Cogging Force Identification Based on Self-Adaptive Hybrid Self-Learning TLBO Trained RBF Neural Network for Linear Motors

1st Xuewei Fu
State Key Laboratory of ASIC & System, School of Microelectronics
Fudan University
Shanghai, China
catfxw@163.com

2nd# Chenyang Ding
Academy for Engineering & Technology
Fudan University
Shanghai, China
dingcy@fudan.edu.cn

3rd Pericle Zanchetta
Department of Electrical and Electronic Engineering&Department of Electrical
University of Nottingham&University of Pavia
Nottingham, UK&Pavia, Italy
pericle.zanchetta@nottingham.ac.uk

4th Xiaofeng Yang
State Key Laboratory of ASIC & System, School of Microelectronics
Fudan University
Shanghai, China
xf_yang@fudan.edu.cn

5th Mi Tang
Power Electronics, Machine and Control Group
University of Nottingham
Nottingham, UK
mi.tang2@nottingham.ac.uk

6th Yang Liu
Center of Ultra-Precision Optoelectronic Instrument Engineering
Harbin Institute of Technology
Harbin, China
hitlg@hit.edu.cn

***Abstract*—The cogging force arising due to the strong attraction forces between the iron core and the permanent magnets, is a common inherent property of the linear motors, which significantly affects the control performance. Therefore, significant research efforts have been devoted to the compensation of the cogging force. In this paper, an identification approach based on the radial basis function neural network (RBFNN) is proposed to obtain an accurate model of the cogging force. A self-adaptive hybrid self-learning teaching-learning-based optimization (SHSLTLBO) method is utilized to train the neural network. Finally, the experimental results confirm the effectiveness and the superiority of the proposed cogging force identification method.**

***Keywords*—cogging force, linear motor, identification, RBFNN, self-adaptive hybrid self-learning TLBO**

I. INTRODUCTION

The ultra-precision motion platforms driven by the linear motors have been widely employed in many nanoscale manufacturing machines such as the lithography wafer scanners [1]. The increasing requirements like higher tracking accuracy, larger velocity and acceleration for the control performance of the motion platforms motivate the necessity of improving the linear motors performance. However, the cogging force arising due to the strong attraction forces between the iron core and the permanent magnets, is a common inherent property of the linear motors, which significantly affects the control performance [2]. The cogging force exists even in the absence of any winding current and it presents a periodic relationship with respect to the position of the translator relative to the magnets. Therefore, significant research efforts have been devoted to the compensation of the cogging force.

The compensation strategies for the cogging force can be approximately summarized into model-free methods and model-based methods. The model-free compensation methods [3-5] do not require accurate cogging force model but need to design complicated self-tuning control approaches which could cause the loss of system stability and could hardly satisfy the highly real-time requirement in practice. In contrast, the model-based compensation approaches [6, 7] require accurate and continuous cogging force model so that the compensation value relative to the position can be obtained in real time. This leads to less challenge of designing the feedback and feedforward controllers, which facilitates the practice applications in the industrial fields. Meanwhile, the control performance can be improved via increasing the identification accuracy of the cogging force.

Therefore, more interests are attracted for developing the effective cogging force identification methods in the linear motors. In [6], the cogging force model was expressed as the sum of a set of sines functions with different frequencies and its parameters were obtained by recursive least square (RLS) algorithm. Although the identification method was quite easy to operate, the fitting result of cogging force in [6] was not good enough to be directly compensated into the system so that an internal model principle based controller was needed to produce better performance. To tackle above problem, a two-stage trained radial basis function neural network (RBFNN) model was proposed in [7] and its superiority over the RLS based identification method was verified by the experiments on a dual-wafer stage. However, the optimization process of the RBF neural network is nonlinear. To overcome the nonlinear optimization problem, in [7] a candidate pool consisting of lots of basis functions was prepared in advance and this transformed the optimization process of the RBF neural network into a linear one through selecting the basis functions in the candidate pool. As a result, this method could not guarantee that the obtained RBFNN model is optimal.

