfy exactly the required constraints such as initial conditions [25, 26] or different types of boundary conditions [27, 28]. This is a fast-moving field, in a short period of time, lots of PINN variants have been proposed for different physical systems. These variants have been extensively applied to various engineering problems with different kinds of physical laws hidden behind. In [29], a

variational PINN is developed within the Petrov-Galerkin framework to lower the order of the differential operator to reduce the training cost. In addition, a fractional PINN is designed to solve space-time fractional advection-diffusion equations [30]. The Bayesian neural network combined with a PINN is proposed to solve forward and inverse nonlinear problems described by PDE and noisy data [31]. Another work employs the domain decomposition in the PINN framework and puts forward a conservative PINN to provide more flexibility in the multiscale problem [32, 33]. Besides, PINN is employed to solve the radiative transfer equations and the rigorous upper bounds on the generalization error of PINN approximating solutions for PDE have been presented in [34, 35]. Furthermore, PINNs have achieved some satisficing performance when facing heat transfer equations, Navier–Stokes equations in fluid mechanics, high speed fluid flow and solid mechanics [36, 37].

Although PINN has shown good potential in various fields, it still has certain limitations. One of the main limitations of PINN is the dependence on a large amount of database, especially for some over-parameterized PINN models [38, 39], a significant amount of data is often necessary to reinforce such physical knowledge and generate predictions that satisfy certain physical laws such as symmetries and conservation rules. In this case, an immediate difficulty relates to the cost of data acquisition, which for many applications in the physical and engineering sciences could be prohibitively large, as observational data may be generated via expensive experiments or large-scale computational models. Therefore, it is crucial to enhance the convergence and accuracy of the network without increasing a large number of samples. On the other side, while many good application results have been obtained from PINN applications; however, a lot of work stops at interpolation, and only a few research studies have investigated the extrapolation capability of PINN [31].

Based on a relatively small database, we applied an energy-based physics-informed neural network (EPINN) to solve the CEM problem with a wide range of geometrical parameters. The main contributions of this work are as follows:

1. To the best of our knowledge, the application of PINN has not been applied to low-frequency CEM. Taking a typical transformer magnetic problem as an example, take the FEM computation results as ground truths, the EPINN network is utilized for the evaluation of the magnetic field of the transformer in a supervised manner. Physical information is integrated into the DL network by modifying the loss function. According to the characteristics of the electromagnetic field and inspired by our previous work on the error estimation for the magnetostatic problem [40, 41], two physics-informed loss functions based on the magnetic field energy are put forwarded for tasks with different requirements, that is the magnetic energy norm error (MENE) and the magnetic energy error (MEE). Among them, the MENE can be seen as a specific MSE that is weighted according to the material permeability, which can help the EPINN to concentrate on the learning objective which has higher energy density, that is, the quantity of interest (QOI). As for the MEE, it is reinforced with the energy minimization principle in the DL loss function, considering that the principle of FEM computation is energy minimization, it is a reasonable strategy to evaluate the magnetic field distribution results via magnetic field energy instead of pure pixel error.
2. We discuss the implementation of various types of loss functions. To investigate the effect of introducing the two physics-informed loss functions, this paper compares the EPINN with the traditional CNN using MAE and MSE. While keeping the network architecture, configurations, and the database unchanged, these four loss functions are respectively adopted to train the network. The numerical experiments demonstrate that when compared to other loss functions, the physics-informed loss function developed for a specific task can achieve better accuracy on that task. With the introduction of the physics-informed loss functions, the neural network is more sensitive to the structural changes in the windings because the magnetic energy is mainly concentrated in the windings and the nearby air. This can help the DL network to better concentrate on the QOI instead of taking the same weight over the whole image only from a pixel-wise point of view.
3. We conduct a sensitivity analysis of EPINN to explore the effect of hyperparameters on the model performance. The physics-informed loss function is a double-edged sword. While it improves the accuracy of the converged network under the same database, it is also observed that it increases the sensitivity of the DL network to the input information and some corresponding network hyperparameters, especially, those that control the generation of Gaussian distribution from the labelled geometric information. Inappropriate hyperparameters can affect the learning accuracy or even lead to divergence. To address this issue, these network hyperparameters that become more sensitive due to changing the loss function are investigated with the two physical-informed loss functions separately. In our previous work, from the perspective of the network architecture, some investigation about the hyperparameter sensitivity analysis has been performed for the magneto-thermal coupled problem [15]. Based on the hyperparameter investigation in our previous work, this paper further extends the study of two other hyperparameters that affect the network input Gaussian distribution image: the standard deviation (STD) and the normalization range. With proper hyperparameter configurations, the accuracy and efficiency of the EPINN have been illustrated through numerical experiments.
4. This work not only tests the interpolation ability of the proposed EPINN model but also examines the extrapolation performance: after the effective training process, the trained model was utilized to predict the sample that outside the training samples, whose input label variables vary outside the training sample. Numerical experiment shows that the EPINN has great potentials in generalization

capability since it can achieve good results in interpolation. As for extrapolation, good performance can be obtained for these samples that are adjacent to the training samples. The effective extrapolation range of the network is investigated.

The rest of the paper is organized as follows: In Section 2, we presented the problem setup. Section 3 gives the methodology of the proposed method including a brief introduction about the physical-informed loss functions. Numerical experiments are performed in Section 4, where various comparisons are made between the CNN and EPINN methods. Finally, in Section 5, we summarize our findings.

## 2 | CONSIDERED MAGNETOSTATIC PROBLEM

Power transformers are indispensable parts of the modern power grid. They transform power from one circuit to another by changing the current without changing the frequency, which can significantly reduce transmission line losses caused by the resistance. When a transformer is operating, current flows in the windings, which generates a magnetic field. Most of the magnetic flux passes through the core, but some of it inevitably passes through the air, which is the leakage magnetic flux. These leakage magnetic fluxes cause eddy current losses in the wires and stray losses in other metal parts such as the oil tank, causing local overheating which will damage insulation or even affect the safe operation and the service life. Power transformer generally operates at 50 Hz, so the low-frequency magnetic problem can be adopted to describe its internal magnetic field distribution, which can be described by the following equation:

$$\nabla \times \mathbf{H} = \mathbf{J}_s \tag{1}$$

$$\nabla \cdot \mathbf{B} = 0 \tag{2}$$

$$\mathbf{B} = \mu(\mathbf{H})\mathbf{H} \tag{3}$$

where $\mu$ denotes the magnetic permeability, $\mathbf{H}$ represents the magnetic field strength, $\mathbf{B}$ stands for the magnetic flux density, $\mathbf{J}_s$ is the current density vector.

This work concentrated on the magnetic flux density distribution inside the transformer with different geometries. For the present example, a S13-M-100 kVA/10 kV transformer has been chosen to carry out the numerical experiments, as demonstrated in Figure 1. To simplify the computation, only one phase of windings and iron core are taken into consideration and the windings are simplified into 3 separate discs. In addition, the FEM model is further simplified to an axisymmetric model based on the structure of the winding, as shown in Figure 2. The leakage magnetic flux distribution is given by the axisymmetric cross-sectional image. The phase of the current in high-voltage (HV) winding and low-voltage (LV) winding is opposite. The coils are made of copper and the conductivity is set to be 5.7e7 S/m. The iron core is composed of B27R90 silicon steel, which is considered to be nonlinear.

**FIGURE 1** S13-M-100 kVA/10 kV transformer

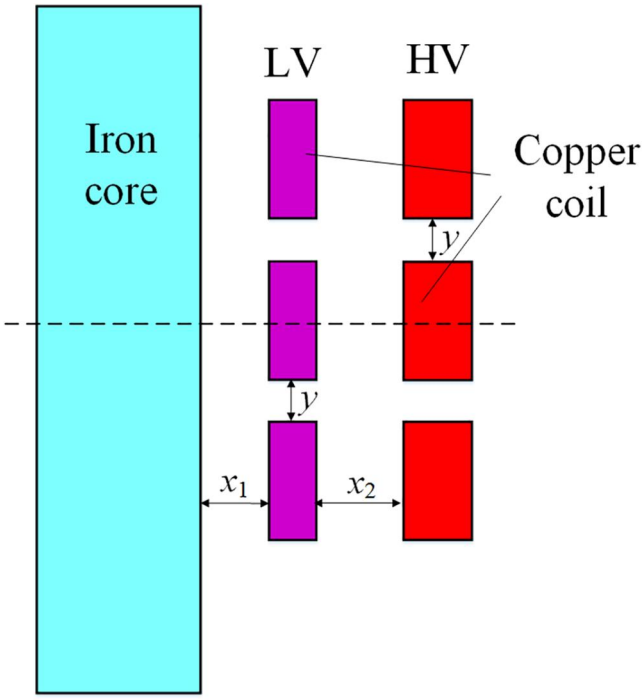



**FIGURE 2** Geometry of the considered 2D transformer

The magnetization curve of the B27R90 silicon steel is illustrated in Figure 3 [42].

Different magnetic field distributions are obtained by changing the geometry of the windings. The dimension changes are demonstrated in Table 1. As an example, three geometric variables of the transformer winding are taken into consideration: $x_1$, $x_2$ and $y$, which, respectively, represent the distance between the LV winding and the iron core, the distance between the LV winding and the HV winding and the gap between the copper coils which are the same in the HV and LV winding. Multiple samples can be generated by automation via geometric variation. Each variable takes 10 values uniformly across the range of values, generating a total of 1000 samples. These samples will constitute the database for the DL process in the following sections. All the FEM computations shown are realized by the free scientific computing software Freefem++ [43].

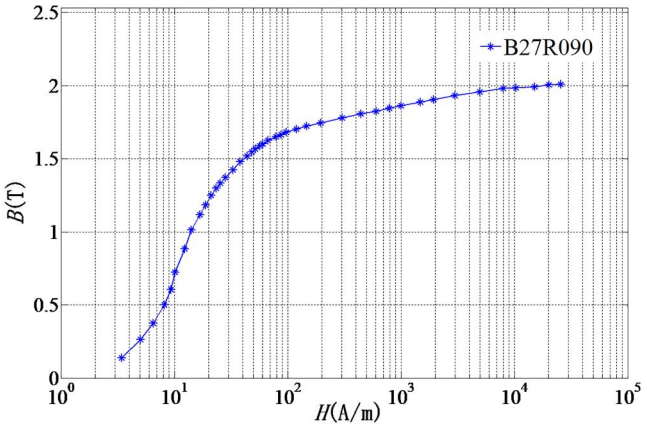



**FIGURE 3** Magnetization curve of the B27R90 silicon steel

**TABLE 1** The range of variation of geometric variables

| Parameters | Min value (mm) | Max value (mm) |
|---|---|---|
| $x_1$ | 10 | 60 |
| $x_2$ | 20 | 50 |
| $Y$ | 15 | 65 |

## 3 | METHODOLOGY

### 3.1 | The EPINN

With ground truth data obtained from FEM being represented as magnetic flux density distributions, EPINN, a CNN with physics embedded into the loss function is utilized to extract hidden features and is trained in a supervised manner. Adopting the aforementioned three geometric variables as labels, the relationship between the magnetic field distribution and corresponding geometric parameters is established via DL. Using part of the database to train the network, once the network converges, the effectively trained EPINN can be used to predict the magnetic field distribution with other geometric parameters without further FEM computations. The schematic diagram of the presented EPINN is shown in Figure 4. First and foremost, we establish the physical model according to the research objectives, as shown in Figure 4b, after setting the material parameters, boundary conditions and initial conditions, etc., we use the geometric parameters as variables and perform FEM computation to obtain the corresponding magnetic field distribution. These FEM computation results are displayed via 256 × 256 × 3 red-green-blue (RGB) images and employed as the database, as shown in Figure 4a. Samples are labelled according to the corresponding geometric parameters $x_1$, $x_2$, and $y$. Let us denote $\tilde{z}_1$, $\tilde{z}_2$, $\tilde{z}_3$ the vector composed by the normalization data for different value issues from the labels $x_1$, $x_2$, and $y$, the Gaussian distributions are generated from these geometric variables and used as input for the DL network, as shown in Figure 4d. The input consists of three layers of mutually independent Gaussian distributions, and each layer is controlled by parts of the geometric variables, denoted by $f_i(1 \le i \le 3)$, where

$$f_i(z_j, z_k) = \exp\left(-\frac{\left(z_j - \mu_{\tilde{z}_j}\right)^2 + \left(z_k - \mu_{\tilde{z}_k}\right)^2}{2\sigma^2}\right) \quad (4)$$

with $1 \le j < k \le 3$, where $j \ne i$ and $k \ne i$, $\mu_{\tilde{z}_i}$ is the average value of all the components for $\tilde{z}_i(1 \le i \le 3)$. Taking the corresponding magnetic field distribution with different geometric parameters as the output (as shown in Figure 4f), the EPINN is adopted to establish the hidden relationship between the input and output.

