Multi-objective Optimization of Permanent Magnet Motors Using Deep Learning and CMA-ES

Ryosuke Mikami[1] , Hayaho Sato[1] , Shogo Hayashi[1] , Hajime Igarashi[1]

[1]Graduate School of Information Science and Technology, Hokkaido University, Sapporo 060-0814, Japan

ABSTRACT: This paper proposes a multi-objective optimization method for permanent magnet motors using a fast optimization algorithm, Covariance Matrix Adaptation Evolution Strategy (CMA-ES), and deep learning. Multi-objective optimization with topology optimization is effective in the design of permanent magnet motors. Although CMA-ES needs fewer population size than genetic algorithm for single objective problems, this is not evident for multi-objective problems. For this reason, the proposed method generates training data by solving the single-objective optimization multiple times using CMA-ES, and constructs a deep neural network (NN) based on the data to predict performance from motor images at high speed. The deep NN is then used for fast solution of multi-objective optimization problems. Numerical examples demonstrate the effectiveness of the proposed method.

KEYWORDS: Deep Learning, CNN, Multi-objective Optimization, CMA-ES, NSGA-II, PM Motor

I. INTRODUCTION

In recent years, the demand for electric vehicles has increased due to environmental issues such as global warming and air pollution. Reflecting this trend, research on permanent magnet (PM) motors for traction of electric vehicles has been active. The development of these motors focuses on maximizing the output torque and minimizing the torque ripple to achieve a PM motor with high efficiency and low vibration and noise.

Parameter optimization (PO) and topology optimization (TO) are two major design methods for optimizing the shape of PM motors [1]. Since the performance of the optimized device highly depends on predetermined geometric parameters, it is often difficult in PO to set appropriate design parameters. In addition, PO has the problem that it is difficult to obtain novel device structure. On the other hand, since TO directly searches for the optimal material distribution, there is no need to set geometric parameters. TO can generate a wide variety of devices with novel structures and superior performance. In addition, multiple characteristics such as average torque, torque ripple, and losses must be considered when optimizing a PM motor. By solving a multi-objective optimization, it is possible to take these characteristics into account simultaneously. For these reasons, multi-objective optimization using TO is effective especially in the initial design phase of PM motors.

In general, TO of a PM motor has more design variables than PO. Genetic algorithms (GA) have been widely used as a method to solve such high-dimensional optimization problems [1-3]. When the number of optimization variables is n, the population size of a GA would be set proportionally to n. On the other hand, the recommended population size of the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) is $O(\log n)$ [4], which would be more suitable for problems with many optimization variables, such as TO. In fact, CMA-ES has been successfully applied to TO of PM motors [5]. However, while CMA-ES has excellent search performance for single-objective optimization, it is still unclear whether the multi-objective version of CMA-ES outperforms NSGA-II [6] which is one of the most widely used methods for multi-objective GA with smaller number of population size for the former [7], [8].

This paper proposes a multi-objective optimization method for PM motors using CMA-ES and deep learning. In the proposed method, training data is generated by performing a single-objective optimization of a PM motor using CMA-ES multiple times. The data obtained during the optimization process is used to train a convolutional neural network (CNN) which predicts the performance from the input motor images. The trained CNN is used to predict the performance of PM motors to solve the multi-objective optimization problem at high speed and obtain motor shapes with high performance and a variety of features, where NSGA-II is used for optimization in this study. The CNN trained by the above procedure can be applied to multi-objective optimization for different

objective functions and constraints.

