Fitting of Dynamic Characters for Magnetic Bearing Control System by using Deep Neural Network

LUO Yanyan[a],CHEN Jinping[a], ZHANG Li[a], ZHANG Haining[a] ,LIU Jun[a]

[a] AVIC Qingan group co., LTD, 710077, Xi'an, China

[a] AVIC Qingan group co., LTD, Daqing Road 628, Xi'an China, 920510397@qq.com

Abstract—**Magnetic Bearing Control System（MBCS）has two major aporias， non-linear and strong coupling. Most of the traditional methods are severely weakened by these two problems. This paper presents a noval control algorithm to optimize the performance of Magnetic Bearing Control System 。 The proposed method estimates the non-linear dynamic characters and decouples system by using a deep neural network（DNN）.The inputs of DNN are system PID controller parameters and some other dynamic coefficients， outputs are Magnetic Bearing speed，torque and power output characters. We utilize root-mean-square error（RMSE）as the loss function of DNN since the network is a regression model . Experiments showed that the proposed method could affectively fit and decouple dynamic characters for MBCS and performs competitively against Humans.**

key words: Magnetic bearing Control system, deep neural network, regression model .

I. INTRODUCTION

Active magnetic suspension bearing makes the rotor stable suspension by a controllable electromagnetic force, which have a lot of advantages compared with the ordinary mechanical contact bearings ,such as no contact, no lubrication, no wear, high speed and the adjustable dynamic characteristics. The active magnetic suspension bearing has wide application prospect in high speed, high precision, high power field of rotating machinery. But the active magnetic suspension bearing has extremely strong nonlinear and time-varying characteristics due to the nature of typical nonlinear and open-loop unstable system .The structural parameters of the active magnetic suspension bearing can change with magnetic coupling, force coupling, magnetic saturation, magnetic flux leakage, temperature rise, the rotor eccentricity, eddy current, rotational speed and load parameters . Therefore, it is very challenging to implement effective control of active magnetic bearing, especially when the rotor has high order flexible mode or strong gyro effect.[1]

Traditional controller design depends on the accurate mathematical model of the object. The design of the controller of the nonlinear system relies on the linearizaed model, which use Taylor formula to linearize the nonlinear model at the equilibrium point . As a result of the traditional control algorithm designed on the linearization model of system, so the system is stable only in small range near the equilibrium point . If the system has a larger external disturbance, it will lead to the suspended objects larger deviation from equilibrium. The nonlinear characteristics of the system will lead to instability. And many of the real system exist some uncertain factors , for example parameter variation, un-modeled dynamics characteristics, sensor noise, unpredictable uncertain external disturbance. Some real system also cannot obtain the mathematical modeling. The literature [2] design a decentralized PID feedback controller of magnetic bearing system. It neglects the motion coupling , modal changes and inconsistencies between sensors and actuators .The decentralized control system make the speed of the system unstable and produces high frequency noise. It is difficult to achieve ideal control effect. Therefore, the traditional linear control theory is difficult to solve this special magnetic bearing control problem of nonlinear system, magnetic bearing system need finer control method to solve complex nonlinear strong coupling of magnetic bearing system.

II. RELATED WORKS

A. *Advanced control algorithm on MBCS*

Researchers have been focusing on on fuzzy control, robust control, synovium variable structure control, two order optimal control and adaptive control method on MBCS in simulation. Meanwhile, these advanced control method were keep showing their superiority on many other traditional control methods.

Chen, Lewis et all [3] firstly proposed a Fuzzy-PID algorithm in 1992. Compared with other methods, Fuzzy-PID could fit the non-linear characters of MBCS more robustly. Meanwhile, Fuzzy-PID does not need accurately mathematical models of MB. However, the hyper-parameters such as Quantizing and rate factors play important roles in control system performances by affecting the astringency and accuracy, which are two key factors of the system.

Optimum control method estimates system coefficients by optimizing a loss function based on unpredictable noise and observation data. The most popular type of Optimum control on magnetic bearings is Linear Quadratic Regulator (LQR). Zhu et all [4] had published a research on state function of MBCS, they had described the principle of two order method decoupling, therefore they presented that LQR requires very high accuracy on system mathematic model. A large amount of sensors are needed to identify parameters and state functions. Noticing that external interference would reduce the sensitivity of identification and then the performance indicators of the system will decrease. Futhermore, these errors would even affair the stability of MBCSs. Thus, robustness control against interference from external disturbances raised great concerns. Related robust control methods include H∞ and μ structure singular value. H.M.N.K.Balini et all [5] utilized H∞ and optimized the model with Youla parameter,μ method could introduce uncertain module to system and release the conservatism of the control system due to uncertainties. *Lv, Jiang et all* [6] used the method of μ synthesis to control robustness of the system with interference, taking singular value of the structure as the optimization index. However, the major challenge of the H∞ control method lies in the setting of the index and the selection of the weight function in practical application.

