Fast Multi-objective Optimization of Electromagnetic Devices Using Adaptive Neural Network Surrogate Model

Hayaho Sato[1], Hajime Igarashi[1]

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

This paper presents a fast population-based multi-objective optimization of electromagnetic devices using an adaptive neural network surrogate model. The proposed method does not require any training data or construction of a surrogate model before the optimization phase. Instead, the neural network surrogate model is built from the initial population in the optimization process, and then it is sequentially updated with high-ranking individuals. All individuals were evaluated using the surrogate model. Based on this evaluation, high-ranking individuals are re-evaluated using high-fidelity electromagnetic field computation. The suppression of the execution of expensive field computations effectively reduces the computing costs. It is shown that the proposed method works two to four times faster-maintaining optimization performance than the original method that does not use surrogate models.

Index Terms—**Genetic algorithm, IPM motor, Machine learning, Magnetic shield, Shape optimization.**

I. Introduction

Design optimization based on electromagnetic field computation is now an indispensable process for the development of electric and electronic devices. In particular, population-based stochastic optimization methods, such as genetic algorithm (GA) and particle swarm optimization, are more effective than other methods, such as gradient-based methods, for the problems encountered in the design process, which contain complicated constraints and multimodal objective functions [1], [2]. Population-based methods have numerous merits: (a) global search ability, (b) flexible treatment of complicated constraints, (c) easy extension to multi-objective problems using, for example, NAGA-II [3] and SPEA-II [4], and (d) robust optimization [5]. However, this method has a relatively large computing cost, which becomes problematic when the objective function evaluation is computationally expensive. In particular, when the performance of an electric device is evaluated by the finite element method (FEM) or other field computing methods, the computing cost for population-based optimization can be unacceptably long. To overcome this difficulty, a surrogate model was employed for the fast evaluation of the objective and constraint functions. In this method, surrogate models based on artificial neural networks (NNs) [6–9], kriging method [10], [11], and response surface [12–14] are built to make approximate computation faster than the field computations. The surrogate-based optimization can be classified into fixed (offline) and adaptive (online) methods [15], [16]. In the former approach, the surrogate model is prepared based on a set of training data before the optimization phase [17]. Once the surrogate model is constructed, the function evaluation during the optimization process can be performed promptly. When the number of design variables is relatively small, this method is effective. However, its performance was determined based on the quality of the prepared surrogate model. It becomes rapidly difficult to ensure sufficient sampling as the degree of freedom in the problem increases. If the sampling is insufficient in the training phase, the optimization performance is unsatisfactory. A remedy for this problem is to conduct intensive sampling around promising points. This can be done using the adaptive method, in which the surrogate model is updated with additional sampling data during the optimization process.

Taran et and D. G. proposed an adaptive surrogate model using the kriging method for the optimal design of electric motors [16]. In this study, an inexpensive Pareto front was built using the surrogate model for a multi-objective problem with respect to the design parameters. The individuals on the inexpensive Pareto front are then evaluated using the FEM to build the expensive Preto front. This method would be effective for obtaining approximated solutions, while its performance would not always be satisfactory because the inexpensive Pareto front that governs the evolution direction would inherently have errors owing to the surrogate model.

In this study, we proposed a different adaptive surrogate-based optimization method for multi-objective problems. In this method, we updated the surrogate model based on the artificial NN, not only from the Pareto front but also from lower-ranking individuals. It will be shown that this extended sampling is important for maintaining the optimization quality. The proposed method based on the extended sampling is shown to make the optimization process faster keeping the performance in the optimized devices. Moreover, the proposed method is shown to be valid not only for parametric but also for topology optimization problems. Because the optimization variables in the topology optimization tends to be more than those in the parameter optimization, the former needs more computing cost. The proposed method particularly effective to make the topology optimization faster without loss of optimization quality. Although the proposed method is for multi-objective optimization, it can also be applied to single-objective problems, which will be discussed in the Appendix.

II. Adaptive Surrogate-based Optimization Method

The nomenclature of parameters is summarized in Table I.

A. Shape Optimization Based on GA

As a population-based optimization method, we adopted the GA, whose algorithm is summarized as follows:

1. For the initial population, individuals were randomly

TABLE I
NOMENCLATURE

Parameter	Quantity	Parameter	Quantity
$\boldsymbol{x}$	coordinate vector	$\mathcal{P}$	population
$\boldsymbol{w}$	design variable	$\mathcal{C}$	children
n	number of design variables	$\mathcal{D}$	stored data
σ	standard deviation of Gaussian basis function	$\mathcal{M}$	surrogate model
$\boldsymbol{\mu}$	average vector of Gaussian basis function	n_{pop}	size of population
F, F_1, F_2	objective function	n_{c}	number of children
$d_{x,i}, d_{y,i}$	width of magnetic shield and air gap	k	number of highest ranks to be re-evaluated
D_x, D_y	width of design region (magnetic shielding)	p_{r}	re-evaluation ratio for individuals in lower rank
$\boldsymbol{B}_i$	magnetic flux density in i-th element	$g_{\max}$	maximum number of generation
$\lvert\boldsymbol{B}\rvert_{\text{avg}}$	average of magnetic flux density	f	ground truth
n_e	number of elements in target region	$\hat{f}$	inferred value
T_{avg}	average torque	N_{data}	number of training data
T_{rip}	torque ripple	n_{p}	number of parents (single-objective)
$T_{\max}$	maximum torque	p_{a}	threshold for fine tuning execution (single-objective)
$T_{\min}$	minimum torque	p_{c}	classification threshold (single-objective)
$T_{\text{avg}}^{\text{ref}}$	reference value of average torque (single-objective)	Ω	design region
$T_{\text{rip}}^{\text{ref}}$	reference value of torque ripple (single-objective)		

generated and analyzed by FEM to evaluate their fitness.
2. Parents are selected.
3. Children are generated from the parents by crossover and mutation.
4. Children were analyzed using FEM for selection. The selected children returned to the population.
5. The algorithm terminates if the maximum number of generations is attained. Otherwise, return to Step 2.