The architecture and the baseline configuration of the DL network applied are shown in Figure 4e and Table 2. The main network framework is described in detail in [15]. The architecture is made up of successive encoder layers and decoder layers, and each layer includes an activation function, a batch normalization, and a convolution or up-convolution. Furthermore, a dropout layer is employed between the encoder and decoder as a regularization technique to prevent the network from overfitting. In our previous studies, from the perspective of the network architecture, the influences of hyperparameters on the training curve and convergence are investigated, with four most dominant factors: training sample size, learning rate, batch size, and optimization algorithm which detailed discussed in detail for the low frequency magneto-thermal coupled problem. Adopting the network architecture and the baseline set of hyperparameters obtained for the magnetic field evaluation in [15], the loss function is adjusted so that some prior physical information and constraints can be embedded into the EPINN thus helping the network to achieve better learning performance. Meanwhile, the physics-informed loss function also makes the DL network more sensitive to the input and relevant hyperparameters and easier to diverge. Two hyperparameters that affect the input Gaussian distribution are investigated in Section 4.1, namely the STD and the normalization range. Finally, the loss functions embedded with physical information based on the research model, as shown in Figure 4c, are used to evaluate the learning performance of the EPINN in the post-test.

### 3.2 | The training and post-test procedure

The training and post-test procedure of the proposed approach are illustrated in Figure 5. One thousand database samples are generated and indexed as #1 to #1000 by incremental iteration of the three geometric parameters one by one in the order $x_1$, $x_2$, $y$. All samples are labelled according to their geometric parameters and divided into 10 groups according to the single digit of the index number. The 10 groups are indexed as #0 to #9, with 100 samples in each group. The 10 groups of samples are divided into three categories: training samples, validation samples and test samples to participate in the network training and post-test

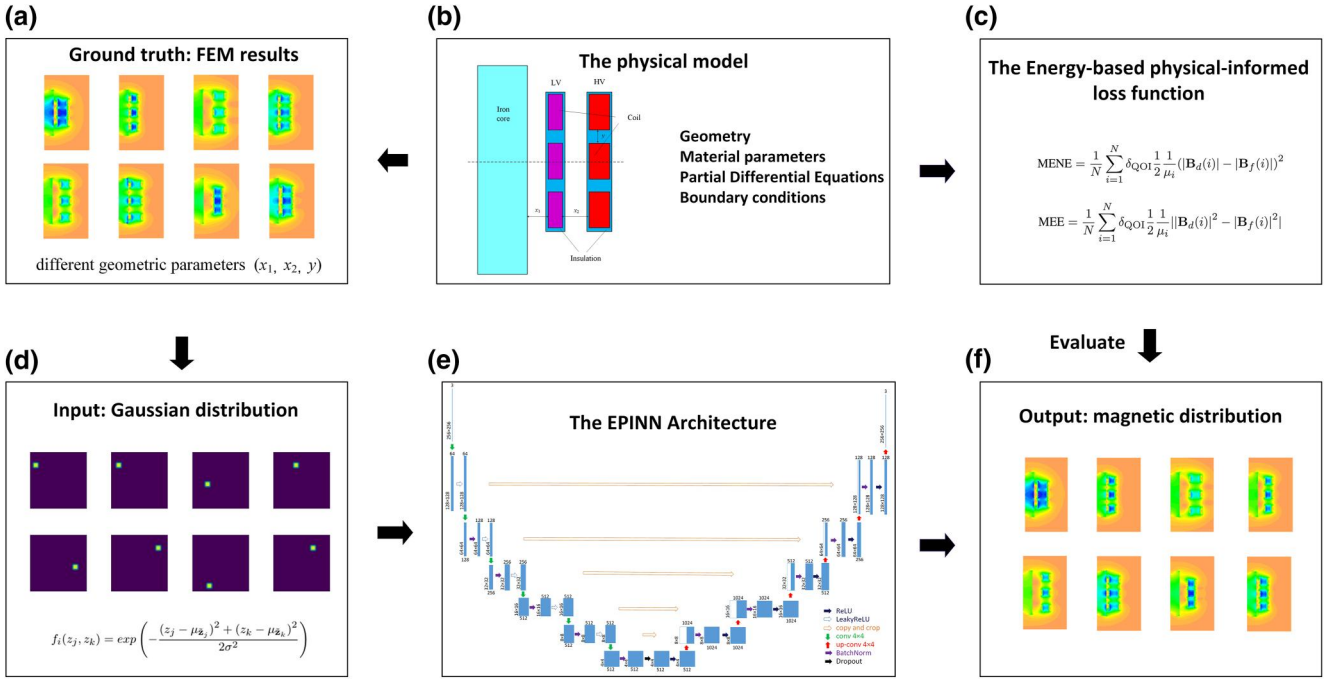



**FIGURE 4** The schematic diagram of the energy-based physics-informed neural network (EPINN). (a) Grounth truth: FEM computation results. (b) Physical model. (c) The energy-based physical-informed loss functions. (d) Network input: Gaussian distribution. (e) The architecture of EPINN. (f) Network output: magnetic distribution

**TABLE 2** Baseline configuration of the EPINN hyperparameters

| Hyperparameters | Value |
|---|---|
| Index of training samples | 400 (#2, #4, #6, #8) |
| Index of validation samples | 100 (#7) |
| Learning rate | 1e−4 |
| Batch size | 4 |
| Optimization algorithms | Adam |
| $\beta_1$ | 0.9 |

Abbreviation: EPINN, energy-based physics-informed neural network.

process. To begin with, training samples are employed to train the network. The EPINN is trained by minimizing the residual of the evaluation losses, and the weight of the network is iteratively updated according to the gradient descent algorithm and backpropagation of the error.

After the trained network converges to the training samples, validation samples are introduced to verify the trained network to see if it meets the error requirement on validation samples. If not, continue to update the network with training samples until the trained model converges in the validation sample. Test samples are not involved in the training procedure of the network, but only introduced to validate the effectiveness and accuracy of the trained network in the post-test. It should be mentioned here that the capability of the interpolation is evaluated when the post-test samples are inside the training samples and validation samples. On the other hand, the capacity of extrapolation is assessed when the post-test samples are different from those who participated in the previous training phase.

## 3.3 | The loss evaluator

DL networks are trained to solve supervised learning tasks by minimizing the residual of the evaluation losses. In the process of training a feedforward neural network, the iteration is generally based on the gradient descent and backpropagation algorithm: the backpropagation algorithm works by computing the gradient of the loss function with respect to each weight by the chain rule, iterating backward from the last layer and update the weights along with the inverse of the corresponding error gradient to minimize the loss. Therefore, choosing the proper loss function is crucial for DL networks to have a good performance on a certain task.

MAE and MSE are two of the most common evaluators employed to measure accuracy for computer vision tasks. The MAE represents the average of the absolute difference between the actual and predicted values in the dataset. For the magnetic flux distribution prediction, the target is the magnetic flux density:

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}||\mathbf{B}_{\mathrm{dl}}(i)| - |\mathbf{B}_{\mathrm{fem}}(i)|| \tag{5}$$

where $N$ represents the number of pixels in an image, which is $256 \times 256 = 65{,}536$ in our case, $\mathbf{B}_{\mathrm{dl}}$ stands for the DL prediction result of the magnetic flux density distribution, $\mathbf{B}_{\mathrm{fem}}$ is

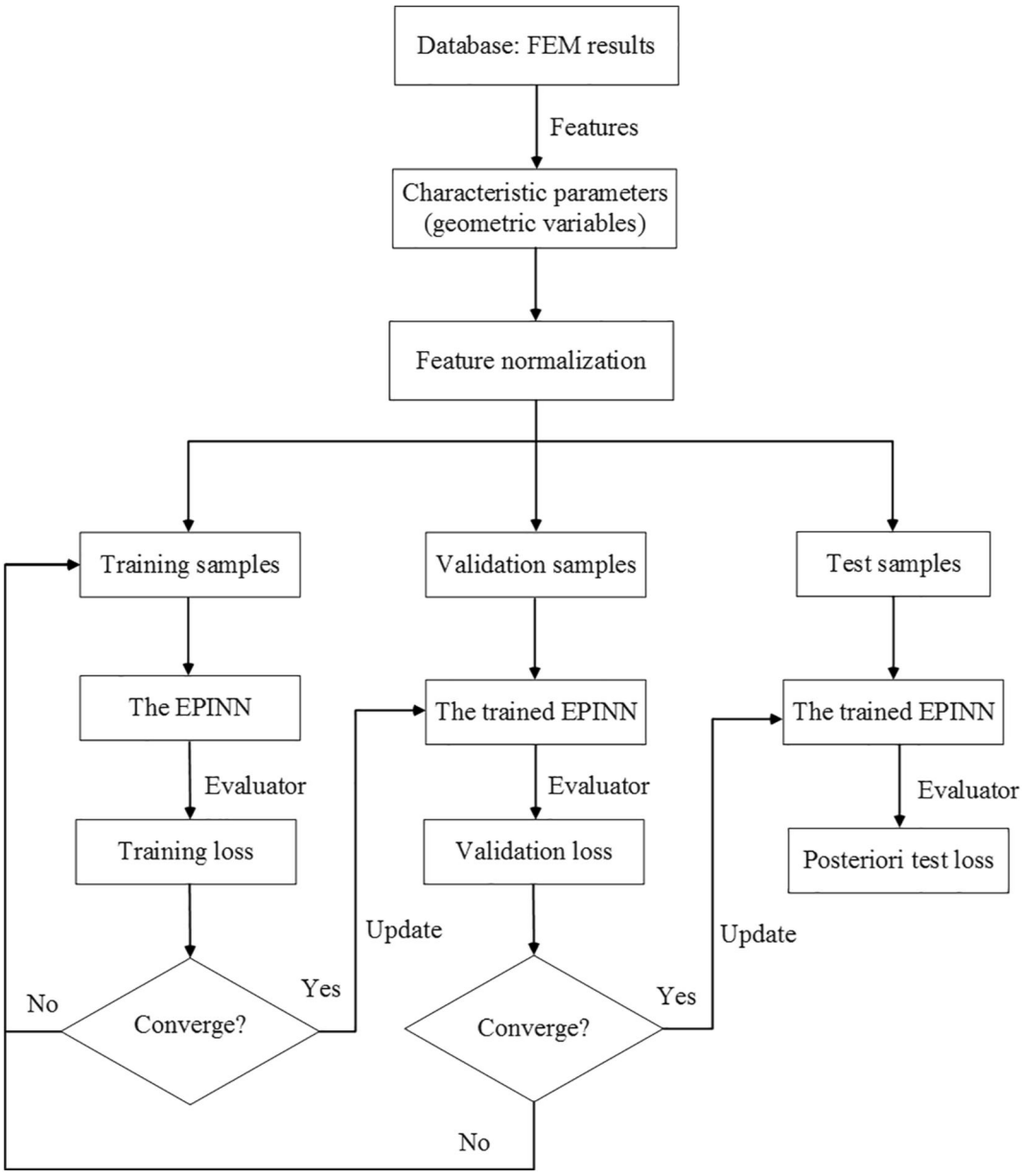



**FIGURE 5** The flowchart of the training and test processes of the energy-based physics-informed neural network

the magnetic flux density distribution obtained from FEM. It measures the average of the residuals in the dataset. As a contrast, the MSE represents the average of the squared difference between the original and predicted values in the dataset:

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N} ||\mathbf{B}_{\mathrm{dl}}(i)| - |\mathbf{B}_{\mathrm{fem}}(i)||^2 \tag{6}$$

It measures the variance of the residuals. These two loss functions have their own advantages and disadvantages: MSE penalizes the large prediction errors compare to MAE, but MAE is more robust to data with outliers. These two complement each other and have conquered many tasks in the field of machine learning and computer vision. Nevertheless, for some problems involving physical systems, they have the common insurmountable drawback that the error can only be calculated from the perspective of the image pixels on average, without focussing on the QOI or taking into account the physical context implied behind this consideration.

In this case, there is a vast amount of prior knowledge that are not being utilized in the above DL practice. These prior information can act as a regularization agent that constrains the solutions. In return, encoding such structured information into a learning algorithm results in amplifying the information content of the data that the algorithm sees, enabling it to quickly steer itself towards the right solution and generalize well in the post-test. Take the considered magnetostatic problem as an example, the ground truths we obtain from FEM are computed following the magnetic energy

minimization principle. For a magnetostatic system, the magnetic field energy $E$ stored in a magnetic field, the energy per unit in a region of permeability $\mu_i$ containing the magnetic field $\mathbf{B}$ is:

$$\mathrm{E} = \sum_{i=1}^{N} \frac{1}{2} \frac{|\mathbf{B}(i)|^2}{\mu_i} \tag{7}$$

where $\mu_i$ denotes the magnetic permeability in the $i$th pixel.

Therefore, naturally, when it comes to the magnetic problem, researchers can be more interested in magnetic energy than magnetic flux density distribution. In this situation, utilizing loss functions that are only considered from the perspective of pixels, we can only measure the difference in magnetic flux density distribution between the predicted results and the dataset, but ignore the meaning of the FEM based on the principle of minimum magnetic energy. To enforce this point, inspired by our previous work about error estimates in the low-frequency problem [40, 41], we try to introduce the energy sense into the EPINN network for the prediction of the magnetic field distributions by modifying the loss function. In addition, when modelling a physical system, instead of the whole computation domain, researchers are tended to be more interested in a particular region based on some prior information or experience. For instance, when analysing the transformer magnetic leakage problem and the corresponding stray losses, we are most concerned with the leakage magnetic flux in the windings and nearby air as a QOI, which dominantly affects the eddy current losses, and less interested in the flux distribution inside the core. In this case, using MAE and MSE, we can only measure the difference in magnetic flux density distribution between the predicted results and the dataset on average, and cannot pay more attention to the area of interest.