Motivated by the above analysis, a radial basis function neural network based identification approach is proposed to obtain an accurate and continuous model of the cogging force and a self-adaptive hybrid self-learning teaching-learning-based optimization (SHSLTLBO) method [8] is utilized to train the neural network. The contribution of this paper is threefold. First, selecting the RBFNN based model enables the possibility to compensate for the cogging force through the feedforward compensation method which directly introduces the compensation value relative with the position into the system such as the mapping compensation method. Second,

This work was supported in part by the research project of Science and Technology Commission of Shanghai Municipality 18DZ1112600, and in part by Academy for Engineering & Technology, Fudan University with grant number GYY_YC_2020-4.

the SHSLTLBO method is utilized to train the RBFNN. Obviously, the parameters identification process of the RBFNN model is nonlinear which brings more challenges to the optimization. The meta-heuristic optimization techniques such as the SHSLTLBO method belong to the multi-dimension zero-order nonlinear optimization operator, which provides an effective optimization tool for the complicated practical off-line nonlinear problems [9]. Furthermore, based on the hybrid self-adaptive mutation frame, the SHSLTLBO makes better tradeoff between the exploration and the exploitation capacity of teaching-learning-based optimization (TLBO) variants [8]. Finally, an experiment on the linear motor demonstrates the effectiveness and the superiority of the proposed cogging force identification method.

The rest of the paper is organized as follows. The problem statement is formulated in Section II. The RBFNN based cogging force model identification method that is based on the SHSLTLBO is presented in Section III. The experimental results are presented with discussions in Section IV. The conclusions are drawn in Section V.

II. Problem Statement

A. Analysis of Cogging Force

The cogging force in the linear motor consists of the detent force and the force caused by the edge-effect. The detent force is generated by interaction of the iron rotor and the magnetic field of the motor stator, which pulls the rotor to a balance point of the magnetic force. While the force caused by the edge-effect is due to the length limitation of the motor slotting and the rotor. According to the electromagnetic theory [6], the detent force F_d and the edge-effect of the primary winding can be expressed as

$$F_d = \frac{p^2 L_k \tau B_r^2}{2\pi\mu_0\left(1/L_m + 1/L_g\right)} \sum_{k=a,b,c} \frac{d\left[\int_{-(1-K_0)\tau}^{(1-K_0)\tau} \left|A_k(x)\right| dx\right]}{dx} \quad (1)$$

$$\begin{aligned} F_+ &= F_0 + \sum_{n=1}^{\infty} F_{\sin} \sin\frac{2n\pi}{\tau} x + \sum_{n=1}^{\infty} F_{\cos} \cos\frac{2n\pi}{\tau} x \\ F_- &= F_0 + \sum_{n=1}^{\infty} F_{\sin} \sin\frac{2n\pi}{\tau}(x+\Delta) + \sum_{n=1}^{\infty} F_{\cos} \cos\frac{2n\pi}{\tau}(x+\Delta) \end{aligned} \quad (2)$$

where $\Delta = L_k + i\tau$, L_k denotes the iron length, τ denotes the polar distance, L_g denotes the length of the air-gap, L_m denotes the thickness of the magnetic steel, B_r denotes the radial magnetic flux density, p denotes the number of the pole-pairs, and x denotes the axial distance. From (1) and (2), it can be seen that for a designed and finished linear motor the cogging force relies on the relative position of the primary winding and the secondary winding. However, the accurate parameters in the above expressions are unavailable due to the manufacturing or assembling errors. Therefore, it is challenging to obtain the accurate model of the cogging force by using the theoretical results in (1) and (2). From the above analysis, it is noted that the cogging force model can be seen as a function of the displacement of the linear motor. As a result, the data-driven approach can be applied to identify the cogging force model through using the measured data of the cogging force and the displacement.

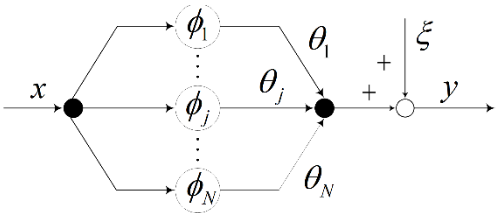

Fig. 1. The structure of RBFNN.

B. Radial Basis Function Neural Network

According to the aforementioned analysis, an RBFNN based model is proposed to depict the cogging force. The RBFNN is a three-layered feedforward neural network and it is a specific form of linear-in-the-parameters model which can be formulated by a linear combination of nonlinear basis functions, whose structure is shown in Fig. 1. x denotes the model input, which is the position of the linear motor. y denotes the model output, which is the cogging force. ξ denotes the measurement noise. ϕ_i is the *i-th* radial basis function and θ_i is the linear coefficient of ϕ_i.