II. NGNET METHOD

In this study, the shape of the magnetic core of the rotor of a PM motor is optimized. The NGnet (Normalized Gaussian Network) method is used as the topology optimization method [1, 2], which is based on the shape function

$$y(\boldsymbol{x}, \boldsymbol{w}) = \sum_{i=1}^{N} w_i \, b_i(\boldsymbol{x}) \tag{1}$$

where $\boldsymbol{x}$ is the coordinates in the design region, w_i is the weighting coefficient, and N is the total number of Gaussian basis functions whose centers are uniformly placed in the design region, and $b_i(\boldsymbol{x})$ denotes the normalized Gaussian basis function defined as

$$b_i(\boldsymbol{x}) = G_i(\boldsymbol{x}) \Big/ \sum_{k=1}^{N} G_k(\boldsymbol{x}) \tag{2}$$

The Gaussian basis function in two dimensions used in this study is given by

$$G_k(\boldsymbol{x}) = \frac{1}{2\pi\sigma_0^2} \exp\left[-\frac{|\boldsymbol{x} - \boldsymbol{\mu}_k|^2}{2\sigma_0^2}\right] \tag{3}$$

where $\sigma_0, \boldsymbol{\mu}_k$ are the standard deviation and central position of the basis function, respectively. In this study, to determine the shape of the magnetic core of the rotor, material attribute M_e finite element e is determined from

$$M_e = \begin{cases} iron & y(\boldsymbol{x}, \boldsymbol{w}) \geq 0 \\ air & y(\boldsymbol{x}, \boldsymbol{w}) < 0 \end{cases} \tag{4}$$

The flow of the NGnet method is shown in Figure 1.

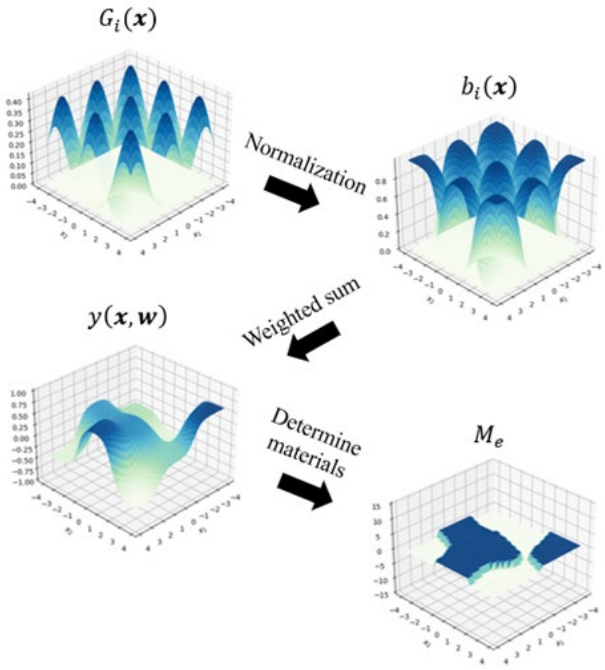


Fig. 1. Flow of NGnet method

III. CMA-ES

CMA-ES is used for pre-optimization to collect training data for the CNN. The population is generated according to a multivariate normal distribution $\mathcal{N}(\boldsymbol{m}, \sigma^2 \mathrm{C})$. The search for the optimal solution is performed by updating not the parameters of the population itself, but the parameters of the multivariate normal distribution: center $\boldsymbol{m}$, covariance matrix C, and step size σ. Because it performs global search with relatively small population size, it is suitable for optimization with field analysis that needs large computing cost. The optimization of CMA-ES proceeds as follows. First, the parameters are initialized. Next, the population is generated. Let the individuals be $\boldsymbol{x}_k$ and $\mathrm{C} = \mathrm{BD}^2\mathrm{B}^{\mathrm{T}}$ is the eigenvalue decomposition of the covariance matrix C, where B is an orthogonal matrix whose columns consist of an orthonormal basis of eigenvectors, and D is a diagonal