Hence, we introduce two loss evaluators that informed with physical laws and the idea of QOI, the first one being the MENE loss function:

$$\mathrm{MENE} = \frac{1}{N} \sum_{i=1}^{N} \delta_{\mathrm{QOI}} \frac{1}{2} \frac{1}{\mu_i} ||\mathbf{B}_{\mathrm{dl}}(i)| - |\mathbf{B}_{\mathrm{fem}}(i)||^2 \tag{8}$$

where $\delta_{\mathrm{QOI}}$ is the characteristic function of the QOI. As the name implies, the loss function is a norm designed from the perspective of magnetic energy. It determines the weights assigned to each region of the magnetic flux density distribution through the magnetic permeability, which makes the magnetic flux distribution of the winding and the surrounding air take up a larger proportion of the loss function. In addition, different weights can be attached to the special QOI according to specific requirements, as illustrated in [41]. The MENE can be seen as a variant of MSE after weighting by material parameters and the QOI characterizes the function.

The MENE can help the DL network to concentrate more on the region of interest by introducing some prior physical information. However, the evaluation of loss is still from the perspective of pixels. As we mentioned before, when it comes to magnetic field analysis, since the magnetic flux density computed by FEM is based on the minimization of magnetic energy, the value within a specific pixel alone does not make too much sense to evaluate the accuracy of the entire FEM magnetic field computation. Instead, the difference between the ground truth and the results obtained by DL should be interpreted in a global energy sense. In this situation, utilizing loss functions that are only considered from the perspective of pixels, we can only measure the difference in magnetic flux density distribution between the predicted results and the dataset, but ignore the meaning of the FEM based on the principle of minimum magnetic energy. To address this issue, we propose another physical-informed loss function from the perspective of magnetic field energy, that is, the MEE loss function:

$$\mathrm{MEE} = \frac{1}{N} \sum_{i=1}^{N} \delta_{\mathrm{QOI}} \frac{1}{2} \frac{1}{\mu_i} ||\mathbf{B}_{\mathrm{dl}}(i)|^2 - |\mathbf{B}_{\mathrm{fem}}(i)|^2| \tag{9}$$

The EPINN with MEE can naturally encode some underlying physical laws as prior information into the DL network. In this way, during the network training procedure, the target is no longer simply the magnetic field distribution, but the implied magnetic field energy.

The MENE and MEE are adopted as the loss function for the hyperparameter investigation in Section 4.1. With fine-tuned hyperparameters, the EPINN can be effectively trained and the converged model can be used to predict the magnetic field distribution. As a comparison, with the same network architecture and hyperparameter configuration, MAE and MSE are respectively utilized to train the DL network from the same database with the same training-validation strategy. The performances of the magnetic field predictions of the networks trained by these four loss functions are respectively evaluated by the MENE and MEE loss function in the post-test.

## 4 | NUMERICAL EXPERIMENTS AND DISCUSSION

As mentioned before, there are pros and cons to introducing the physics-informed loss functions into the EPINN. On the one hand, physics-informed learning has the advantage of strong generalization in the small data regime: by embedding physics, the EPINN is more effectively concentrated on the regions or physical quantities that we are interested in, thus improving the accuracy of the converged DL network with the same database. On the other hand, it is observed that it can make the DL network more sensitive to the input data and some of the relevant network hyperparameters. As discussed in [15], one of the necessary premises for a DL network to achieve good accuracy and reproducibility is a proper set of hyperparameters. Before comparing the learning performance of the EPINN and traditional CNN, we must first ensure its convergence. In all the following numerical experiments, the $\delta_{\mathrm{QOI}}$ is set to be 1 which means the weights are assigned by the

material magnetic permeability, and it can also be defined as a special QOI according to different tasks.

## 4.1 | The investigation of hyperparameters

The network hyperparameters play a crucial role in the convergence of the DL network. To analyse the performance of the proposed EPINN approach, we shall discuss the effect of these hyperparameters on the predictive accuracy of the solution of the low-frequency magnetic problem. On the basis of the CNN U-net, from the perspective of network architecture, some of the most dominant hyperparameters have been investigated in our previous work. The baseline configuration of the EPINN hyperparameters is shown in Table 2.

On this basis, this article adjusts the DL network by modifying the loss function to introduce prior physical information. In order to focus on the QOI during the training process instead of uniformly obtaining the global error, we introduce the MENE. For the sake of considering the loss from the perspective of magnetic field energy, we adopt the MEE. Through numerical experiments, it is found that the sensitivity of the two hyperparameters is different from the previous work, that is, the two key hyperparameters that affect the Gaussian distribution generated from the sample geometric information. As illustrated in Figure 4, solving of the magnetic problem via DL has been transformed into a task for the image-to-image EPINN network to find the hidden relationship between the input image and the output image, that is, the relationship between the Gaussian distribution and the magnetic flux density distribution. Naturally, the generation process of the Gaussian distribution will also influence the learning performance of the DL network. The generation of Gaussian distribution is mainly controlled by the following two parameters: the STD and the normalization range. The former controls the size and gradient of bright spots on the Gaussian distribution and the latter determines the movement range and step size of the bright spot. Among them, the normalization range consists of two numbers that are often in the range between 0.0 and 1.0, which determines the upper and lower limits of the moving position of the bright spot in the Gaussian distribution map. Taking values of 0.0 and 1.0 means that the centre of the bright spot moves from the leftmost to the rightmost and from the top to the bottom of the picture in the 1000 Gaussian distribution input database.

Based on the network baseline configuration as shown in Table 2, these two hyperparameters are respectively investigated with the two proposed loss functions. For consistency, other hyperparameters of the network are fixed during the study of a particular hyperparameter. The training procedure has been described in detail in Section 3, where four groups of samples (400 samples) are adopted as the training sample and one group of samples is used as the validation sample. The training loss and validation loss are calculated separately at the end of each epoch until both losses are small than the expected threshold, and we consider that the DL network converges. Starting with the MENE, the training curves of EPINN with different STD and normalization ranges are demonstrated in Figure 6.

From Figure 6, we can find that STD has little effect on the training loss, but it has a greater impact on the validation loss. When the STD is greater than 10, the training performance of the EPINN is better. Considering that the training procedure is highly stochastic due to the variance caused by estimation via Adam, random dropout and random weight initialization. We replicated the training procedure 10 times and record the mean, minimum and maximum value of training loss and validation loss observed over 10 replications. These results about training loss and validation loss are demonstrated respectively in Tables 3 and 4.

The training loss is smaller and more stable than the validation loss. The difference of training loss between the maximum and minimum in these 10 replications is very small, which means the repeatability of training loss is very good. In contrast, validation loss is larger and the upper and lower limits fluctuate greatly, especially when the STD is smaller than 10. For instance, when STD is set to be 5, the maximum validation loss is 1.05e−3, while the minimum validation loss is 3.66e−4. This indicates that the network has oscillations during the training process. When STD takes a value between 10 and 100, the training process of the network is more stable and the training loss and validation loss perform more or less the same. To sum up, the optimal STD for magnetic field learning with EPINN with MENE is about 10–100. The normalization range is investigated using the same approach as STD. The training curves are illustrated in Figure 6a,b.

Figure 7 shows the training curve of EPINN with MENE with different normalization ranges. Similar to the results of STD, the normalization range has little influence on the training loss, and the training curve converges steadily for different values. On the other hand, the normalization range has a significant effect on the validation loss. Unsuitable values will not only cause convergence oscillations and affect the learning accuracy, but even lead to divergence. Once again, in order to prevent the randomness in the learning process from affecting our conclusion, the training procedure is repeated 10 times for each value. The average, maximum and minimum losses of these training results are recorded and shown in Tables 5 and 6, respectively, for the training loss and validation loss.

We can observe that the normalization range has a greater impact on the training procedure than STD. An inappropriate range of values will oscillate the training process and invalidate the model. To sum up, consistent with our previous analysis, for the EPINN with MENE, both STD and normalization range influence the convergence curve of the DL network. Among them, relatively speaking, the normalization range has a greater impact on network errors after convergence. When the STD value is too small, the network's training process suffers, and the proper STD for magnetic field learning with MENE is about 10–100. As for the normalization range, when they take a wide range of values and the bright spot in the Gaussian distribution map moves over a wide range, the network converges less well, and the approximate range of the good training performance is between 0.25 and 0.6. It needs to be mentioned here that the purpose of hyperparameter investigation is not to propose the optimal solution,

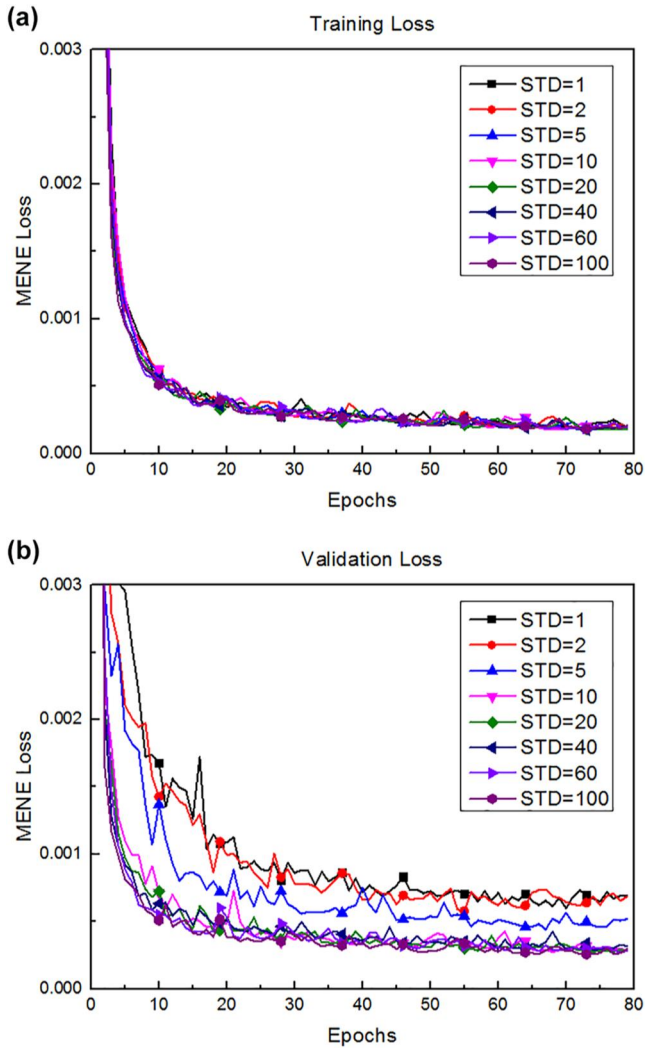



**FIGURE 6** Training curve with different standard deviation (magnetic energy norm error)

**TABLE 3** The MENE training loss of the EPINN with different STDs

| STD | Average | Minimum | Maximum |
|---|---|---|---|
| 1 | 1.80e−4 | 1.70e−4 | 1.88e−4 |
| 2 | 1.80e−4 | 1.70e−4 | 1.95e−4 |
| 5 | 1.78e−4 | 1.70e−4 | 1.90e−4 |
| 10 | 1.78e−4 | 1.72e−4 | 1.82e−4 |
| 20 | 1.77e−4 | 1.68e−4 | 1.88e−4 |
| 40 | 1.76e−4 | 1.68e−4 | 1.88e−4 |
| 60 | 1.76e−4 | 1.69e−4 | 1.83e−4 |
| 100 | 1.77e−4 | 1.68e−4 | 1.83e−4 |

Abbreviations: EPINN, energy-based physics-informed neural network; MENE, magnetic energy norm error; STD, standard deviation.

but to provide a "possible best" range for each hyperparameter through numerical experiments to ensure the effectiveness of the DL training.

**TABLE 4** The MENE validation loss of the EPINN with different STDs

| STD | Average | Minimum | Maximum |
|---|---|---|---|
| 1 | 6.93e−4 | 4.27e−4 | 8.85e−4 |
| 2 | 6.15e−4 | 5.10e−4 | 7.98e−4 |
| 5 | 5.40e−4 | 3.66e−4 | 1.05e−3 |
| 10 | 2.91e−4 | 2.75e−4 | 3.03e−4 |
| 20 | 2.78e−4 | 2.73e−4 | 2.91e−4 |
| 40 | 2.83e−4 | 2.63e−4 | 2.98e−4 |
| 60 | 2.68e−4 | 2.59e−4 | 2.72e−4 |
| 100 | 2.61e−4 | 2.56e−4 | 2.70e−4 |

Abbreviations: EPINN, energy-based physics-informed neural network; MENE, magnetic energy norm error; STD, standard deviation.