The mathematical output is formulated as

$$y(t) = \sum_{i=1}^{N} \phi_i\left(x(t), v_i\right) \cdot \theta_i + \xi(t) \quad (3)$$

where v_i is the nonlinear parameter set of ϕ_i, and $t = 1, 2, \ldots, M$, M is the number of the trained data. In this paper, the radial basis function ϕ_i of the input $x(t)$ is selected as the Gaussian function defined below

$$\phi_i\left(x(t), v_i\right) = \exp\left(-\frac{\left\|x(t) - \mu_i\right\|^2}{2\sigma_i^2}\right), i = 1, 2, \ldots, N \quad (4)$$

where σ_i is the width and μ_i is the center of the *i-th* hidden node.

Since the optimization of θ_i, σ_i, and μ_i is nonlinear, the identification of these parameters becomes a key problem. In light of the aforementioned analysis, the self-adaptive hybrid self-learning TLBO (SHSLTLBO) method [8] is employed to train the RBFNN model to obtain these linear parameters.

III. RBFNN Model Identification Based on SHSLTLBO

The self-adaptive hybrid self-learning TLBO method consists of a teaching phase, a learning phase and a self-learning phase. The hybrid self-adaptive mutation frame is applied in the teacher phase and the learner phase, which makes better tradeoff between the global exploration to explore the unknown regions and the convergence ability to rapidly exploit the high-precision solutions in the known regions. Meanwhile, the self-learning phase is involved into the self-adaptive frame. In the self-learning phase, two mutation strategies embedded into hybrid self-adaptive mutation frame are used to reduce the possibility of early-maturation in the initialization according to the individual neighborhood information, which improves the convergence performance compared with the original TLBO.

A. *Teacher Phase*

In the teacher phase, the teacher is generated and the individuals are updated through the updating law. As the best performer of the whole population, the teacher shares his/her knowledge with the whole class to improve the performance. The students absorb the experience randomly and mimic different learning capabilities of the students. The updating law of the teacher phase for the *i-th* variable to be optimized of the *j-th* student St_{ij} is expressed as

$$St_{ij}^{new} = \begin{cases} St_{ij}^{old} + rand_1 \cdot \left[T_i - T_F \,| \cdot Mean_i \right], & randp1 < p_1 \\ St_{ij}^{old} + N\left(0,1\right) \cdot \left| T_i - St_{ij}^{old} \right|, & otherwise \end{cases} \quad (5)$$

where St_{ij}^{old} and St_{ij}^{new} denote St_{ij} before and after the teaching process respectively, $rand_1$ and $randp1$ are *D*-dimensions vector of random numbers distributed uniformly in (0,1), *D* is the number of the problem dimensions, p_1 is a mutation control const, T_i is the teacher, $Mean_i$ is the mean value at the *i-th* iteration, and T_F is a teaching factor. T_F is presented as

$$T_F = round\left(1 + rand_2\left(0,1\right)\right) \quad (6)$$

where $rand_2$ is also a random number uniformly distributed among (0,1).

B. *Learner Phase*

After the teacher phase, the optimization algorithm leads the population into a learner phase. In the learner phase, the students will learn knowledge from each other. The diversity of the population is improved by exchanging knowledge among students. The cooperation of the students boosts the exploring capacity of the optimization algorithm. The hybrid mutation law in learner phase is presented as

$$St_{ij}^{new} = \begin{cases} \left.\begin{cases} St_{ij}^{old} + rand_3\left(St_{ik} - St_{ij}^{old}\right) \; if \; f\left(St_{ik}\right) < f\left(St_{ij}^{old}\right) \\ St_{ij}^{old} + rand_3\left(St_{ij}^{old} - St_{ik}\right) \; if \; f\left(St_{ij}^{old}\right) < f\left(St_{ik}\right) \end{cases}\right., randp1 < p_1 \\ St_{ij}^{old} + N\left(0,1\right) \cdot \left| St_{ij}^{old} - St_{ik} \right| \qquad otherwise \end{cases} \quad (7)$$

where $rand_3$ is a random number uniformly distributed among (0,1).