In parameter optimization, the genes comprise design parameters, such as material size, position, angle, and material type. In topology optimization, the material distribution is determined, allowing the generation and elimination of holes such that the cost function is minimized. Here, we adopt the NGnet-based on/off method, which has been shown to be fairly suitable for the optimization of electric apparatus, such as electric motors [18] and wireless power transfer devices [19]. In this method, the material distribution (for example, magnetic core/air) is determined from the sign of the shape function $y(\boldsymbol{x}, \boldsymbol{w})$ defined by

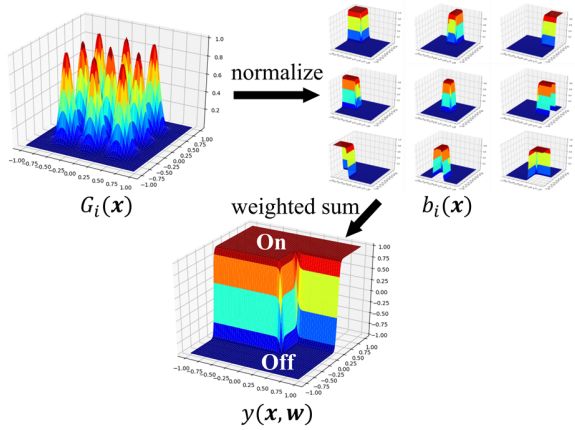


Fig. 1. NGnet-based on/off method.

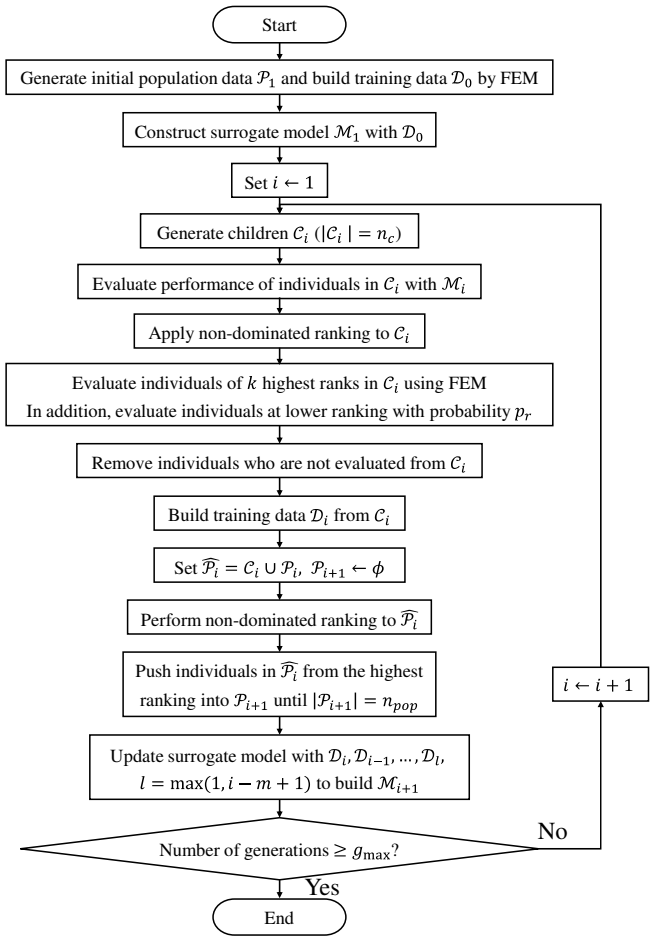


Fig. 2. Algorithm of adaptive surrogate model for multi-objective optimization, where k, m are hyperparameters given in the text.

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

$$b_i(\boldsymbol{x}) = \frac{G_i(\boldsymbol{x})}{\sum_{j=1}^{n} G_j(\boldsymbol{x})}, \tag{1b}$$

$$G_i(\boldsymbol{x}) = \frac{1}{2\pi\sigma} \exp\left\{-\frac{(\boldsymbol{x} - \boldsymbol{\mu}_i)^T(\boldsymbol{x} - \boldsymbol{\mu}_i)}{2\sigma^2}\right\}. \tag{1c}$$

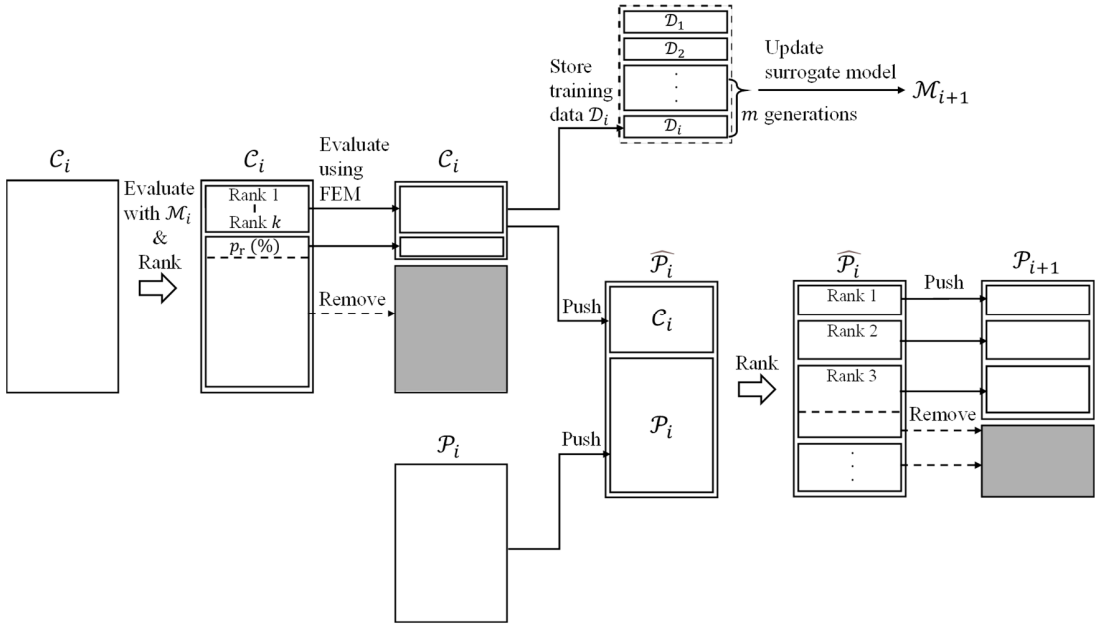


Fig. 3. Overview of algorithm of adaptive surrogate model for multi-objective optimization.