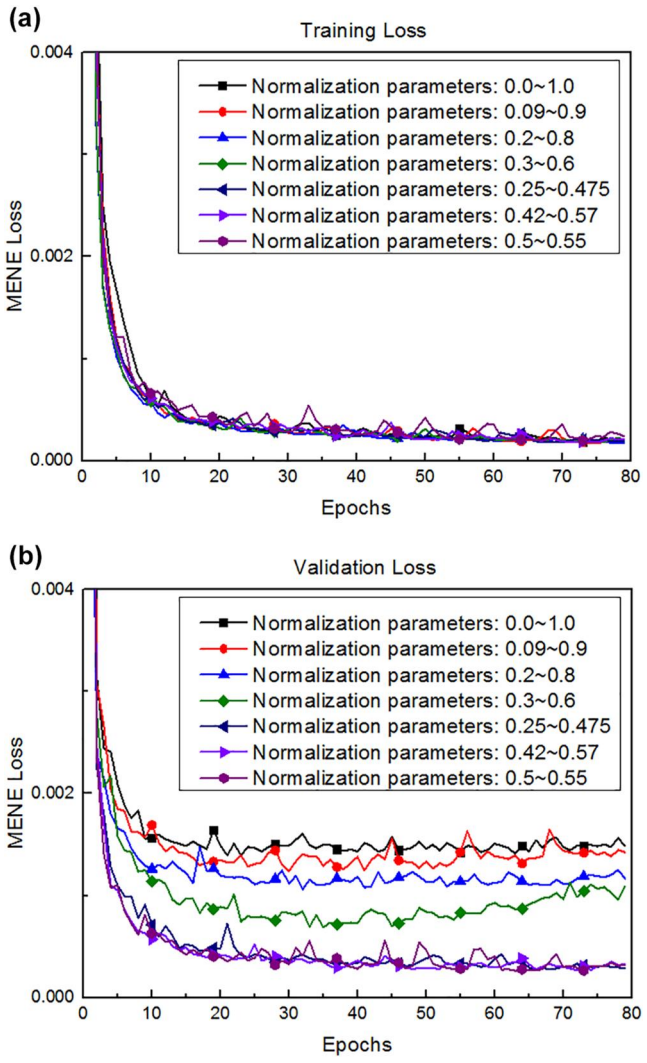



**FIGURE 7** Training curves with different normalization ranges (magnetic energy norm error)

Next, we consider the EPINN in the sense of magnetic energy by replacing the loss function from MENE to MEE. Repeat the above hyperparameter investigation process,

**TABLE 5** The MENE training loss of the EPINN with different normalization ranges

| Normalization range | Average | Minimum | Maximum |
|---|---|---|---|
| 0.0–1.0 | 1.86e−4 | 1.75e−4 | 1.96e−4 |
| 0.09–0.9 | 1.79e−4 | 1.74e−4 | 1.90e−4 |
| 0.2–0.8 | 1.75e−4 | 1.65e−4 | 1.83e−4 |
| 0.25–0.7 | 1.76e−4 | 1.70e−4 | 1.87e−4 |
| 0.3–0.6 | 1.81e−4 | 1.67e−4 | 1.88e−4 |
| 0.25–0.475 | 1.78e−4 | 1.72e−4 | 1.82e−4 |
| 0.42–0.57 | 1.79e−4 | 1.69e−4 | 1.85e−4 |
| 0.5–0.55 | 1.87e−4 | 1.77e−4 | 1.98e−4 |

Abbreviations: EPINN, energy-based physics-informed neural network; MENE, magnetic energy norm error.

**TABLE 6** The MENE validation loss of the EPINN with different normalization ranges

| Normalization range | Average | Minimum | Maximum |
|---|---|---|---|
| 0.0–1.0 | 1.38e−3 | 1.30e−3 | 1.47e−3 |
| 0.09–0.9 | 1.30e−3 | 1.03e−3 | 1.43e−3 |
| 0.2–0.8 | 1.08e−3 | 9.04e−4 | 1.23e−3 |
| 0.25–0.7 | 4.67e−3 | 1.75e−3 | 6.83e−3 |
| 0.3–0.6 | 7.19e−4 | 6.45e−4 | 8.85e−4 |
| 0.25–0.475 | 2.91e−4 | 2.75e−4 | 3.03e−4 |
| 0.42–0.57 | 2.99e−4 | 2.55e−4 | 3.61e−4 |
| 0.5–0.55 | 2.72e−4 | 2.63e−4 | 3.01e−4 |

Abbreviations: EPINN, energy-based physics-informed neural network; MENE, magnetic energy norm error.

keeping the DL network architecture, other hyperparameters and the training strategy unchanged. Figure 8 shows the training curves of EPINN with different STDs. Consistent with the research about MENE, for the EPINN with the MEE loss function, STD has basically no effect on the training loss, but has some effect on the validation loss, and the network performance is more stable when STD is greater than 10. As before, the above training process was repeated 10 times and the average, maximum and minimum training loss and validation loss are illustrated in Table 7 and Table 8.

Similar to MENE, we can observe that the convergence process of the training sample is better than the validation samples. Different values of STD have little effect on the training loss. On the other hand, the validation loss will be affected by the STD, but the convergence performance of MEE is more stable than MENE, and the difference between the upper and lower bounds of validation loss with the same STD is smaller. Taking the STD value of 5 as an example, the minimum value over 10 replications is 4.32e−3 and the maximum value is 6.22e−3. As for EPINN with MENE, the maximum value of the validation loss can be up to 3 times

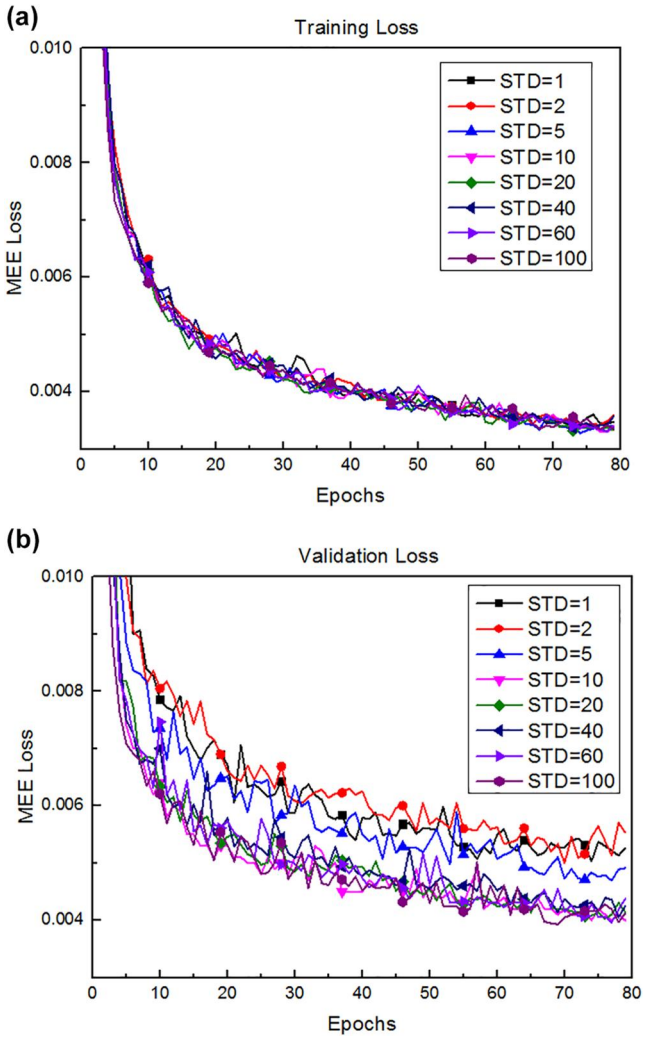



**FIGURE 8** Training curve with different standard deviations (magnetic energy error)

**TABLE 7** The MEE training loss of the EPINN with different STDs

| STD | Average | Minimum | Maximum |
|---|---|---|---|
| 1 | 3.33e−3 | 3.25e−3 | 3.44e−3 |
| 2 | 3.33e−3 | 3.28e−3 | 3.38e−3 |
| 5 | 3.31e−3 | 3.26e−3 | 3.38e−3 |
| 10 | 3.31e−3 | 3.25e−3 | 3.40e−3 |
| 20 | 3.30e−3 | 3.24e−3 | 3.39e−3 |
| 40 | 3.32e−3 | 3.28e−3 | 3.36e−3 |
| 60 | 3.30e−3 | 3.18e−3 | 3.33e−3 |
| 100 | 3.31e−3 | 3.24e−3 | 3.36e−3 |

Abbreviations: EPINN, energy-based physics-informed neural network; MEE, magnetic energy error; STD, standard deviation.

the minimum value. Like MENE, MEE also converges steadier when STD takes values between 10 and 100.

Figure 9 shows the training curve of EPINN with MEE with different normalization ranges. The normalization range

**TABLE 8** The MEE validation loss of the EPINN with different STDs

| STD | Average | Minimum | Maximum |
|---|---|---|---|
| 1 | 4.97e−3 | 4.67e−3 | 5.23e−3 |
| 2 | 5.58e−3 | 4.57e−3 | 7.66e−3 |
| 5 | 5.03e−3 | 4.32e−3 | 6.22e−3 |
| 10 | 4.03e−3 | 3.99e−3 | 4.06e−3 |
| 20 | 4.04e−3 | 3.97e−3 | 4.12e−3 |
| 40 | 4.07e−3 | 3.97e−3 | 4.22e−3 |
| 60 | 3.95e−3 | 3.90e−3 | 3.99e−3 |
| 100 | 3.96e−3 | 3.89e−3 | 4.06e−3 |

Abbreviations: EPINN, energy-based physics-informed neural network; MEE, magnetic energy error; STD, standard deviation.

has little effect on the training loss, as the training curve converges smoothly for varied values, similar to the results of STD. On the other hand, it has a considerable impact on the validation loss. Tables 9 and 10 demonstrated the average, maximum and minimum losses over 10 replications. To summarize, similar to the findings of the previous studies for MENE, the training process of the EPINN with MEE network is affected by both the STD and normalization range. Like MENE, the proper STD value for the EPINN with MEE is 10–100 as well. Unlike MENE, the EPINN based on MEE is more sensitive to the normalization range and more likely to lead to network divergence due to improper hyperparameter. The suitable normalization range is smaller, about 0.25–0.475.

In order to ensure that the EPINN with both loss functions can be effectively trained, in the following post-test procedure, all these networks are trained with the STD set to be 10 and the normalization range ia set to be 0.25–0.475. The fine-tuned DL network can be effectively trained by only 50% of the datasets (40% samples for training, 10% samples for validation), the successfully trained EPINN models can be used to predict the distribution of the magnetic flux density with different geometry parameters, hence accelerating the computation. Comparison of the magnetic field distribution around the transformer windings computed by FEM and the corresponding DL prediction results are demonstrated in Figures 10 and 11. It can be observed that the DL predictions differ little from the FEM computations, the errors mainly come from the winding area, which is where the magnetic field distribution is concentrated. The relative errors are below 4% for both physical-informed loss functions. In addition, in order to compare systematically the DL performance and generalization capacity of EPINN with different loss functions in detail, post-tests will be performed to test the interpolation and extrapolation in the next section.

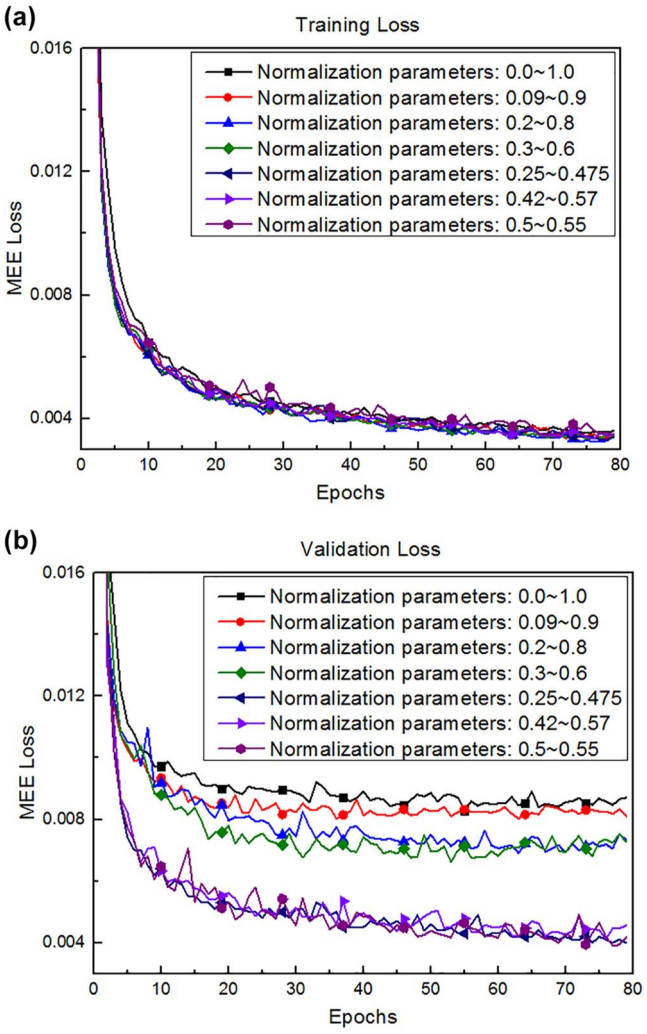



**FIGURE 9** Training curve with different normalization ranges (magnetic energy error)

**TABLE 9** The MEE training loss of the EPINN with different normalization ranges

| STD | Average | Minimum | Maximum |
|---|---|---|---|
| 0.0–1.0 | 3.52e−3 | 3.46e−3 | 3.58e−3 |
| 0.09–0.9 | 3.36e−3 | 3.31e−3 | 3.42e−3 |
| 0.2–0.8 | 3.28e−3 | 3.23e−3 | 3.34e−3 |
| 0.25–0.7 | 3.27e−3 | 3.16e−3 | 3.34e−3 |
| 0.3–0.6 | 3.30e−3 | 3.21e−3 | 3.35e−3 |
| 0.25–0.475 | 3.31e−3 | 3.25e−3 | 3.40e−3 |
| 0.42–0.57 | 3.31e−3 | 3.22e−3 | 3.35e−3 |
| 0.5–0.55 | 3.43e−3 | 3.37e−3 | 3.51e−3 |

Abbreviations: EPINN, energy-based physics-informed neural network; MEE, magnetic energy error; STD, standard deviation.