From (5) and (7), it can be seen that the parameter p reflects the usage frequency of the two mutations during the optimization, which affects both the convergence speed and the exploration ability. However, it is difficult to select the value of p without enough prior knowledge. Therefore, the updating law for p is proposed in [8], which is formulated as

$$p = \frac{vs_1 \cdot \left(ns_2 + nf_2\right)}{vs_2 \cdot \left(ns_1 + nf_1\right) + vs_1 \cdot \left(ns_2 + nf_2\right)} \quad (8)$$

where ns_1 and ns_2 denote the number of the individuals that are successfully updated by the first equation in (7) and the second equation in (7) respectively, while nf_1 and nf_2 denote the number of those don't. vs_1 and vs_2 denote the descending ratio of the cost-function caused by the first equation in (7) and the second equation in (7) respectively, which are introduced into (8) to enhance the exploration ability since involving the success ratio of the mutation could not directly reflect the convergence performance of the algorithm. Taking the local convergence as an example, although the individuals are continually updated, the improvement of the cost-function could be limited due to the limited exploration ability. Therefore, the descending ratio of the cost-function caused by the successful mutation is involved to conquer above issue and provide more usage possibility to the preponderant mutation in the next evolution iteration, which is expressed as

$$\begin{cases} vs_1 = \sum \Delta f\left(St_{ij}^{new}\right), if \; f\left(St_{ij}^{new}\right) < f\left(St_{ij}^{old}\right) and \; rand < p \\ vs_2 = \sum \Delta f\left(St_{ij}^{new}\right), if \; f\left(St_{ij}^{new}\right) < f\left(St_{ij}^{old}\right) and \; rand \geq p \end{cases} \quad (9)$$

$$\Delta f\left(St_{ij}^{new}\right) = \frac{f\left(St_{ij}^{old}\right) - f\left(St_{ij}^{new}\right)}{f\left(St_{ij}^{old}\right)} \quad (10)$$

C. *Self-learning Phase*

To further improve the exploitation ability and the convergence performance, a self-learning phase is introduced to the optimization algorithm. In the self-learning phase, the population information is replaced by that of the individual to be used in the updating law. Through this phase, the own information of the individual can be further used to explore the space around themselves. The updating law of the SHSLTLBO algorithm in the self-learning phase is presented as

$$St_{ij}^{new} = \begin{cases} St_{ij}^{old} + R \cdot rand_5 \cdot randDir, & randp2 < p_2 \\ St_{ij}^{old} \cdot \left[1 + \left(rand_4 - 0.5\right) \cdot w\right], & otherwise \end{cases} \quad (11)$$

where $rand_4$ is a random value uniformly distributed in (0,1), $rand_5$ is a vector of random numbers uniformly distributed in (0,1), $randDir = \left[Dir_1, Dir_2, \ldots, Dir_D\right]$ and it denotes a random direction in the solution space, Dir_i is random selected from {-1,0,1}, R is a radius vector, w is a ratio factor, $randp2$ uniformly lies in (0,1), and p_2 is a coefficient adjusting the mutation possibility which is independent with p_1.

When the SHSLTLBO algorithm is used to optimize a problem that is lacking in prior knowledge, the value of pn can be set as an initial value closed to 1 which will play more exploration ability in the initial stage. With the optimization process running, the value of pn will be adjusted by the hybrid self-adaptive mutation frame to balance the exploration and the exploitation capacity. The detailed information of both the pseudocode and the flowchart for SHSLTLBO can be referred to [8].

IV. Experimental Results

To illustrate the proposed cogging force identification approach and evaluate its validity, the experimental study on a linear motor of the wafer stage is implemented in this section.

A. *Experimental Setup*

The RBFNN model input data x and the RBFNN model output data y are obtained through an experiment on a linear motor of the wafer stage. The experimental setup is shown in Fig. 2. The permanent magnet of the linear motor is installed in the stator and the winding groups are assembled in the rotor. The motor drive can make the bandwidth of the current-loop

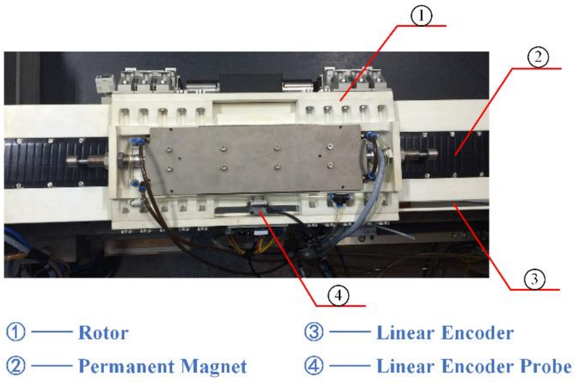

Fig. 2. Block diagram of the experimental setup.