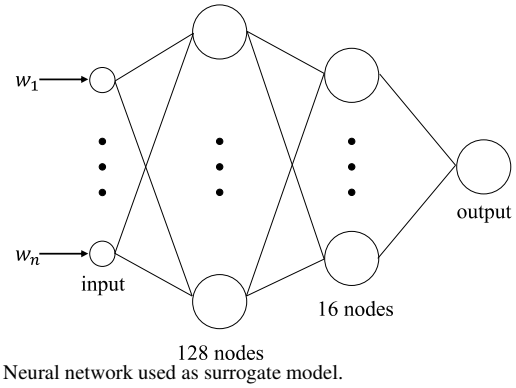


Fig. 4. Neural network used as surrogate model.

The NGnet on/off method is schematically shown in Fig. 1. If $y(\boldsymbol{x}, \boldsymbol{w}) > 0$, the material is set to the magnetic core ("on"), and otherwise, it is set to air ("off"). Here, the weight coefficients $\boldsymbol{w} = \{w_1, w_2, \ldots, w_n\}$ are the design parameters of the gene.

B. Adaptive NN Surrogate Model

Fig. 2 shows the algorithm of the proposed adaptive surrogate model for multi-objective optimization, which is schematically depicted in Fig. 3. In the first stage of this method, the initial population $\mathcal{P}_1 = \{\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{n_{\mathrm{pop}}}\}$ is randomly generated, and its performance is evaluated by field computation, where FEM is used to build the initial training data $\mathcal{D}_0 = \{(\boldsymbol{w}_1, f_1), (\boldsymbol{w}_2, f_2), \ldots, (\boldsymbol{w}_{n_{\mathrm{pop}}}, f_{n_{\mathrm{pop}}})\}$. Subsequently, the surrogate model $\mathcal{M}_1$ is constructed based on $\mathcal{D}_0$, and n_c children, $\mathcal{C}_1$, are generated by mutation from $\mathcal{P}_1$ and they are ranked based on the approximate evaluation using $\mathcal{M}_1$. The individuals of the highest k ranks in $\mathcal{C}_1$ are re-evaluated by FEM to build new data $\mathcal{D}_1$ and the others are eliminated. Using $\mathcal{D}_1$ (data sets $\mathcal{D}_i, \mathcal{D}_{i-1}, \ldots, \mathcal{D}_l$, $l = \max(1, i - m + 1)$, in general), the surrogate model is updated to obtain $\mathcal{M}_2$ ($\mathcal{M}_{i+1}$). The new population, $\mathcal{P}_2$ is built from $\mathcal{C}_1$ and $\mathcal{P}_1$.

The population $\mathcal{P}$ generated in the above procedure is occupied by the individuals evaluated by FEM, and the surrogate model $\mathcal{M}$ is constructed from the selected children in the latest m generations through the FE computations. The computing cost is reduced by the limited number of FE evaluations for individuals with the highest k ranks. It will be shown that the resultant solutions can be unsatisfactory if the surrogate model is constructed only from the highest-ranking individuals, composing the tentative Pareto front.

In this study, we adopted an artificial NN for the surrogate model $\mathcal{M}$, which can be replaced by other methods, such as kernel regression and kriging. One of the advantages of NN is that it works well for a relatively large number of design variables. In this study, we apply the proposed method to topology optimization, which can have more than 100 variables. Fig. 4 shows the structure of the NN. The input and output of the NN are the design parameters and the predicted value of the device performance, respectively. The ReLU function $f(x) = \max(0, x)$ is used as the activation function. In addition, in each layer, we introduce a dropout with a probability of 0.2.

As shown in Figs. 2 and 3, the surrogate mode, NN, is updated for every generation. In this update, the initial NN weights are set to the weights in the previous generation.

The loss function L for training is:

$$L = \frac{1}{N_{\mathrm{data}}} \sum_{i=1}^{N_{\mathrm{data}}} \left(f_i - \hat{f}_i\right)^2. \tag{2}$$

III. Application to Parameter Optimization

In Sections III and IV, we describe numerical examples for the parameter and topology optimization problems, respectively. In these computations, we used an Intel (R) Xeon (R) CPU (3.2 GHz, 8 cores, 16 threads).

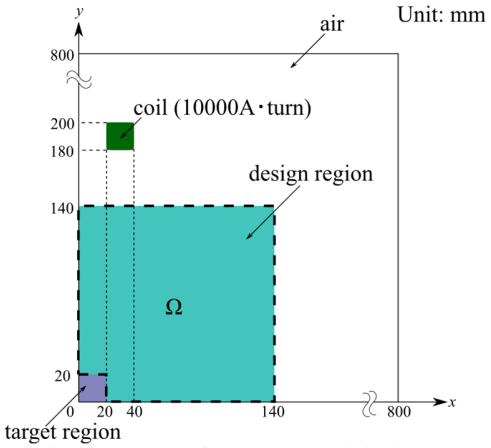


Fig. 5. Magnetic shield problem for parameter optimization (mm).