## 4.2 | Post-test with MENE

After the training of the network, and taking the geometric label parameters as input, the corresponding magnetic field distribution is generated using the trained EPINN network and compared with the ground truth to calculate the error. What's more, in order to compare the performance of EPINN and

**TABLE 10** The MEE validation loss of the EPINN with different normalization ranges

| STD | Average | Minimum | Maximum |
|---|---|---|---|
| 0.0–1.0 | 8.30e−3 | 7.90e−3 | 8.70e−3 |
| 0.09–0.9 | 7.79e−3 | 7.28e−3 | 8.24e−3 |
| 0.2–0.8 | 6.90e−3 | 6.58e−3 | 7.04e−3 |
| 0.25–0.7 | 1.71e−2 | 1.41e−2 | 1.95e−2 |
| 0.3–0.6 | 6.53e−3 | 6.11e−3 | 7.07e−3 |
| 0.25–0.475 | 4.03e−3 | 3.99e−3 | 4.06e−3 |
| 0.42–0.57 | 4.15e−3 | 3.78e−3 | 4.84e−3 |
| 0.5–0.55 | 3.99e−3 | 3.93e−3 | 4.06e−3 |

Abbreviations: EPINN, energy-based physics-informed neural network; MEE, magnetic energy error; STD, standard deviation.

traditional CNN, based on the above hyperparameter configurations, MAE, MSE, MENE and MEE are utilized as the loss function to respectively train the DL network from the same training sample and validation sample, as illustrated in Table 3.

As mentioned before, when facing the transformer winding leakage field problem, researchers tend to focus more on the magnetic field distribution near the winding than the other regions. Therefore, the error evaluation between the DL prediction results and the database ground truth is calculated using the MENE loss function. It should be mentioned here that the capability of the interpolation is evaluated when the test samples are inside the training samples and validation samples. On the other hand, the capacity of extrapolation is evaluated when the test samples are outside the target. In order to evaluate the generalization capacity of the EPINN in a

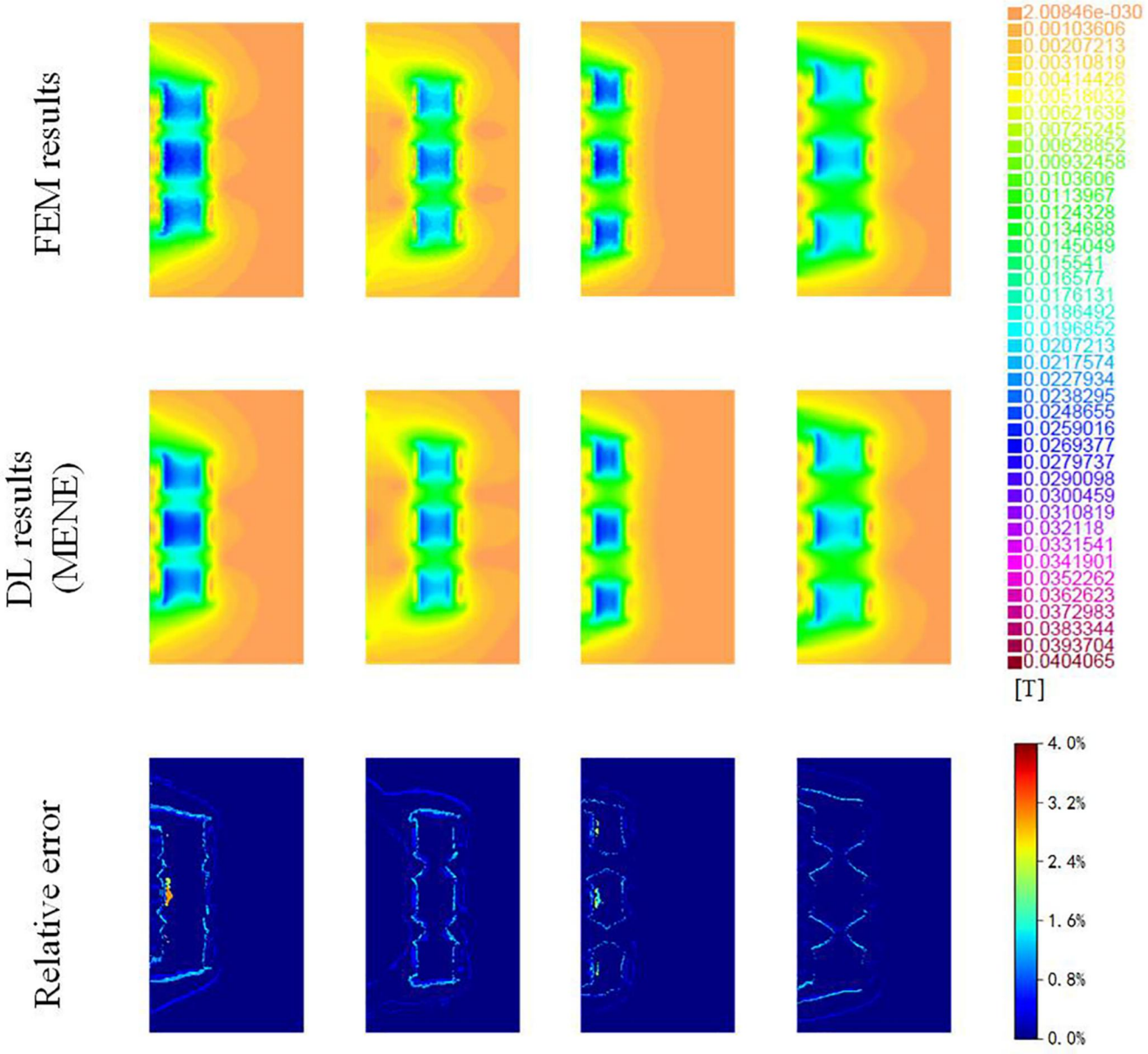



**FIGURE 10** Comparison of finite element method results and deep learning predictions (magnetic energy norm error)

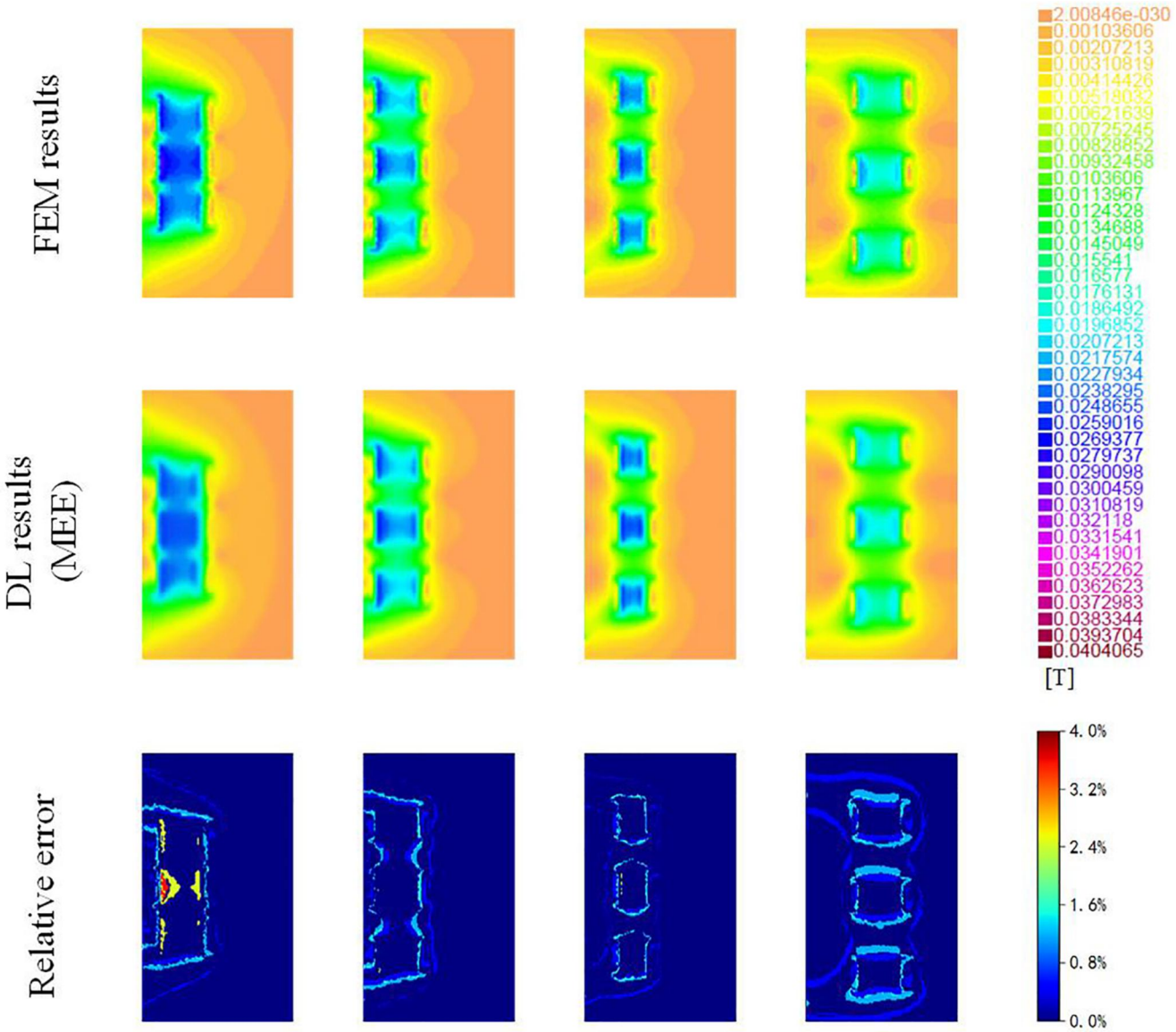



**FIGURE 11** Comparison of finite element method results and deep learning predictions (magnetic energy error)

comprehensive and systematic way, we performed the post-test on different kinds of samples in the following. For clarity, we have tabulated the data after each graph for comparison.

To begin with, the validation samples (i.e. Group #7) are used to compute the post error. The Gaussian distribution generated by the test sample labels is fed into the four separate network models trained with different loss functions to obtain the corresponding outputs, and then the MENE is utilized to calculate the error between these four sets of prediction results that output from the DL network with different loss functions and the corresponding ground truth in the database. The post-test evaluation results of the DL model that trained with MAE, MSE, MENE and MENE in functions of the sample index are separately shown in Figure 12. As we can see, among the 100 samples in Group #7, except a few, the network trained by MENE generates the best performing output with the smallest post error, as the blue line plots in Figure 12.

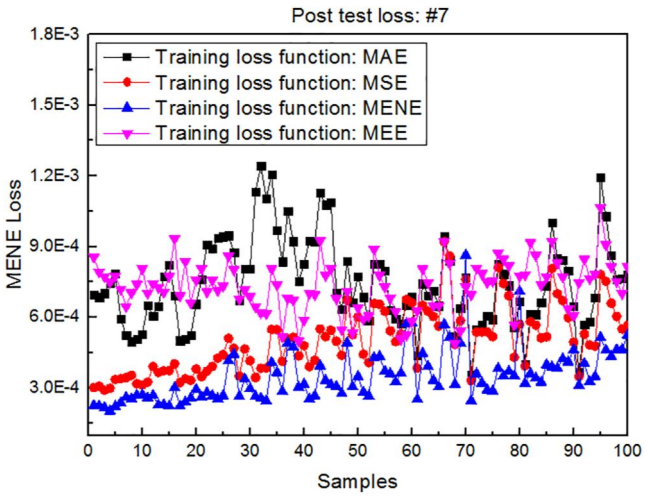



**FIGURE 12** The post-test magnetic energy norm error loss of Group #7

The same with the aforementioned hyperparameter investigation process, in order to avoid the influence of randomness on the conclusion, the same set of networks is trained 10 times with the same hyperparameters and the same database, then the post-test error is calculated for these 10 trained model respectively. The results are shown in Table 11, each cell reports the mean (min, max) of summary performance calculated over 10 times post-test.