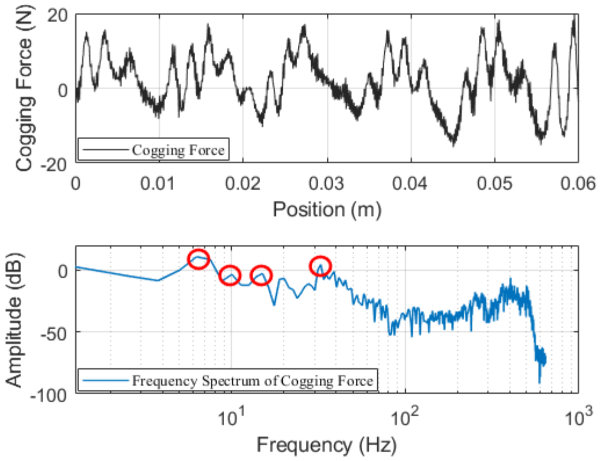

Fig. 3. The measured cogging force varying with the position and spectrum analysis of the coggging force.

TABLE I. Statistic Information of Trained Data

Data	Min	Max	Mean	Variance
Input/m	0	0.0600	0.0300	2.9971×10^{-4}
Output/N	-15.7800	19.7580	1.9487	50.0384

achieve 2000Hz and its peak current is 60A. The position sensor is a Heidenhain linear incremental encoder with the effective resolution of 0.05μm and the maximum velocity of 0.3m/s. The wafer stage is mounted on an air bearing with 400kPa air pressure, which leads to nearly no friction force in the system. The sampling period is set as 200μs.

To measure the real value of the cogging force relative to the position, the reference trajectory is set as a ramp signal with the velocity of 0.075m/s. The data of x is measured by the linear encoder and the system control signal is regarded as the data of y. The measured cogging force varying with position and its spectrum analysis are shown in Fig. 3. In addition, the statistic information of the measured data that is to be trained is shown in TABLE I.

To verify the effectiveness of the proposed method, different cogging force models and different optimization methods are tested via training above measured data.

B. Results of Using Different Cogging Force Model

To illustrate the superiority of the proposed RBFNN based model compared with other models, the nonlinear least square method based approach mentioned in [6] and the BP neural network based method proposed in [10] are tested. According to [6] and Fig. 3, the cogging force is fitted as the expression in (12) and the salient frequencies with peak amplitude are selected as the feature-points. Here, only low-order harmonics are used in the signal model structure with consideration for the existing of the high frequency noise, which leads to i=1,2,3,5. Therefore, the estimated result of the least square based method is shown in Fig. 4.

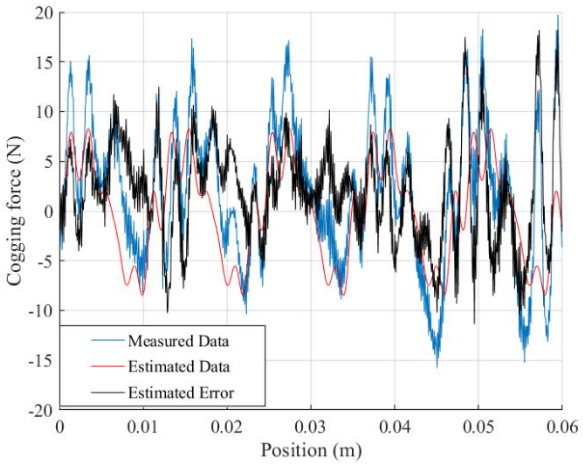

Fig. 4. The estimated result of using the least square method.

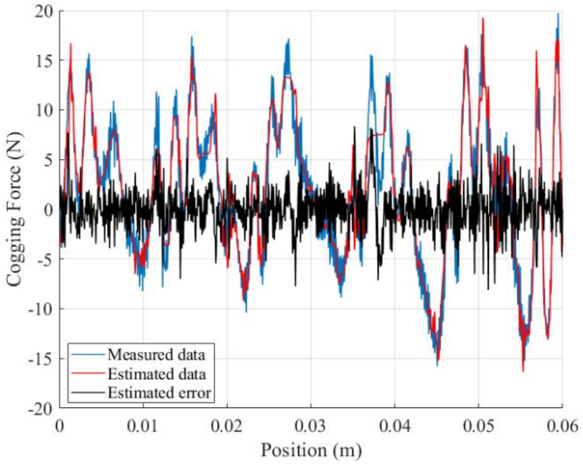

Fig. 5. The estimated result of using the BP neural network model.