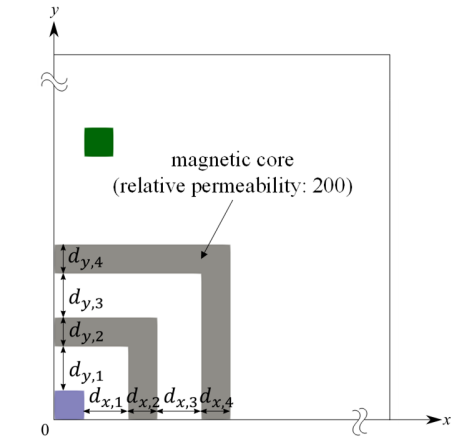


Fig. 6. Design parameters for double magnetic shields.

A. *Optimization of Magnetic Shielding Structure*

We consider the optimization of the magnetic shield shape placed near a coil, as shown in Fig. 5 [20] to minimize the magnetic flux density averaged over the target region, $|\boldsymbol{B}|_{\text{avg}}$, and simultaneously minimize the magnetic core area S inside the design region Ω. Fig. 6 shows the design parameters for the assumed double-shield structur e, which are represented by the optimization variables $\boldsymbol{w} = \{w_{x,1}, \dots, w_{x,4}, w_{y,1}, \dots, w_{y,4}\}$ in such a way that

$$d_{x,i} = \frac{D_x w_{x,i}}{\sum_{j=1}^{4} w_{x,j}}, d_{y,i} = \frac{D_y w_{y,i}}{\sum_{j=1}^{4} w_{y,j}}, i = 1, \dots, 4. \quad (3)$$

B. *Problem Definition*

The multi-objective optimization problem for F_1, F_2 is defined as follows:

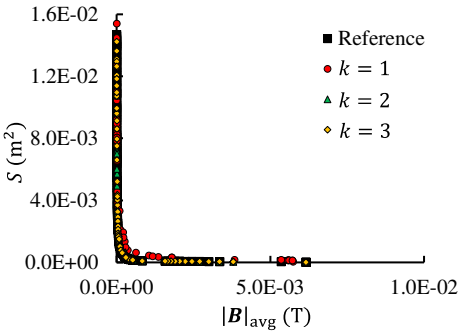
$$F_1(\boldsymbol{w}) = |\boldsymbol{B}|_{\text{avg}} = \frac{1}{n_e}\sum_{i=1}^{n_e} |\boldsymbol{B}_i| \rightarrow \min., \quad (4a)$$

$$F_2(\boldsymbol{w}) = S \rightarrow \min. \quad (4b)$$

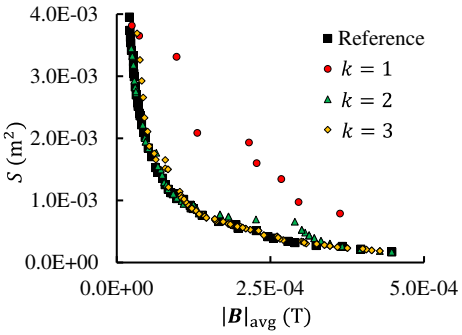
The optimization settings are summarized in Table II.

TABLE II
OPTIMIZATION SETTINGS IN MULTI-OBJECTIVE OPTIMIZATION OF MAGNETIC SHIELD

Parameters	Value
n	8
n_{pop}	160
n_{c}	80
$g_{\max}$	200
m	1
Number of epochs for training of $\mathcal{M}_1$	50
Number of epochs for training of $\mathcal{M}_i, i \geq 2$	2
p_{r}	10%
Design variable	$0 \leq w_{x,i}, w_{y,i} < \infty$ $(0 \leq d_{x,i}, d_{y,i} \leq 120)$



(a) Entire solutions.



(b) Solutions near origin.

Fig. 7. Pareto solutions for magnetic shielding problem.

Because F_2 can be computed directly from the design parameters, we build an NN only for F_1. The search for the Pareto solutions is performed based on NSGA-II, which is a widely used multi-objective GA [3]. For crossover, we adopt Simulated Binary Crossover [21] with a crossover probability of 90% and a crossover parameter η_{d}, which is a parameter that controls the range of children generated, set to 2.

During the optimization process, we update the NN using the proposed method in Fig. 2 and 3 with the parameters summarized in Table II. For training the NN during the optimization process, which corresponds to $\mathcal{M}_i$, $i \geq 2$, the number of epochs is set to be much smaller than that for the training of $\mathcal{M}_1$ in the initial step. This enables us to avoid overfitting and shorten the training time. We consider three different rank ranges for the construction of the NN, $k = 1, 2, 3$,

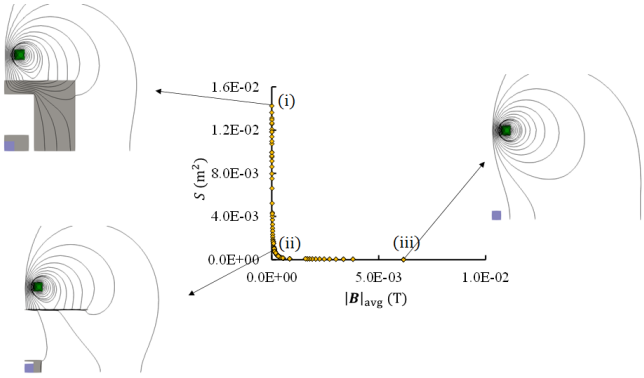


Fig. 8. Shapes for Pareto solutions when $k = 3$ for multi objective optimization of magnetic shield. Black line: magnetic flux.

TABLE III
REDUCTION OF FUNCTION CALLS AND COMPUTING TIME FOR MAGNETIC SHIELD PROBLEM

	Num. of function calls for FEM (a) (-)	Reduction in (a) (%)	Optimization time (b) (s)	Reduction in (b) (%)
Reference	16160	100.0	340.2	100.0
$k = 1$	2415	14.9	110.6	32.5
$k = 2$	4185	25.9	165.2	48.6
$k = 3$	4446	27.5	157.7	46.3

to test the optimization performance of the proposed method.