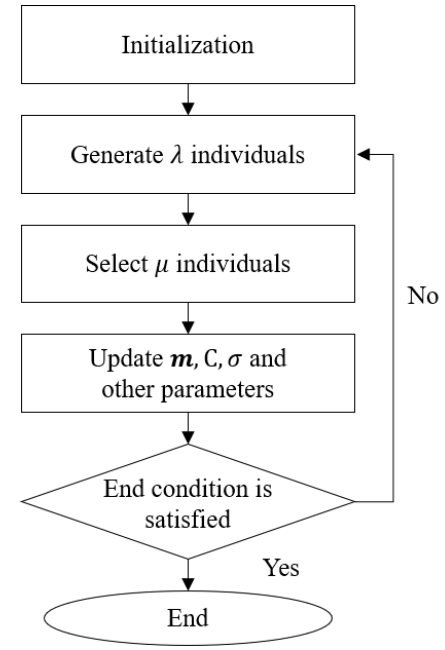


Fig. 2. Algorithm of CMA-ES

matrix consisting of the square roots of the positive eigenvalues. We generate the individuals according to

$$\boldsymbol{x}_k = \boldsymbol{m} + \sigma \mathrm{B}\mathrm{D}\boldsymbol{z}_k \sim \mathcal{N}(\boldsymbol{m}, \sigma^2 \mathrm{C}) \tag{5}$$

$$\boldsymbol{z}_k \sim \mathcal{N}(\boldsymbol{0}, \mathbf{I}) \tag{6}$$

The sampled populations are evaluated for the objective function, and the top μ individuals with the highest evaluation values are used to update the center as follows:

$$\boldsymbol{m}^{(g+1)}(\boldsymbol{x}) = \boldsymbol{m}^{(g)} + c_{\mathrm{m}} \sum_{i=1}^{\mu} w_i \, (\boldsymbol{x}_{i:\lambda}^{(g+1)} - \boldsymbol{m}^{(g)}) \qquad (7)$$

Covariance matrix C and step size σ and other parameters can be found in [4]. Sampling, population evaluation, and parameter updating are repeated until a predetermined number of generations is reached. The optimization flow is shown in Fig. 2.

IV. CNN FOR PREDICTION OF MOTOR PERFORMANCE

A. *Structure of CNN*

A regression model is constructed using a CNN for motor performance prediction which works much faster than finite element method (FEM). The CNN predicts average torque and torque ripple from the cross-sectional images of a PM motor [9]. The input image set is composed of normalized 224 x 224 dimensional RGB images. As shown in Figure 3, the convolutional layers of the VGG16 model [10] are used for transfer learning, and layers deeper than the ninth layer are re-trained [11]. To prevent overlearning, a dropout was performed between the two layers shown in Figure 3 as Dense1 and Dense2.

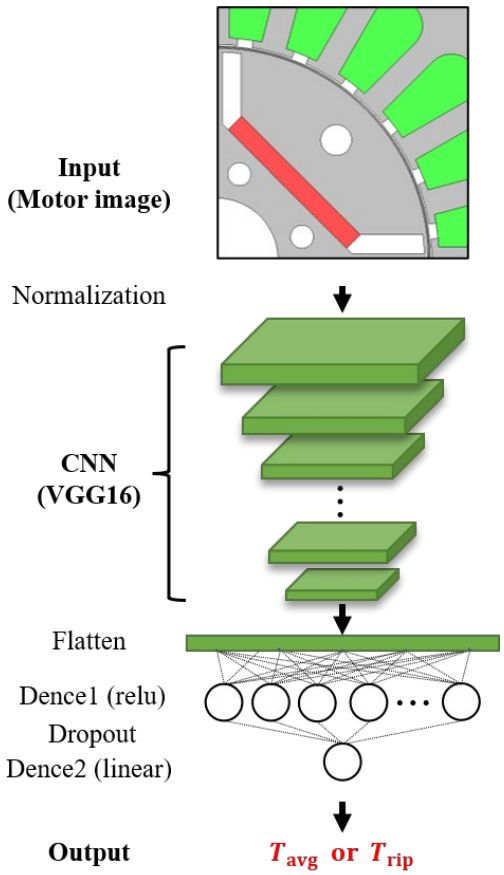


Fig. 3. Structure of CNN

B. *Preliminary Optimization to Generate Training Data*

To generate the training data for the CNN, single-objective optimizations are performed for multiple times in advance, which is here defined by