$$F_c = \sum C_i \sin\left(i\frac{2\pi x}{\tau}+\theta_i\right) \qquad (12)$$

Obviously, the estimated result of using the least square method is not desirable, which is mainly caused by the too simple model that cannot fit the cogging force due to the inherently complicated frequency component of the measured cogging force. To provide more comparison, the BP neural network model is used to fit the cogging force. According to [10], three hidden layers are used in the BP neural network and there are 60 nodes in each hidden layer. The gradient descent backpropagation with adaptive learning rate is selected to train the network. The iteration period and learning rate are set as 20000 and 0.1, respectively. Since the gradient descent backpropagation requires initial values of the BP neural network, the genetic algorithm (GA) is used to optimize these initial values. For GA, the crossover probability is set as

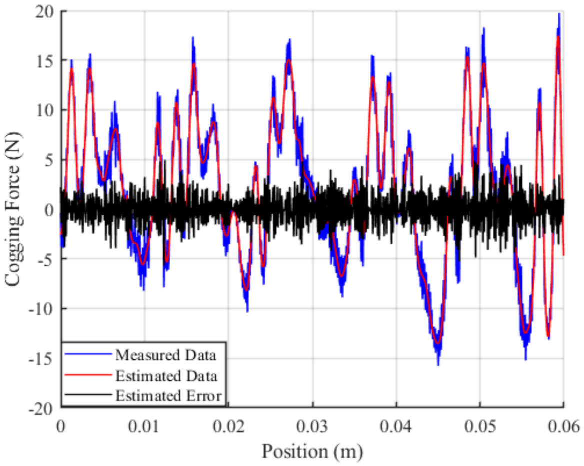

Fig. 6. The estimated result of using the proposed method.

0.2 and the mutation probability is set as 0.1. The estimated result is shown in Fig. 5.

Seen from Fig. 5, the estimated result of using the BP neural network model is better than the result obtained through the least square method. Using the BP neural work to estimate the cogging force model, the maximum estimated error is 8.88056N and the root-mean-square error (RMSE) is 1.9672N. Although the estimated result curve of BP neural network in Fig. 5 seems to good, the statistical analysis of the estimated error shows the unavailability of the estimated result.

From the above results, the necessity of exploiting the proposed cogging force model identification approach is confirmed. The hidden node number of RBFNN is set as 300. The single learning period for SHTLBO is set as 100 and maximal evolutionary generations is set as 20. The population size is set to 100. The ratio factor w is linear varying from 2 to 4. The initial probability parameters p_1 is set as 0.5 and p_2 is set as 0.9. The other initial setups can be referred to [8] in detail. The estimated result of the proposed method is shown in Fig.6. To more clearly observe the difference among the above three methods, the comparative result of the estimated errors is shown in Fig. 7.

From Fig. 7, it can be obviously seen that the proposed method performs better than other two methods in fitting the cogging force model. Using the proposed method, the maximum estimated error of the cogging force identification model is 5.6814N and the RMSE of the estimated error is 1.6679N, which is outperform the BP neural network based model identification method. Compared with the BP neural network based method, using the proposed approach improves the maximum estimated error and the RMSE by 35.48% and 15.21% respectively. Consequently, the above results fully confirm the effectiveness of the RBFNN model in identifying the cogging force model.

C. *Results of Using Different Optimization Algorithms*

To illustrate the superiority of the SHSLTLBO algorithm, four optimization methods that are widely applied in the engineering practice are served as the comparison. The comparative group consists of several classical optimization algorithms, including the differential evolution (DE) [11], the particle swarm optimization (PSO) [12], the heterogeneous comprehensive learning PSO (HCLPSO) [13], and TLBO [8].

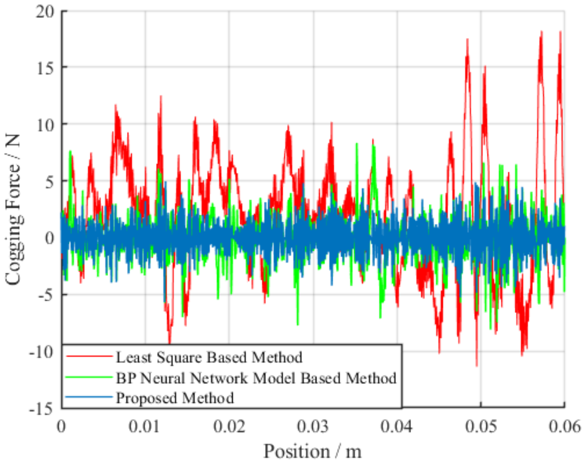

Fig. 7. The comparative result of the estimated errors among three different cogging force models.