C. Optimization Result

Fig. 7 shows the Pareto solutions obtained by the proposed method and a conventional approach, represented by "Reference," in which all the individuals are evaluated by FEM. It can be seen that the Pareto front obtained by the proposed method approaches "Reference" as k increases. In particular, when $k = 3$, the former becomes almost identical to the latter. It is stressed that the NN constructed only from the tentative Pareto front, which corresponds to $k = 1$, does not work well in this problem.

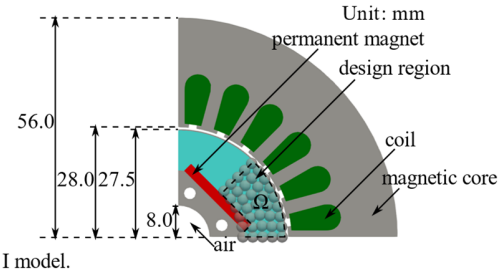
Fig. 8 shows the typical solutions on the Pareto front when $k = 3$. Solution (i) significantly reduced $|\boldsymbol{B}|_{\text{avg}}$ with thick shields. In contrast, solution (iii) has no magnetic shield, where $S = 0$. Solution (ii) has a suitable balance, $|\boldsymbol{B}|_{\text{avg}}$ is reduced with the magnetic shield of a small area.

The number of function calls for the FE analysis and computing time are summarized in Table III. The number of function calls was reduced to approximately 25% even when $k = 3$. The computing time was reduced to approximately 50% when $k = 3$, where we include the training time of the NN.

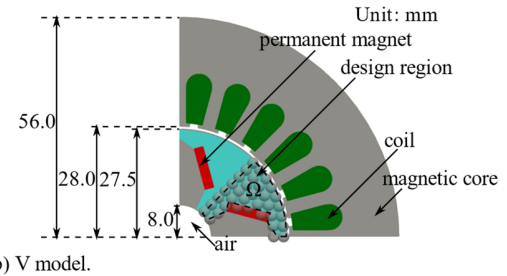
IV. APPLICATION TO TOPOLOGY OPTIMIZATION

A. Optimization of Motor Structure

Fig. 9 shows the internal permanent magnet (IPM) motor models for topology optimization. The IPM motor models, which are based on the IEEJ D model [22], shown in (a) and (b), have I-shaped and V-shaped permanent magnets. Hereafter, they are referred to as I and V models, respectively. The magnetization was assumed to be normal to the wide surface of the magnets. The BH curve of the magnetic core is shown in Fig. 10. The other settings are summarized in Table IV. Although we consider here I and V models, we can apply the



(a) I model.



(b) V model.

Fig. 9. IPM motors for topology optimization (mm), $\sigma = 1.5$ mm.

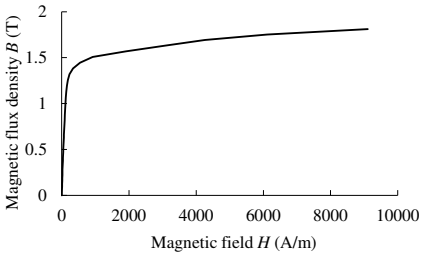


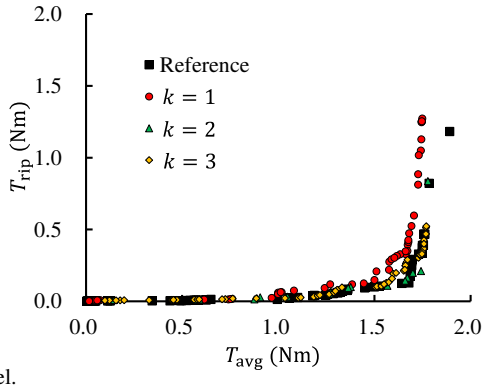
Fig. 10. BH curve of magnetic core (50A400).

TABLE IV
MACHINE PARAMETERS

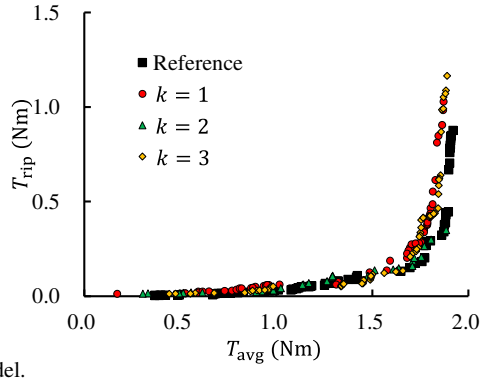
Parameters	Value
Thickness (mm)	65.0
Current amplitude (A)	3.0
Number of turns of coil (turns)	35
Current phase angle (°)	20
Residual flux density of permanent magnets (T)	1.4
Material of magnetic core	50A400

TABLE V
OPTIMIZATION SETTINGS IN MULTI-OBJECTIVE OPTIMIZATION OF IPM MOTORS

Parameters	(a) I model	(b) V model
n	37	35
n_{pop}	740	700
n_{c}	370	350
$g_{\max}$	200	200
m	10	10
Number of epochs for training of $\mathcal{M}_1$	50	50
Number of epochs for training of $\mathcal{M}_i, i \geq 2$	5	5
p_{r}	10%	10%
Design variable	$-\infty < w_i < \infty$	$-\infty < w_i < \infty$

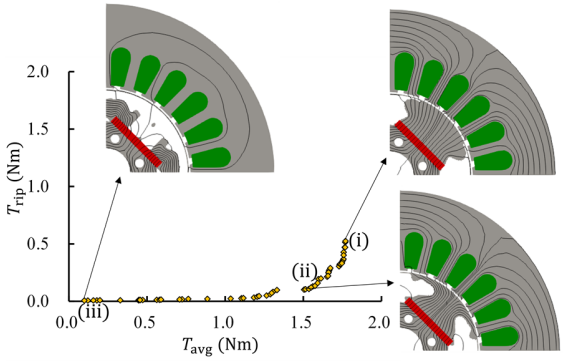


(a) I model.