$$F = -\alpha \frac{T_{\mathrm{avg}}}{T_{\mathrm{avg}}^{\mathrm{ref}}} + (1-\alpha)\frac{T_{\mathrm{rip}}}{T_{\mathrm{rip}}^{\mathrm{ref}}} \rightarrow \min. \qquad (10)$$

where T_{avg}, T_{rip} are the average torque and torque ripple, the latter being defined as $T_{\mathrm{rip}} = T_{max} - T_{min}$. Moreover, we set $T_{\mathrm{avg}}^{\mathrm{ref}} = 1.63(\mathrm{Nm}), T_{\mathrm{rip}}^{\mathrm{ref}} = 1.04(\mathrm{Nm})$ with reference to the D model proposed by Institute of Electrical Engineers of Japan. This optimization assumes symmetry so that 60 Gaussian basis functions are placed in half of the design domain. Once the shape of one half of the domain is determined, the shape of the other half can be determined from the symmetry. CMA-ES was used to solve (10), and FEM was used for field analysis. Table 1 summarizes the optimization settings. By changing the value of the weighting coefficient α as $0.1, 0.3, 0.5, 0.6, 0.7$ and 0.9 in equation (10), the obtained 38,784 data, which were used to train the CNN.

Table 1. Settings of pre-optimization

Number of generations	100
Number of individuals	64

We performed FE analysis of the magnetostatic field in the quarter region of the D-model by changing the mechanical angle every 6 degrees obtain the training set. The computing time was about 86,400 sec. using 16 parallel computers with Intel(R) Xeon(R) E5-2667 v3.

V. OPTIMIZATION

After constructing the trained CNN, we perform multi-objective optimization in which CNN predicts the performance of generated PM motors. The optimization problem is defined by

$$T_{\mathrm{avg}} \rightarrow \max. \quad T_{\mathrm{rip}} \rightarrow \min. \qquad (11)$$

We employed NSGA-II [5] for the multi-objective optimization method based on GA. The optimization

Table 2. Settings of multi-optimization

Number of parents	96
Number of children	96
Number of generations	300
Crossover method	Simulated Binary Crossover
Mutation method	Polynomial Mutation

settings are listed in Table 2.

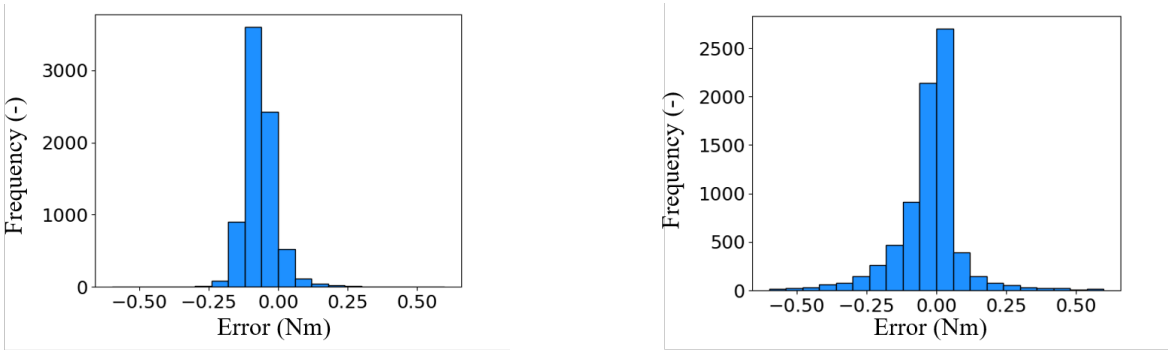
VI. RESULTS

A. Training Results

The 38,784 data generated by the preliminary optimization were split 6:2:2 and used as training, validation, and test data, respectively. The training settings are listed in Table 3. The loss function is the mean absolute error (MAE). Predictions on the test data were performed on the trained CNNs. The error distributions between the FEM results and the CNN predictions for $T_{\mathrm{avg}}, T_{\mathrm{rip}}$ are shown in Figure 4. These MAEs and correlation coefficients are summarized in Table 4. From Table 4 and Figure 4, we concluded that the prediction accuracy of the CNN was sufficiently high to test the validity of the proposed method especially for T_{avg}, whereas the prediction accuracy for T_{rip} was lower than that for T_{avg}. The result of multi-objective optimization using these trained CNNs will be described below.