To provide a fair comparison, the total number of the generations and the population size of these four optimization methods are set the same as SHSLTLBO. Meanwhile, the detailed parameter settings of these algorithms are set as follows. For DE, F=0.7, CR=0.9 are set. For PSO, the parameters c_1 is set as 1 and c_2 is set as 3. For HCLPSO, the subpopulation sizes are set as $g_1 = g_2 = 50$. The acceleration coefficients c is linear varying from 3 to 1.5, c_1 is linear varying from 2.5 to 0.5, and c_2 is linear varying from 0.5 to 2.5. The inertia weight is linear varying from 3 to 1.5. For TLBO, there is no parameter setting. The RMSE of predicted results is set as the fitness function of each optimization method.

For the iteration number, there are some explanations need to be made for the clarity. For SHSLTLBO the population is updated three times during each iterations, while for other optimization algorithms it could be one or two, for example for DE the updating is with only one time and for TLBO the updating will be run two times. Therefore, to provide a fair comparison, the total updating times of the population is set as 6000, which means for SHSLTLBO the iteration times will be 2000 and for DE the iteration times will be 6000.

Under the above settings, the comparative result of using different optimization algorithms is shown in Fig. 8, and the detailed estimated error results are shown in TABLE II. Furthermore, the comparative result of the estimated errors among HCLPSO, TLBO and SHSLTLBO is shown in Fig. 9.

From these comparative results, several conclusions and analysis can be drawn. First of all, Fig. 8 shows that both TLBO and SHSLTLBO can rapidly converge to optimum but HCLPSO, PSO and DE fall into the local optimum soon after the optimization processes start, which demonstrates the outstanding exploration ability of SHSLTLBO. Second, from Fig. 9 and TABLE II, it can be concluded that SHSLTLBO outperform TLBO in the identification accuracy. Furthermore, Fig. 8 confirms that SHSLTLBO is with more rapid convergence speed than TLBO. Consequently, the above results full illustrate that among different optimization methods, the SHSLTLBO is with superiority of both the identification accuracy and the convergence speed in identifying the cogging force RBFNN model.

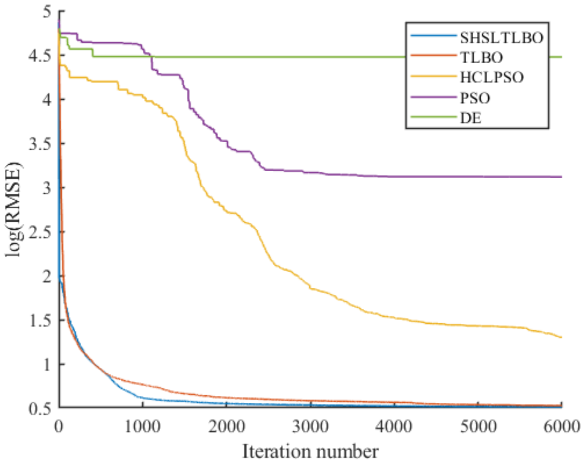

Fig. 8. The comparative result of the estimated errors RMSE with five different optimization methods.

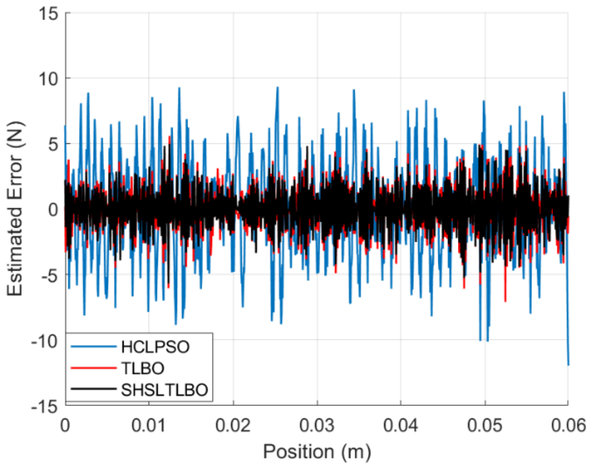

Fig. 9. The comparative result of the estimated errors among HCLPSO, TLBO and SHSLTBO.