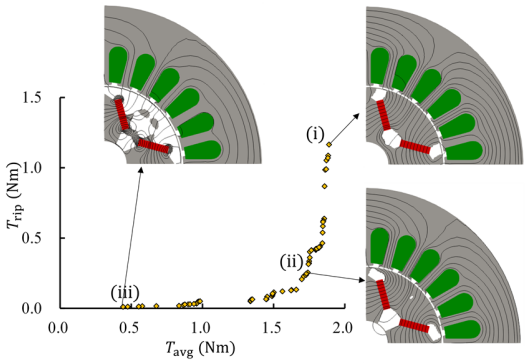


(b) V model.

Fig. 11. Pareto solutions for multi objective optimization of IPM motors.



(a) I model.



(b) V model.

Fig. 12. Pareto solutions for multi objective optimization of IPM motors.

TABLE VI
REDUCTION OF FUNCTION CALLS AND COMPUTING TIME FOR OPTIMIZATION OF IPM MOTORS

	(a) I model				(b) V model			
	Num. of function calls for FEM (a) (-)	Reduction in (a) (%)	Optimization time (b) (s)	Reduction in (b) (%)	Num. of function calls for FEM (a) (-)	Reduction in (a) (%)	Optimization time (b) (s)	Reduction in (b) (%)
Reference	74740	100.0	74631.1	100.0	70700	100.0	67639.2	100.0
$k = 1$	11379	15.2	12349.0	16.5	11909	16.8	11537.0	17.1
$k = 2$	16145	21.6	17203.4	23.1	17036	24.1	15492.7	22.9
$k = 3$	24123	32.3	25580.8	34.3	23505	33.2	22099.3	32.7

proposed method to other PM motors such as IPM motors with different magnet configurations, spoke-type PM motors, and surface PM motors.

The distribution of iron and air in Ω was determined by the NGnet on/off method. The Gaussian basis functions are uniformly distributed in Ω, where the circles in Fig. 9 represent their contour lines $|\boldsymbol{x} - \boldsymbol{\mu}_i|^2 = \sigma^2$, $\sigma = 1.5$ mm. Moreover, assuming mirror symmetry, we considered only half of the region.

B. Problem Definition

The multi-objective optimization problem for F_1, F_2 is defined as follows:

$$F_1(\boldsymbol{w}) = T_{\text{avg}} \rightarrow \max., \tag{5a}$$

$$F_2(\boldsymbol{w}) = T_{\text{rip}} \rightarrow \min., \tag{5b}$$

$$T_{\text{rip}}(\boldsymbol{w}) = T_{\max} - T_{\min}. \tag{5c}$$

The optimization variables are the weighting coefficients $\boldsymbol{w}$ in (1a). We trained the two NNs to predict F_1 and F_2 from $\boldsymbol{w}$.

The optimization settings are summarized in Table V. To stabilize the training, we assume a larger value for m compared to that in Section III.

C. Optimization Results

Fig. 11 shows the Pareto solutions, where "Reference" represents the solutions obtained without the surrogate model, as described in Section III. It can be seen that the solutions approximate the distribution of the reference solution when $k = 2, 3$. In Fig. 11 (a), the solutions obtained for $k = 2$ govern the solutions for $k = 3$ near the corner of the Pareto front at approximately $T_{\text{avg}} = 1.8$ Nm. However, the latter had a denser distribution. The same tendency can be observed in Fig. 11 (b). This can be attributed to the stochastic nature of the GA.

Fig. 12 shows the typical Pareto solutions for $k = 3$. In both Fig. 12 (a) and (b), solutions (i) have a narrow magnetic path

concentrating the magnetic flux to increase T_{avg}. In contrast, solution (iii) has extremely large air gaps to reduce T_{rip} and also T_{avg}. Solutions (ii) realize a good balance; the magnetic flux concentration is relaxed with the wider magnetic core surfaces facing the air gap, and simultaneously, the average magnetic flux increases with small air gaps.

The number of function calls for the FEM and the computing time are summarized in Table VI. To evaluate T_{avg} and T_{rip}, FE analysis must be repeated by changing the mechanical angle of the rotor. In this numerical example, we performed 15 FE computations for each individual. Therefore, in contrast to the numerical example in Section III, the computing time for NN training is much shorter than that of FE analysis. Therefore, the reduction ratio for the function calls is almost similar to that of the computing time. As a result, the computing time is reduced to approximately 25% and 35 % when $k = 2, 3$ by the proposed method.

Although we do not consider constraints in the aforementioned optimization problems for simplicity, the proposed method can also be applied to constrained problems. To do so, we would employ constrained NSGA-II [3] or other optimization algorithms for constrained problems instead of NSGA used for the problems discussed above. In constrained NSGA-II, the individuals are classified into feasible and infeasible solutions to be ranked according to the fitness as well as feasibility. This process can easily be implemented to the proposed algorithm shown in Fig. 2 and 3.

V. Discussion

We consider the advantages and disadvantages of the proposed online (adaptive) method compared with the offline (fixed) methods, which is summarized in Table VII. When using the offline method, the optimization results highly depend on the surrogate model constructed from the sampled data before the optimization process. Therefore, a sufficient number of sampling points and their uniform distributions are important for this approach. For this reason, the merit of this method reduces as the number of optimization variables n increases owing to the curse of dimensionality. For example, when $n = 10$, for example, we need approximately 6×10^4 samplings, even assuming three levels for each variable. This may be unacceptable unless we have sufficient computing environments. Smaller samplings would be sufficient if the objective function was sufficiently flat. However, this is not always the case. Conversely, the offline optimization works fast because it does not require field computations during the optimization process. Moreover, the surrogate model can be reused for other objective functions and constraints.