Table 3. Settings of training CNN estimator

Number of data	38784
Batch size	256
Number of epochs	60
Loss function	MAE



(a) Average torque T_{avg} (b) Torque ripple T_{rip}

Fig. 4. Error distribution for test data

Table 4. Prediction accuracy for test data

	Average torque T_{avg}	Torque ripple T_{rip}
MAE (Nm)	0.076	0.097
Correlation coefficients (-)	0.995	0.901

B. Optimization Results

The multi-objective optimization was performed using CNN and NSGA-II. Figure 5 shows the resulting Pareto solutions whose performances are evaluated by CNN and the corresponding values re-computed by FEM, as well as the performance values for the six optimal solutions obtained by the preliminary optimization. The motor shapes corresponding to Pareto solutions A, B, and C are shown in Figure 6, and the performance values of these motors are summarized in the caption of Table 5. Here, CNN and FEM

refer to the values predicted by CNN and the values obtained from the FEM analysis. That is, the latter values are the ground truth. From Figures 5 and 6, it can be seen that the multi-objective optimization yielded PM motor shapes with a variety of featured and characteristics. It can also be seen that these shapes have characteristics that could not be obtained by the preliminary optimization. Table 6 shows the prediction accuracy for the optimal solutions. Since the population of optimal solutions contains shapes that are very close in both shape and performance value, the distance d between individuals is measured from

$$d_{ij} = \sqrt{\sum_{k=1}^{N} (w_{ik} - w_{jk})^2} \tag{12}$$

where w_{ik} denotes the k-th design variable of i-th motor geometry. For those individuals for which $d \leq d_{th}$, only one of them is included in the calculation, and here we assume that $d_{th} = 0.3$.

As shown in Table 6 and Figure 5, there is no significant difference between the predictions of CNN and FEM results for the optimal solutions, indicating that the optimization is performed with sufficient prediction performance. As with the prediction results for the test data, the error for T_{rip} was slightly larger than that for be T_{avg}. Therefore, the

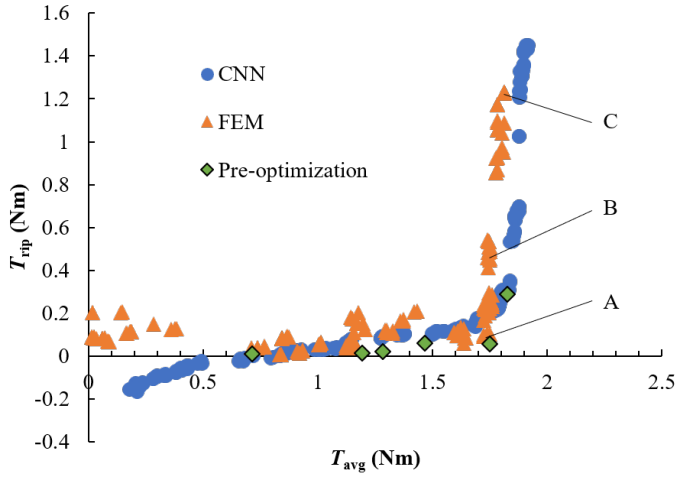


Fig. 5. Analysis results of optimal solutions by FEM and optimal solutions of pre-optimization