Next, four groups of training samples are adopted as test samples to carry out the post-test, which is to evaluate the interpolation capability of the network. The results of Group #2, Group #4, Group #6 and Group #8 are shown in Figure 13. Consistent with the test results of Group #7, these results demonstrate that the MENE performs significantly better than the other three loss functions in all four training groups, as the blue line indicated. The summary of these four groups of training samples is shown in Table 12. The results suggest that the EPINN trained with MENE has better interpolation performance compared with the other three loss functions. This means that the desired physical information has been integrated into the learning objectives and takes a greater weight in the error estimation process during the DL training by adjusting the loss function. Naturally, better performance can be obtained in the post-test when using the corresponding physical-informed loss function as the error evaluator. In other words, we get the same results as theoretically expected. In addition, we can also find that the interpolation performance of the network is very stable. For all the five groups of samples involved in the training procedure, including the training samples and validations samples, the post-test loss with MENE is in the range from 1.5e−4 to 3e−4, with little difference in the results between groups. This indicates that the network performs well in the interpolation test after effective convergence.

The above five groups of samples are all involved in the training process to update the weight of the EPINN network. The post-test process is to test the interpolation capacity of the network after effectively training. The purpose of employing DL is to provide an efficient surrogate model for complex and time-consuming computations, utilizing parts of the computation results as the database to train the DL network and then using the trained model to predict the results of subsequent samples, thus accelerating the computation and reducing the cost. Therefore, the generalization ability and the extrapolation capacity of the model are also crucial for evaluating the performance of a DL network. Hence, we select four groups from the samples that have not participated in the network training process, that is, other than the training samples and validation samples, to perform the post-test process described above. The post-test error computation results of Group #1, Group #3, Group #5 and #9 are shown in Figure 14. The summary of these results in 10 replications are shown in Table 13.

The results can be observed as follows: the EPINN trained with MENE has advantages in the extrapolation post-test as well, as the blue line illustrated in Figure 14. We can observe two findings from these results. First, the performance of extrapolation is much worse than interpolation; the post-test MENE loss for these four groups of other samples is in the range from 2.7e−4 to 1.5e−3, as a comparison, the interpolation error of the five groups of the training sample and validation sample is in the range from 1.5e−4 to 3e−4, which is much less than the extrapolation error. Second, the performance of extrapolation is unstable and is strongly related to the values of the input variables. All 1000 samples are arranged in ascending order of the labelled variables, which means that the variables of group #3 take values between group #2 and group #4, and similarly, the variables of group #5 take values between group #4 and group #6. Their input labels take values within the range of variable values of the training samples, and this is the reason for their better extrapolation performance. On the other hand, for group #1 and group #9, their input variables take values outside the training samples. Hence, the extrapolation results of these two groups are much worse than group #3 and group #5. This indicates that for EPINN with MENE, good extrapolation performance can only be obtained for samples within the vicinity of the training samples, and the applicability of extrapolation is limited. As we defined in Equation (8), the MENE loss function can be seen as a specially designed MSE whose weights have been determined by the material permeability and the QOI. This increases the proportion of magnetic leakage flux error in the MENE loss due to the high permeability of the iron core. Different from MSE, instead of considering an average accuracy of the magnetic flux density, more attention has been paid to the accuracy of the leakage flux during the learning process.

**TABLE 11** The summary of the post-test MENE loss of Group #7

| Training loss function | Post-test loss (#7) |
|---|---|
| MAE | 6.32e−4 (5.34e−4, 7.48e−4) |
| MSE | 4.72e−4 (4.17e−4, 5.02e−4) |
| MENE | 3.57e−4 (3.50e−4, 3.65e−4) |
| MEE | 5.23e−4 (4.32e−4, 7.32e−4) |

Abbreviations: MAE, mean absolute error; MEE, magnetic energy error; MENE, magnetic energy norm error; MSE, mean square error.

## 4.3 | Post-test with MEE

The EPINN with MENE can directly integrate the physical information into the DL procedure by properly weighting the given data and have shown good potentials in accuracy and generalizability in the post-tests. However, there are some more complex physical contexts that cannot be characterized only by weighting the input data. For example, noting that during the FEM computations, researchers prefer to concentrate on the magnetic energy rather than merely on the magnetic flux distribution, and it makes sense that the error between the DL prediction results and the FEM computation results through the perspective of magnetic energy, that is, the

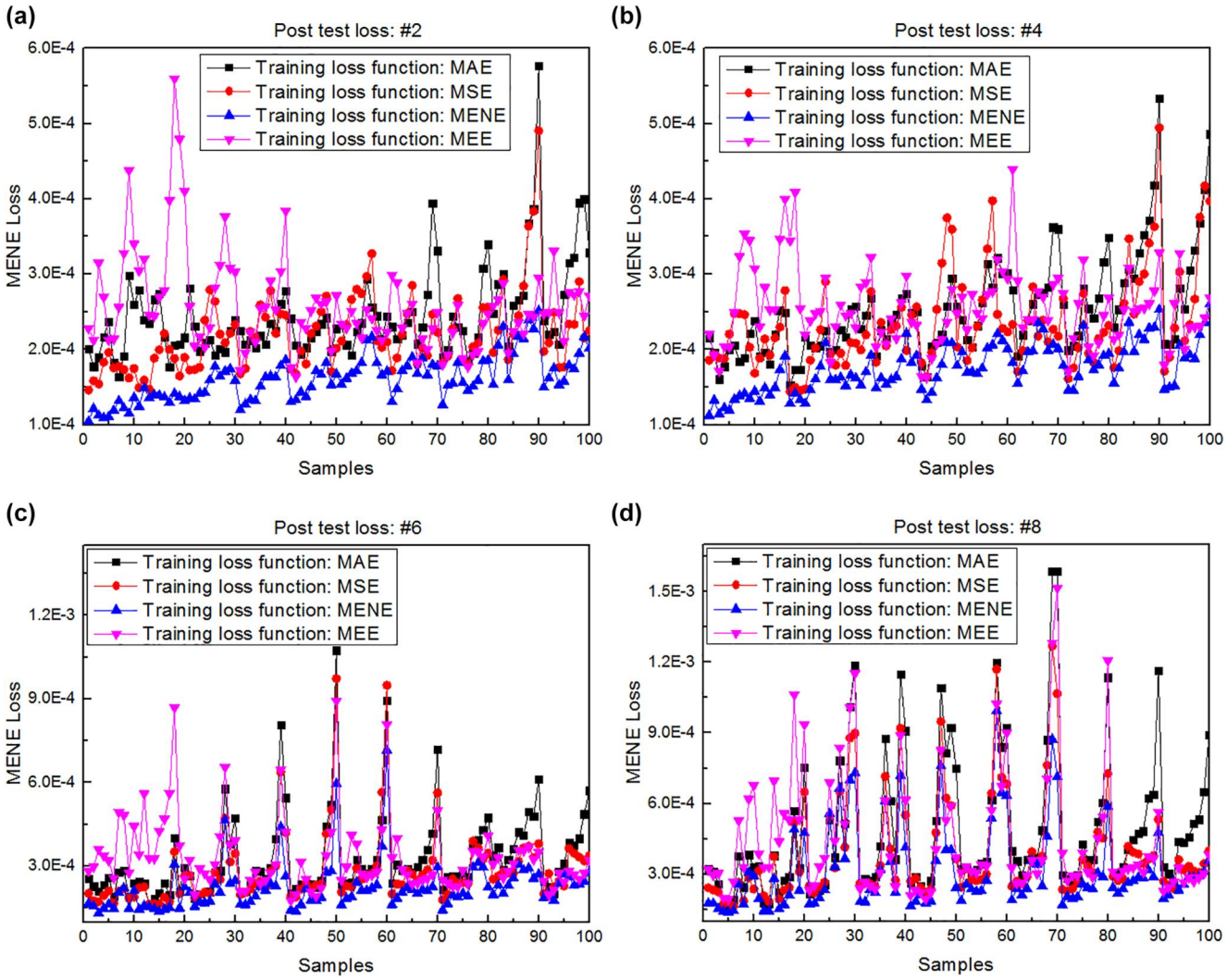



**FIGURE 13** The post-test magnetic energy norm error loss of training samples (interpolation)

**TABLE 12** The post-test MENE loss of training samples

| Training loss function | Post-test loss (#2) | Post-test loss (#4) |
|---|---|---|
| MAE | 2.08e−4 (1.85e−4, 2.44e−4) | 2.28e−4 (2.02e−4, 2.53e−4) |
| MSE | 2.01e−4 (1.88e−4, 2.20e−4) | 2.19e−4 (2.03e−4, 2.36e−4) |
| MENE | 1.68e−4 (1.61e−4, 1.72e−4) | 1.85e−4 (1.76e−4, 1.94e−4) |
| MEE | 2.15e−4 (1.94e−4, 2.58e−4) | 2.30e−4 (2.17e−4, 2.56e−4) |
| **Training loss function** | **Post-test loss (#6)** | **Post-test loss (#8)** |
| MAE | 2.98e−4 (2.80e−4, 3.29e−4) | 4.50e−4 (4.05e−4, 4.92e−4) |
| MSE | 2.73e−4 (2.52e−4, 2.84e−4) | 3.71e−4 (3.59e−4, 3.92e−4) |
| MENE | 2.32e−4 (2.23e−4, 2.42e−4) | 3.34e−4 (3.14e−4, 3.46e−4) |
| MEE | 2.84e−4 (2.66e−4, 3.33e−4) | 4.03e−4 (3.88e−4, 4.47e−4) |

Abbreviations: MAE, mean absolute error; MEE, magnetic energy error; MENE, magnetic energy norm error; MSE, mean square error.