The proposed online method performs adaptive sampling during optimization. This means that the sampling can be concentrated in the promising region, unlike in the offline method. In principle, this offline method can be applied to larger problems. Indeed, from the numerical examples in Section IV, the proposed method is shown to be valid if n is smaller than 40. The surrogate model built in the optimization process can be reused for other objective and constraint functions. One of the weaknesses of this method is the moderate speed-up ratio in comparison with the offline method.

TABLE VII
Comparison of Proposed On-line Method with Off-line Method

	On-line (proposed)	Off-line
Sampling	Adaptive	Uniform sampling before optimization*
Typical problem size	10–50	10
Speed up	moderate	high
Reuse of surrogate model	possible	possible

* Sufficient number of sampling is needed. The surrogate model and optimization results highly depend on the sampled data.

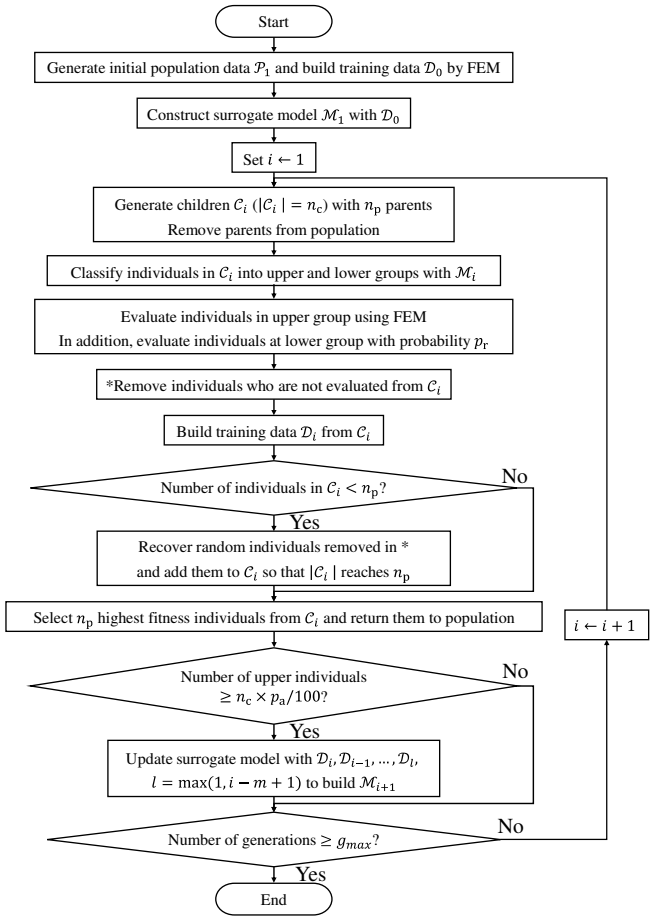


Fig. 13. Algorithm for single objective optimization with online method.

VI. Conclusions

In this study, the adaptive surrogate model was proposed as an effective solution to multi-objective optimization problems. The surrogate model is realized using an NN whose weights are updated during the optimization process. The proposed method allows adaptive sampling of promising individuals. It has been shown that NN works well if it is constructed from individuals with the highest rankings. The proposed method can accelerate optimization without loss of quality. The speed-up rate depends on the cost of the field computation; it increases with the computing cost. Those for magnetic shield and IPM motor

TABLE VIII
OPTIMAL VALUE OF OBJECTIVE FUNCTION FOR SINGLE OBJECTIVE OPTIMIZATION OF IPM MOTORS (FIVE TRIALS)

	(a) I model		(b) V model	
	Average (-)	Standard deviation (-)	Average (-)	Standard deviation (-)
Reference	-1.01	0.00116	-1.43	0.0252
Proposed	-0.991	0.0122	-1.38	0.0428

TABLE IX
REDUCTION OF FUNCTION CALLS AND COMPUTING TIME FOR SINGLE OBJECTIVE OPTIMIZATION OF IPM MOTORS (AVERAGE OF FIVE TRIALS)

	(a) I model				(b) V model			
	Num. of function calls for FEM (a) (-)	Reduction in (a) (%)	Optimization time (b) (s)	Reduction in (b) (%)	Num. of function calls for FEM (a) (-)	Reduction in (a) (%)	Optimization time (b) (s)	Reduction in (b) (%)
Reference	37940.0	100.0	32105.2	100.0	35900.0	100.0	27697.7	100.0
Proposed	19251.8	50.7	18653.4	58.1	17266.4	48.1	14737.5	53.2

problems were approximately two to four, respectively.

ACKNOWLEDGEMENT

This work was supported in part by JSPS KAKENHI Grant Number 21H01301.

APPENDIX

The proposed adaptive surrogate model was applied to a single objective problem. The algorithm for the single-objective problem is illustrated in Fig. 13. For single-objective optimization, it seems sufficient to classify children into upper and lower groups. All individuals in the upper group and p_r (%) in the lower group were evaluated. We set $p_\mathrm{a} = 75$ %, $p_\mathrm{c} = 25$ %, and $p_\mathrm{r} = 10$ %.

Here, we consider the IPM motor models mentioned in Section IV. We perform topology optimization, where the problem is defined as follows:

$$F(\boldsymbol{w}) = -\frac{T_\mathrm{avg}}{T_\mathrm{avg}^\mathrm{ref}} + 0.2\frac{T_\mathrm{rip}}{T_\mathrm{rip}^\mathrm{ref}} \rightarrow \min. \qquad (6)$$

We adopt the real-coded GA realized by the JGG+REXstar [23] with step size 1.0. Each optimization was performed five times, where $g_\mathrm{max} = 200$.