A B C

Fig. 6. Motor shapes corresponding to Pareto solutions A, B and C

Table 5. Performances of motors shown in Fig. 6

		A	B	C
CNN	T_{avg} (Nm)	1.688	1.857	1.899
	T_{rip} (Nm)	0.139	0.646	1.427
FEM	T_{avg} (Nm)	1.722	1.748	1.811
	T_{rip} (Nm)	0.093	0.499	1.228

Table 6. Prediction accuracy of optimal solutions

	Average torque T_{avg}	Torque ripple T_{rip}
MAE (Nm)	0.070	0.097
Correlation coefficients (-)	0.989	0.976

Table 7. Comparison of computation time per generation

	Time (s)
FEM	514.592
CNN	27.319

prediction accuracy of T_{rip} needs to be improved to obtain more accurate Pareto solutions.

C. Computing Cost

Table 7 lists the computation time per generation for each method when FEM and CNN are used for performance analysis of the generated PM motors in multi-objective optimization using NSGA-II. In this study, 16 threads of CPU (Intel(R) Xeon(R) E5-2637v4) were used for FEM, and one GPU (NVIDIA Tesla V100 PCIE 16GB) was used for CNN. The computational speed was approximately 18.8 times faster when using CNN than when using FEM, indicating a significant speedup. Although training a CNN is computationally expensive, once a CNN is trained to predict different properties, multi-objective optimization with different objective functions and constraints can be effectively performed.

VII. CONCLUSION

In this paper, we proposed a method to obtain a variety of shapes by performing fast multi-objective optimization using a CNN trained on data obtained by single-objective optimizations with CMA-ES. The proposed method produces a variety of motor shapes with characteristics that cannot be obtained by the preliminary optimization through fast and accurate multi-objective optimization. The trained CNN can be applied to multi-objective optimization with additional objective variables, such as iron loss and permanent magnet area, and with different constraints. A future task is to improve the prediction accuracy, especially for torque ripple, to further enhance the optimization performance.

REFERENCES.

[1]. T. Sato and K. Watanabe and H. Igarashi, Multimaterial topology optimization of electric machines based on normalized Gaussian network, IEEE Trans. Magn. 51 no. 3 (2015), 7202604.

[2]. S. Sato and T. Sato and H. Igarashi, Topology optimization of synchronous reluctance motor using normalized Gaussian network, IEEE Trans. Magn. 51 no. 3 (2015), 8200904.

[3]. S. Hiruma and M. Ohtani and S. Soma and Y. Kubota and H. Igarashi, Novel hybridization of parameter and topology optimizations: application to permanent magnet motor, IEEE Trans. Magn. 57 no. 7 (2021), 8204604.

[4]. N. Hansen, The CMA Evolution Strategy: A Tutorial, arXiv:1604.00772 (2016).

[5]. S. Hayashi and H. Igarashi, Parameter-topology hybrid optimization of electric motor with multiple permanent magnets, Int. J. Appl. Electromag. Mech., vol. 71, no. S1, (2022), S245-S255.

[6]. K. Deb and A. Pratap and S. Agarwal and T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evolutionary Computation, 6 no. 2 (2002), 182-197.

[7]. C. Igel and N. Hansen and S. Roth, Covariance matrix adaptation for multi-objective optimization, Evolutionary Computation, 15 no. 1 (2007), 1-28

[8]. J. Nagar and D. H. Werner, A Comparison of three uniquely different state of the art and two classical multiobjective optimization algorithms as applied to electromagnetics, IEEE Trans. Antennas and Propagation, 65 no. 3 (2017), 1267-1280

[9]. H. Sasaki and H. Igarashi, Topology optimization accelerated by deep learning, IEEE Trans. Magn., 55 no. 6 (2019), 7401305.

[10]. K. Simonyan and A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv:1409.1556 (2015).

[11]. J. Asanuma and S. Doi and H. Igarashi, Transfer learning through deep learning: application to topology optimization of electric motors, IEEE Trans. Magn., 56, no.

3 (2020), 7512404.