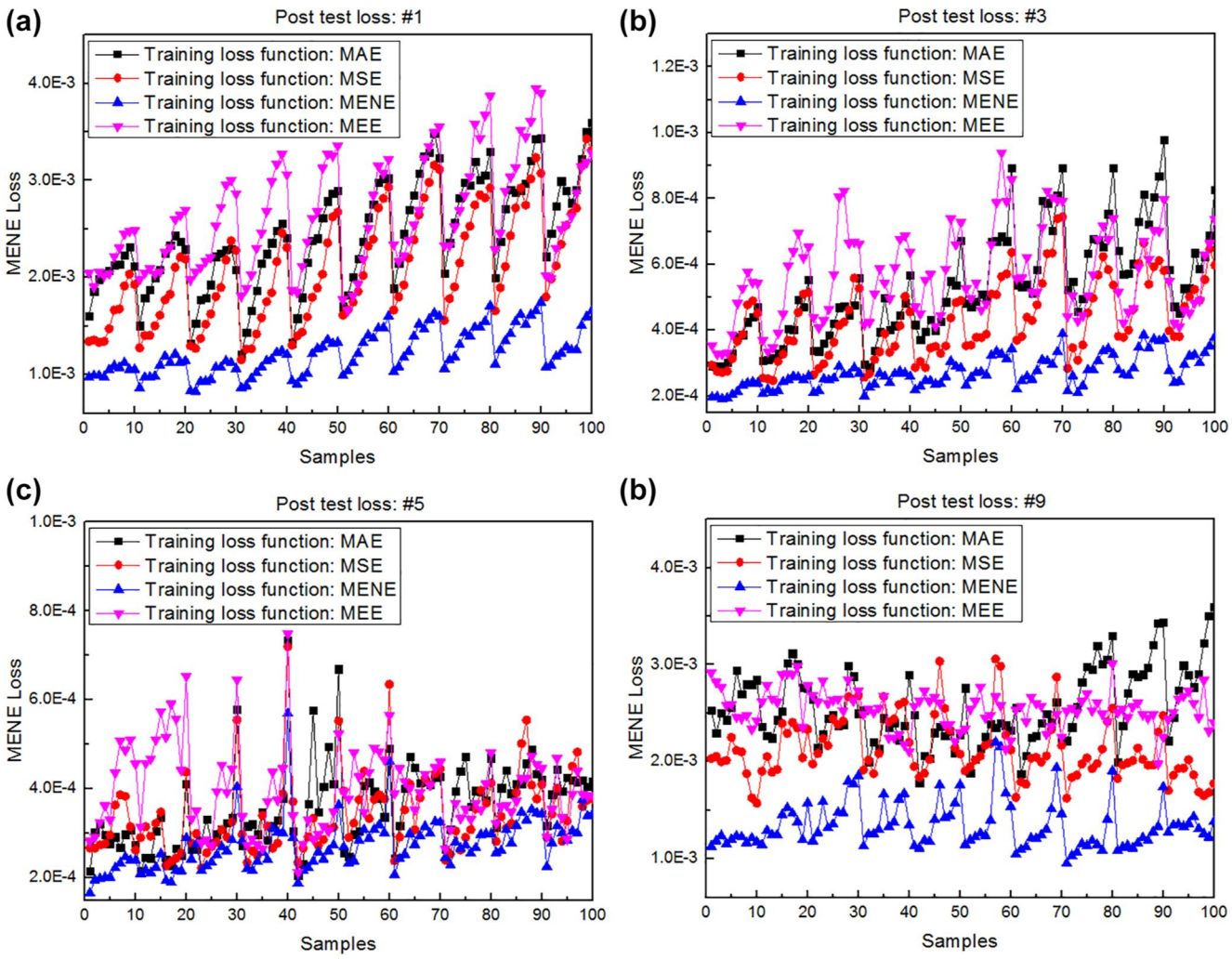



**FIGURE 14** The post-test magnetic energy norm error loss of other samples (extrapolation)

**TABLE 13** The post-test MENE loss of other samples

| Training loss function | Post-test loss (#1) | Post-test loss (#3) |
|---|---|---|
| MAE | 2.23e−3 (2.09e−3, 2.43e−3) | 5.31e−4 (4.82e−4, 5.78e−4) |
| MSE | 1.75e−3 (1.58e−3, 2.10e−3) | 3.90e−4 (3.56e−4, 4.31e−4) |
| MENE | 1.34e−3 (1.22e−3, 1.45e−3) | 2.94e−4 (2.71e−4, 3.15e−4) |
| MEE | 2.17e−3 (2.01e−3, 2.63e−3) | 4.27e−4 (3.87e−4, 4.91e−4) |
| **Training loss function** | **Post-test loss (#5)** | **Post-test loss (#9)** |
| MAE | 3.41e−4 (3.28e−4, 3.61e−4) | 2.27e−4 (2.08e−4, 2.48e−4) |
| MSE | 3.23e−4 (3.05e−4, 3.42e−4) | 1.90e−4 (1.72e−4, 2.13e−4) |
| MENE | 2.81e−4 (2.73e−4, 2.88e−4) | 1.42e−4 (1.34e−4, 1.52e−4) |
| MEE | 3.47e−4 (3.29e−4, 3.99e−4) | 2.28e−4 (2.14e−4, 2.54e−4) |

Abbreviations: MAE, mean absolute error; MEE, magnetic energy error; MENE, magnetic energy norm error; MSE, mean square error.

MEE loss function. Same as the post-test process with MENE, firstly, we use validation samples in the network training process to perform the test. The MEE is adopted to compute the error between these four sets of DL predictions and the corresponding FEM results. The errors of 100 samples from Group #7 are illustrated in Figure 15. As we can see, among the 100 samples in Group #7, EPINN with MEE performs the best in the post-test in the energy sense, as the pink line plots in Figure 15.

Similar to the post-test with MENE, we now repeat the post-test with MEE as well. The summary of these replication results are demonstrated in Table 14. Similar to the results of MENE, MEE also showed advantages in the corresponding post-test. The average loss of the EPINN trained with MEE is 4.32e−3, while the losses of networks that trained with these conventional loss functions without physical meanings are all above 5e−3.

Following that, four groups of training samples are adopted to carry out the post-test, and the MEE losses of Group #2, Group #4, Group #6 and Group #8 are shown in Figure 16. In all these groups, the results show that the MEE performs better than the other three loss functions. Table 15 depicts the post-test summary of these groups. Compared to MENE, EPINN with MEE have less advantage in the post-test with training samples, but the advantages are still non-negligible, implying that EPINN trained with MEE has better interpolation performance than the other three loss functions in the magnetic field energy sense. Besides, the interpolation performance of MEE, like MENE, is very stable as well, and the post-test errors difference between training sample groups are very small.

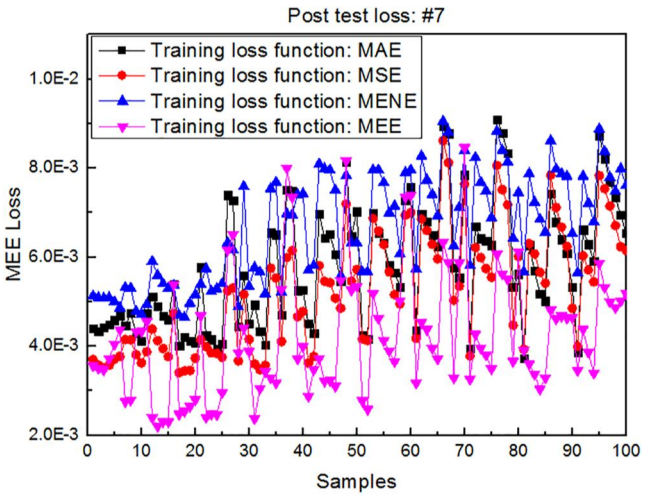



**FIGURE 15** The post-test magnetic energy error loss of Group #7

**TABLE 14** The summary of the post-test MEE loss for Group #7

| Training loss function | Post-test loss (#7) |
|---|---|
| MAE | 5.18e−3 (4.75e−3, 5.81e−3) |
| MSE | 5.01e−3 (4.63e−3, 5.26e−3) |
| MENE | 5.27e−3 (4.72e−3, 6.65e−3) |
| MEE | 4.32e−3 (4.23e−3, 4.41e−3) |

Abbreviations: MAE, mean absolute error; MEE, magnetic energy error; MENE, magnetic energy norm error; MSE, mean square error.

To verify the generalization ability and the extrapolation capacity of the EPINN with MEE, 4 groups of samples outside the training samples and validation samples are selected to test the trained model. The MEE losses of Group #1, Group #3, Group #5 and Group #9 are shown in Figure 17. The summary of these four groups are shown in Table 16. From the results it can be observed that the EPINN trained with MEE has advantages in the extrapolation test as well. In addition, the two findings in the extrapolation of MENE are also working in the EPINN with MEE, which are: (1) extrapolation performance is generally worse than the interpolation; (2) the effective range of extrapolation is limited and good results can be obtained for the sample around the training samples.

The above numerical experiments about the EPINN with MENE and MEE manifest that it is effective to adjust the DL networks by introducing physical information into the loss function. The physics-informed loss function designed for a specific task can obtain better performance on the corresponding task compared with other loss functions. With fine-tuned hyperparameters, the EPINN with these two physics-informed loss functions has good accuracy in the interpolation test. In terms of extrapolation, good generalization ability can be guaranteed by ensuring that the extrapolation samples do not differ significantly from the training samples. The farther the extrapolation range, the poorer the accuracy.

## 5 | CONCLUSION

In this paper, an EPINN network has been presented to solve the low-frequency magnetic problem of transformers with different geometries. Physical information are integrated into the DL network by modifying the loss function. Extra terms are included in the loss function to incorporate the physical model as an additional constraint, guaranteeing that the network outputs satisfy the physical model as well. This approach can be viewed as a special form of target-specific DL, in which a DL model is forced to match the ground truth while also producing predictions that roughly comply with a set of physical restrictions. Two loss functions are proposed for tasks with different physical meanings, namely MENE and MEE.

From the numerical example, it can be found that the physics-informed loss functions can improve the network performance while increasing the sensitivity of some network hyperparameters, which makes the DL model easier to diverge. These hyperparameters that become more sensitive due to changing the loss function are investigated in detail. In addition, to verify the enhancement of the physics-informed loss functions, as a comparison, conventional CNN using MAE and MSE and the EPINN were respectively utilized to

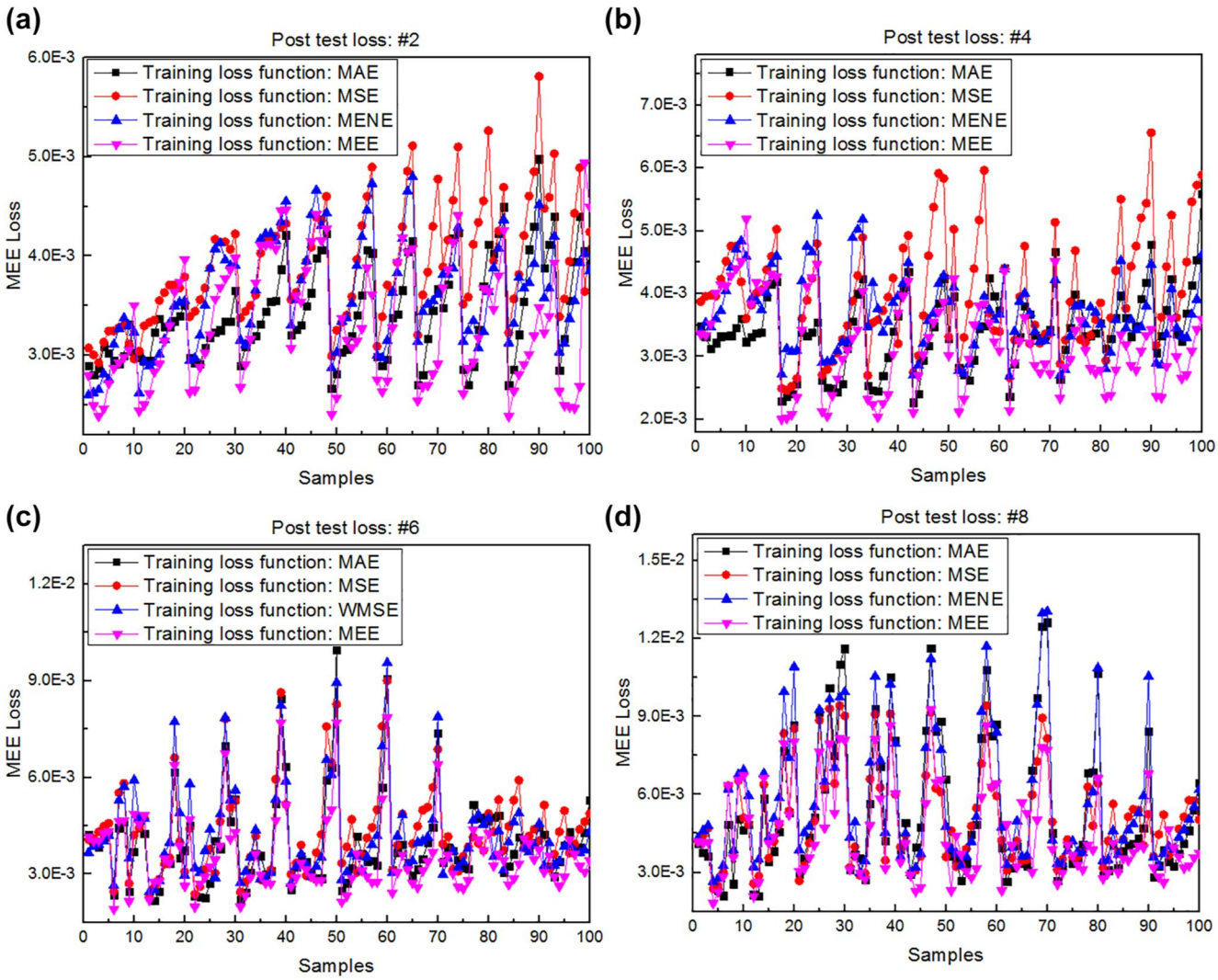



**FIGURE 16** The post-test magnetic energy error loss of training samples (interpolation)

**TABLE 15** The post-test MEE loss of the training samples

| Training loss function | Post-test loss (#2) | Post-test loss (#4) |
|---|---|---|
| MAE | 3.32e−3 (3.22e−3, 3.42e−3) | 3.31e−3 (3.21e−3, 3.40e−3) |
| MSE | 3.75e−3 (3.48e−3, 3.98e−3) | 3.76e−3 (3.56e−3, 4.01e−3) |
| MENE | 3.46e−3 (3.40e−3, 3.60e−3) | 3.48e−3 (3.34e−3, 3.69e−3) |
| MEE | 3.33e−3 (3.19e−3, 3.40e−3) | 3.28e−3 (3.25e−3, 3.31e−3) |
| **Training loss function** | **Post-test loss (#6)** | **Post-test loss (#8)** |
| MAE | 3.88e−3 (3.73e−3, 3.97e−3) | 5.12e−3 (5.01e−3, 5.32e−3) |
| MSE | 4.03e−3 (3.82e−3, 4.31e−3) | 4.98e−3 (4.80e−3, 5.07e−3) |
| MENE | 3.93e−3 (3.74e−3, 4.20e−3) | 5.19e−3 (4.96e−3, 5.25e−3) |
| MEE | 3.66e−3 (3.59e−3, 3.71e−3) | 4.61e−3 (4.52e−3, 4.67e−3) |

Abbreviations: MAE, mean absolute error; MEE, magnetic energy error; MENE, magnetic energy norm error; MSE, mean square error.