Table VIII summarizes the average and standard deviation of F, and Table IX includes the number of function calls for the FE analysis and computing time. It can be said that the results obtained by the adaptive surrogate model are slightly inferior to the reference results without the surrogate model in terms of average and standard deviation. Conversely, the adaptive method effectively reduces the number of function calls for the FE analysis and computing time. The speed-up ratio is approximately two for this problem; however, it increases with the computing cost for the FE analysis.

REFERENCES

[1] P. Di Barba, "Multiobjective Shape Design in Electricity and Magnetism," Springer, ISBN: 978-9400731462, 2012.

[2] Z. Bingul, "Adaptive genetic algorithms applied to dynamic multiobjective problems," Applied Soft Computing, vol. 7, no. 3, pp. 791–799, 2007.

[3] K. Deb, A. Pratap, and S. Agarwal, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation,* vol. 6, no. 2, pp. 115–148, 2002.

[4] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach," *IEEE Transactions on Evolutionary Computation,* vol. 3, no. 4, pp. 257–271, 1999.

[5] T. Maruyama and H. Igarashi, "Effective robust optimization based on genetic algorithm," *IEEE Transactions on Magnetics*, vol. 44, no. 6, pp. 990–993, 2008.

[6] M. Baldan, P. Di Barba, and B. Nacke, "Magnetic Properties Identification by Using a Bi-Objective Optimal Multi-Fidelity Neural Network," *IEEE Transactions on Magnetics*, vol. 57, no. 6, 2021.

[7] D. Liu and Y. Wang, "Multi-Fidelity Physics-Constrained Neural Network and Its Application in Materials Modeling," *Journal of Mechanical Design*, vol. 141(12):1, doi: 10.1115/1.4044400, 2019.

[8] J. E. Rayas-Sanchez, "EM-based optimization of microwave circuits using artificial neural networks: the state-of-the-art," *IEEE Transactions on Microwave Theory and Techniques*, vol. 52, no. 1, pp. 420–435, 2004.

[9] M. H. Mohammadi, T. Rahman, R. Silva, M. Li, and D. A. Lowther, "A Computationally Efficient Algorithm for Rotor Design Optimization of Synchronous Reluctance Machines," *IEEE Transactions on Magnetics*, vol. 52, no. 3, pp. 1–4, 2016.

[10] E. S. Siah, M. Sasena, J. L. Volakis, P. Y. Papalambros, and R. W. Wiese, "Fast parameter optimization of large-scale electromagnetic objects using DIRECT with Kriging metamodeling," *IEEE Transactions on Microwave Theory and Techniques*, vol. 52, no. 1, pp. 276–285, 2004.

[11] S. Sakata, F. Ashida, and M. Zako, "Structural optimization using Kriging approximation," *Computer Meth. Appl. Mech. Eng.*, vol. 192, no. 78, pp. 923–939, 2003.

[12] Y. S. Ong, P. B. Nair, and A. J. Keane, "Evolutionary optimization of computationally expensive problems via surrogate modeling," *AIAA J.,* vol. 41, no. 4, pp. 687–696, 2003.

[13] P. S. Shin, S. H. Woo, and C. S. Koh, "An optimal design of large scale permanent magnet pole shape using adaptive response surface method with Latin hypercube sampling strategy," *IEEE Transactions on Magnetics*, vol. 45, no. 3, pp. 1214–1217, 2009.

[14] J. Mueller, T. Krityakierne, and C.A. Shoemaker, "SO-MODS: Optimization for high dimensional computationally expensive multi-modal functions with surrogate search," *Proc. 2014 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1092–1099, 2014.

[15] H. Sasaki and H. Igarashi, "Topology Optimization Accelerated by Deep Learning," *IEEE Transactions on Magnetics*, vol. 55, no. 6, art. no. 7401305, 2019.

[16] N. Taran, D. M. Ionel, and D. G. Dorrell, "Two-Level Surrogate-Assisted Differential Evolution Multi-Objective Optimization of Electric Machines Using 3-D FEA," *IEEE Transactions on Magnetics*, vol. 54, no. 11, art. no. 8107605, 2018.

[17] S. Shimokawa, H. Oshima, K. Shimizu, Y. Uehara, J. Fujisaki, A. Furuya, and H. Igarashi, "Fast 3-D Optimization of Magnetic Cores for Loss and Volume Reduction," *IEEE Transactions on Magnetics*, vol. 54, no. 11, art. no. 8400904, 2018.
[18] H. Sasaki and H. Igarashi, "Topology optimization using basis functions for improvement of rotating machine performances," *IEEE Transactions on Magnetics*, vol. 54, no. 3, art. no. 8201504, 2018.
[19] Y. Otomo and H. Igarashi, "A 3-D Topology Optimization of Magnetic Cores for Wireless Power Transfer Device," *IEEE Transactions on Magnetics*, vol. 55, no. 6, art. no. 8103005, 2019.
[20] Y. Hidaka, T. Sato, and H. Igarashi, "Topology Optimization Method Based on On–Off Method and Level Set Approach," *IEEE Transactions on Magnetics*, vol. 50, no. 2, pp. 617–620, 2014.
[21] K. Deb and R. Agrawal, "Simulated Binary Crossover for Continuous Search Space," *Complex Systems*, vol. 9, no. 2, pp. 115–148, 1995.
[22] T. Sato, K. Watanabe, and H. Igarashi, "Multimaterial Topology Optimization of Electric Machines Based on Normalized Gaussian Network," *IEEE Transactions on Magnetics*, vol. 51, no. 3, art. no. 7202604, 2015.
[23] S. Kobayashi, "The Frontiers of Real-coded Genetic Algorithms," *Transactions of the Japanese Society for Artificial Intelligence*, vol. 24, no. 1, pp. 147–162, 2009.