TABLE II. ESTIMATED RESULTS WITH FIVE OPTIMIZATION METHODS

Method	SHSL TLBO	TLBO	HCL PSO	PSO	DE
RMSE /N	1.6679	1.6911	3.6714	22.7614	87.6421
Max Error/N	5.6814	6.4534	11.9535	63.4893	249.0551

From the above two subsections, the cogging force identification results respectively with different models and different optimization algorithms fully prove the effectiveness and the superiority of the proposed cogging force identification strategy, which provides a new reference for the research on the cogging force compensation in practice.

V. CONCLUSIONS

This paper addresses the practical problems of the accurate compensation for the cogging force in the linear motors usually driving the precision motion platforms. First, an RBF neural network based cogging force model has been established, which enables the possibility of achieving a continuous accurate model to be directly used to compensate for the cogging force. Subsequently, the SHSLTLBO algorithm is utilized to identify the nonlinear parameters of the RBFNN model. The comparative identification results respectively with different models and different optimization algorithms fully demonstrate the benefits of the proposed identification approach, listed as 1) its ability to achieve more accurate cogging force model; 2) its advantage of improving the convergence speed; 3) its superiority to increasing the exploration and exploitation ability. Finally, the future work can be predicted towards introducing the obtained RBFNN based cogging force model into the control of the motion platform driven by the linear motors.

REFERENCES

[1] T. Oomen, “Learning for advanced motion control,” 16th IEEE International Workshop on Advanced Motion Control, Agder, Norway, 2020.

[2] K. K. Tan, S. N. Huang, and T. H. Lee, “Robust adaptive numerical compensation for friction and force ripple in permanent-magnet linear motors,” IEEE Transactions on Magnetics, vol. 38, no. 1, pp. 221-228, January 2002.

[3] B. Yao, C. Hu, L. Lu, and Q. Wang, “Adaptive robust precision motion control of a high-speed industrial gantry with cogging force compensations,” IEEE Transactions on Control Systems technology, vol. 19, no. 5, pp. 1149-1159, September 2011.

[4] W. Zhang, N. Nan, Y. Yang, W. Zhong, and Y. Chen, “Force ripple compensation in a PMLSM position servo system using periodic adaptive learning control,” ISA Transactions, vol. 95, pp. 266-277, December 2019.

[5] H. Stearns, S. Mishra, and M. Tomizuka, “Iterative tuning of feedforward controller with force ripple compensation for wafer stage,” 10th IEEE International Workshop on Advanced Motion Control, Trento, Italy, March 2008.

[6] Y. Liu, Z. Chen, and T. Yang, “Cogging force rejection method of linear motor based on internal model principle,” Ninth International Symposium on Precision Engineering Measurement and Instrumentation, vol. 9446, pp. 94464Y-1-94464Y-6, March 2015.

[7] Y. Liu, X. Xu, Z. Chen, K. Li, and J. Tan, “RBFNN based linear motor cogging force identification for lithography machines,” IFAC-PapersOnline, vol. 48, issue. 28, pp. 650-655, 2015.

[8] Z. Chen, Y. Liu, Z. Yang, X. Fu, J. Tan, and X. Yang, “An enhanced teching-learning-based optimization algorithm with self-adaptive and learning operators and its search bias towards origin,” Swarm and Evolutionary Computation, vol. 60, February 2021.

[9] A. Geraili, P. Sharma, and J. A. Romagnoli, “A modeling framework for design of nonlinear renewable energy systems through integrated simulation modeling and metaheuristic optimization: Applications to biorefineries,” Computers and Chemical Engineering, vol. 61, pp. 102-117, February 2014.

[10] L. He and Y. Liu, “Cogging force identification for linear motor based on neural network optimized by Genetic algorithm,” Automation and Instrumentation, vol. 2, pp. 1-3, February 2016.

[11] S. Das and P. N. Suganthan, “Differential evolution: A survey of the state-of-the art,” IEEE Transactions on Evolutionary Computation, vol. 15, no. 1, pp. 4-31, February 2011.

[12] Y. Zhang, S. Wang, and G. Ji, “A comprehensive survey on particle swarm optimization algorithm and its applications,” Mathematical Problems in Engineering, vol. 2015, no. 2015-10-7, pp. 1-38, 2015.

[13] N. Lynn and P. N. Suganthan, “Heterogeneous comprehensive learning particle swarm optimization with enhanced exploration and exploitation,” Swarm and Evolutionary Computation, vol. 24, no. 2015, pp. 11-24, 2015.