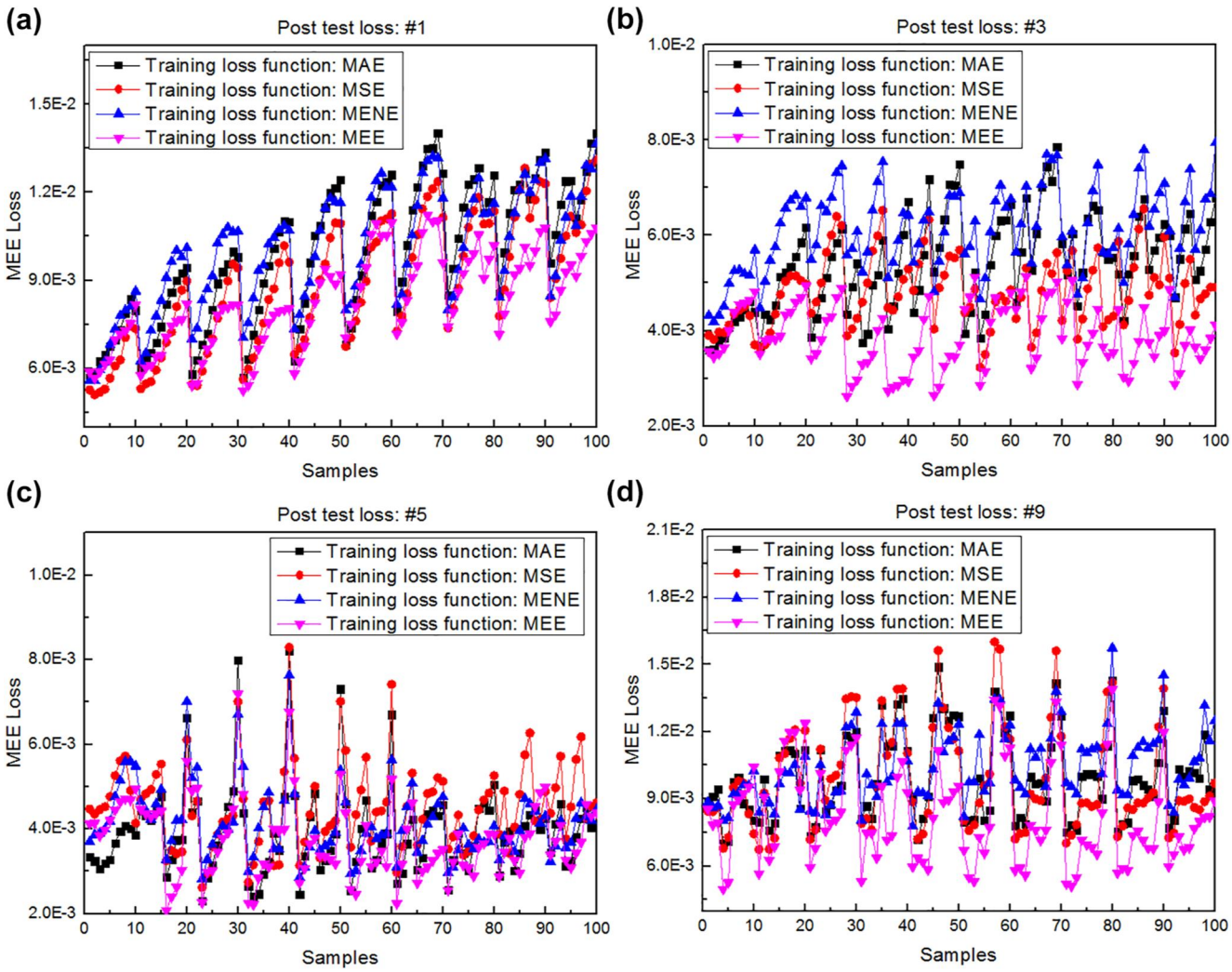



**FIGURE 17** The post-test magnetic energy error loss of other samples (extrapolation)

**TABLE 16** The post-test MEE loss of other samples

| Training loss function | Post-test loss (#1) | Post-test loss (#3) |
|---|---|---|
| MAE | 9.53e−3 (9.28e−3, 9.87e−3) | 4.99e−3 (4.64e−3, 5.40e−3) |
| MSE | 8.93e−3 (8.75e−3, 9.19e−3) | 4.42e−3 (4.05e−3, 4.81e−3) |
| MENE | 9.22e−3 (8.65e−3, 9.95e−3) | 4.72e−3 (4.18e−3, 5.12e−3) |
| MEE | 8.60e−3 (8.25e−3, 8.83e−3) | 4.09e−3 (3.88e−3, 4.27e−3) |
| **Training loss function** | **Post-test loss (#5)** | **Post-test loss (#9)** |
| MAE | 3.91e−3 (3.90e−3, 3.93e−3) | 9.54e−3 (9.11e−3, 9.95e−3) |
| MSE | 4.25e−3 (4.10e−3, 4.53e−3) | 9.19e−3 (8.84e−3, 9.88e−3) |
| MENE | 3.99e−3 (3.92e−3, 4.15e−3) | 9.56e−3 (8.98e−3, 1.04e−2) |
| MEE | 3.82e−3 (3.74e−3, 3.87e−3) | 8.60e−3 (8.23e−3, 8.83e−3) |

Abbreviations: MAE, mean absolute error; MEE, magnetic energy error; MENE, magnetic energy norm error; MSE, mean square error.

train from the same database with the same network architecture and configurations. The numerical experiments demonstrate that the designed physics-informed loss functions MENE and MEE have significant advantages over conventional DL loss functions MAE and MSE in terms of prediction for magnetic field distribution. Besides, the EPINN has shown good potential and stability can obtain outstanding performance in interpolation tests. As for the extrapolation, the prediction performance of the network is limited by the input label values of the test samples; good results can be obtained only for samples with label values in the vicinity of the training samples.

## CONFLICT OF INTEREST

The authors declare that there is no conflict of interest.

## DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

## ORCID

*Zuqi Tang* https://orcid.org/0000-0002-5402-5858

## REFERENCES


1. Yilmaz, M., Krein, P.T.: Capabilities of finite element analysis and magnetic equivalent circuits for electrical machine analysis and design. In: 2008 IEEE Power Electronics Specialists Conference, pp. 4027–4033 (2008)
2. Rizzo, R., et al.: Magnetic FEM design and experimental validation of an innovative fail-safe magnetorheological clutch excited by permanent magnets. IEEE Trans. Energy Convers. 29(3), 628–640 (2014)
3. Schmidthausler, D., Clemens, M.: Low-order electroquasistatic field simulations based on proper orthogonal decomposition. IEEE Trans. Magn. 48(2), 567–570 (2012)
4. Henneron, T., Clénet, S.: Model order reduction of non-linear magnetostatic problems based on POD and DEI methods. IEEE Trans. Magn. 50(2), 1–4 (2014)
5. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Proceedings of Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
6. Khan, A., Ghorbanian, V., Lowther, D.: Deep learning for magnetic field estimation. IEEE Trans. Magn. 55(6), 1–4 (2019)
7. Arnoux, P., Caillard, P., Gillon, F.: Modeling finite-element constraint to run an electrical machine design optimization using machine learning. IEEE Trans. Magn. 51(3), 1–4 (2015)
8. Li, J., et al.: Integrated high-frequency coaxial transformer design platform using artificial neural network optimization and FEM simulation. IEEE Trans. Magn. 51(3), 1–4 (2015)
9. Azari, M., et al.: Optimum design of a line-start permanent-magnet motor with slotted solid rotor using neural network and imperialist competitive algorithm. IET Electr. Power Appl. 11, 1–8 (2017)
10. Elbir, A.M., Mishra, K.V.: Deep learning design for joint antenna selection and hybrid beamforming in massive MIMO. In: 2019 IEEE International Symposium on Antennas and Propagation and USNC-URSI Radio Science Meeting
11. Minervini, M., et al.: Convolutional neural networks for automated rolling bearing diagnostics in induction motors based on electromagnetic signals. Appl. Sci. 11, 7878 (2021)
12. Kumar, P., Hati, A.: Convolutional neural network with batch normalisation for fault detection in squirrel cage induction motor. IET Electr. Power Appl. 15, 39–50 (2021)
13. Sasaki, H., Igarashi, H.: Topology optimization accelerated by deep learning. IEEE Trans. Magn. 55(6), 1–5 (2019)
14. Kirchgässner, W., Wallscheid, O., Böcker, J.: Deep residual convolutional and recurrent neural networks for temperature estimation in permanent magnet synchronous motors. In: 2019 IEEE International Electric Machines and Drives Conference (IEMDC), pp. 1439–1446 (2019)
15. Gong, R., Tang, Z.: Investigation of convolutional neural network U-Net under small datasets in transformer magneto-thermal coupled analysis. COMPEL. 39(4), 959–970 (2020)
16. Gong, R., Tang, Z.: Training sample selection strategy applied to CNN in magneto-thermal coupled analysis. IEEE Trans. Magn. 57(6), 1–4 (2021)
17. Kingma, D.P., Ba, J.L.: Adam: a method for stochastic optimization. arXiv:1412.6980 (2014)
18. Ruder, S.: An overview of gradient descent optimization algorithms. arXiv:1609.04747 (2016)
19. Fan, S., et al.: On line detection of defective apples using computer vision system combined with deep learning methods. J. Food Eng. 286, 110102 (2020)
20. IÅ§Äśn, A., DirekoÄ§lu, C., Şah, M.: Review of MRI-based brain tumor image segmentation using deep learning methods. Procedia Comput. Sci. 102, 317–324 (2016)
21. Miglani, A., Kumar, N.: Deep learning models for traffic flow prediction in autonomous vehicles: a review, solutions, and challenges. Veh. Commun. 20, 100184 (2019)
22. 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. 378, 686–707 (2019)
23. Karniadakis, G.E., et al.: Physics-informed machine learning. Nat. Rev. Phys. 3, 422–440 (2021)
24. Haghighat, E., et al.: A physics-informed deep learning framework for inversion and surrogate modeling in solid mechanics. Comput. Methods Appl. Mech. Eng. 379, 113741 (2021)
25. Lagaris, I.E., Likas, A., Fotiadis, D.I.: Artificial neural networks for solving ordinary and partial differential equations. IEEE Trans. Neural Netw. 9(5), 987–1000 (1998)
26. Sheng, H., Yang, C.: PFNN: a penalty-free neural network method for solving a class of second-order boundary-value problems on complex geometries. J. Comput. Phys. 428(5), 110085 (2021)
27. McFall, K.S., Mahan, J.R.: Artificial neural network method for solution of boundary value problems with exact satisfaction of arbitrary boundary conditions. EEE Trans. Neural Netw. 20(8), 1221–1233 (2009)
28. Beidokhti, R.S., Malek, A.: Solving initial-boundary value problems for systems of partial differential equations using neural networks and optimization techniques. J. Franklin Inst. 346(9), 898–913 (2009)
29. Kharazmi, E., Zhang, Z., Karniadakis, G.E.: Variational physics-informed neural networks for solving partial differential equations. arXiv:1912.00873 (2019)
30. Pang, G., Lu, L., Karniadakis, G.E.: fPINNs: fractional physics-informed neural networks. SIAM J. Sci. Comput. 41(4), A2603–A2626 (2019)
31. Liu, Y., Meng, X., Karniadakis, G.E.: B-PINNs: Bayesian physics-informed neural networks for forward and inverse PDE problems with noisy data. arXiv:2003.06097 (2020)
32. Japtap, A.D., Kharazmi, E., Karniadakis, G.E.: Conservative physics-informed neural networks on discrete domains for conservation laws: applications to forward and inverse problems. Comput. Methods Appl. Mech. Eng. 365, 113028 (2020)
33. Shukla, K., Jagtap, A.D., Karniadakis, G.E.: Parallel physics-informed neural networks via domain decomposition. arXiv:2104.10013 (2021)
34. Mishra, S., Molinaro, R.: Physics informed neural networks for simulating radiative transfer. J. Quant. Spectrosc. Radiat. Transf. 57, 107705 (2021)
35. Cai, S., et al.: Physics-informed neural networks for heat transfer problems. ASME. J. Heat Transfer. 143(6), 060801 (2021)
36. Mishra, S., Molinaro, R.: Physics-informed neural networks (PINNs) for fluid mechanics: a review. arXiv:2009.13291 (2021)
37. Jin, X., et al.: NSFnets (Navier–Stokes flow nets): physics-informed neural networks for the incompressible Navier-Stokes equations. J. Comput. Phys. 426, 109951 (2021)

38. Li, Z., et al.: Fourier neural operator for parametric partial differential equations. arXiv:2010.08895 (2021)
39. Lu, L., et al.: Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. Nat Mach Intell. 3, 218–229 (2021)
40. Tang, Z., et al.: Residual and equilibrated error estimators for magnetostatic problems solved by finite element method. IEEE Trans. Magn. 49(12), 5715–5723 (2013)
41. Tang, Z., et al.: Guaranteed quantity of interest error estimate based on equilibrated flux reconstruction. IEEE Trans. Magn. 57(6), 1–4 (2021)
42. Ruan, J., et al.: HST calculation of a 10 kV oil-immersed transformer with 3D coupled-field method. IET Electr. Power Appl. 14, 921–928 (2020)
43. Hecht, F.: New development in FreeFem++. J. Numer. Math. 20, 251–266 (2012)

**How to cite this article:** Gong, R., Tang, Z.: Further investigation of convolutional neural networks applied in computational electromagnetism under physics-informed consideration. IET Electr. Power Appl. 16(6), 653–674 (2022). https://doi.org/10.1049/elp2.12183