III. DEEP LEARNING ALGORITHM

A. Deep learning in MBCS

Based on the major challenges discussed above, there are few current quality methods for automatic optimizing the PID controller of MBCS. Thus we turn to another region for solution, which is pattern recognition. Deep neural network (DNN) has been utilized to modeling the high-dimension and non-linear functions or predicting the testing sample classes through studying the distribution of training dataset for a long time. DNN is also proved to be an stable and powerful tool for optimizing multi-input-multi-output(MIMO) systems. Comparing with tradition methods, DNN has two following advantages.[7][8]

The first superiority is high accuracy, in some normal using situation, such as image and voice recognition. Deep learning methods had got a significant improvement for system performance in the past five years. For some public datasets, such as ImageNet the image recognition accuracy had arrived 100% by using a very deep convolution neural network(CNN).[9][10][11]

The second superiority is high dimension capacity, in some control system such as MBCS, the input and output dim are always less than 20. However the input dim for a DNN could be much higher (more than 10000). In addition, the dim of DNN is very flexible, it could ranges from 1 to 10k or higher for different utilizing situation and performs stability.

Therefore, the usage of DNN for MBCS controller optimizing is reasonable with the above two advantages. The rest parts of this paper are arranged as follows. The algorithm will be discussed in Sec. 2, MBCS framework will be shown in Sec. 3 and the simulation experiment results will be listed in Sec. 4.

B. Algorithm design

MBCSs often contain dozens of key parameters $P_N \in R^N$, P_N is essential for most of the researchers for designing controllers. However, this process needs a very large amount of formula derivation and matrix calculation. We try to seek a pervasive DNN framework to deal with this question. The basic knowledge DNN should learn are the information of PID controller performance, the coefficients of PID and the parameters of MB. In addition, DNN should know the trend of system performance(getting better or worse). Thus the input of DNN is set in eq. 1.

$$IN = \{P, I, D, P^-, I^-, D^-, T, O, E, T^-, O^-, E^-, S_1, S_2\} \quad (1)$$

In eq. 1, P, I, D are current coefficients in PID controller, P^-, I^-, D^- are coefficients in PID of the previous step,T is stable time,O is overshoot,E is stable error, T^-, O^-, E^- are the former three parameters of the previous step, S_1 is the order of numerator in transfer function, S_2 is the order of denominator in transfer function.[12][13]

For output setting, DNN is needed to predict correct moving direction and step rate of PID parameter optimizing process with the input data. Noticing that moving direction stands for whether the value of P,I,D should become higher or lower, and step length controls how much these values should increase or decrease in this step. Therefore, output is settled in eq. 2.

$$OUT = \{P_C, I_C, D_C, P_S, I_S, D_S\} \quad (2)$$

Where P_C, I_C, D_C are moving direction, P_S, I_S, D_S are step rate. P_C, I_C, D_C have three classes 0,1,-1, 0 stands for P don't change, 1 for increase and 2 for decrease.

The DNN contains a 14 dim input layer, corresponding to the 14 input parameters in eq. 1. Three hidden layers are following the input layer, the respective dims are 256, 512 and 256. Hidden layers are all fully connected layers. The output layer is a 6-dim regression layer, that is because the model we defined is a regression model.[14][15][16]

The loss function we utilized is Root Mean Square Error (RMSE), as listed in eq. 3.

$$J = \sqrt{\frac{\sum_{i=1}^{n}(y_i - t_i)^2}{n}} \quad (3)$$

Where y_i is the output, t_i is training data. J is the loss we used to optimize DNN. The above DNN gives us a tool for optimizing PID coefficients. However, we still

Table 1

PID coefficient optimizing process Pseudo code
Initialize PID randomly in a reasonable range
Give new PID by experiment value
Set literation time N
Get IN_0 and OUT_0 From simulation
for i=1 to N
$\mathrm{OUT}_i = DNN(\mathrm{IN}_i)$
if $T_i, O_i, E_i \gg T_{i-1}, O_{i-1}, E_{i-1}$
$\{P, I, D\}_i = \{P, I, D\}_{i-1} + \Delta$
else
$\{P, I, D\}_i = \{P, I, D\}_i + \{P_c \cdot P_s, I_c \cdot I_s, D_c \cdot D_s\}$
end if
end for
// Δ are disturbances to help literation going out of local optimum

need another framework to use this powerful tool. We had built a literation framework to optimize PID in N steps of using DNN, the process of our framework is listed in tab. 1.[17][18][19]

IV. EXPERIMENT

Before starting, there is a major challenge in the usage of DNN for MBCS need to be discussed. DNNs have a weak point, which decreases the amount of DNN usage in the major of control system. This weak point is data thirsty. For training a DNN, we need a large amount of labeled training data. However, these labeled data are significantly hard to get in many situations. Fortunately, it is quite conveniently to get labeled training data in simulation environments. Through a matlab simulation framework, we had got three databases, training set, validation set and testing set. These three dataset contain 19876, 2000 and 2500 couples of input-output data respectively. Noticing that these training datasets are settled by daily used MBCS coefficients. Network training

Based on the above training data, we trained the DNN

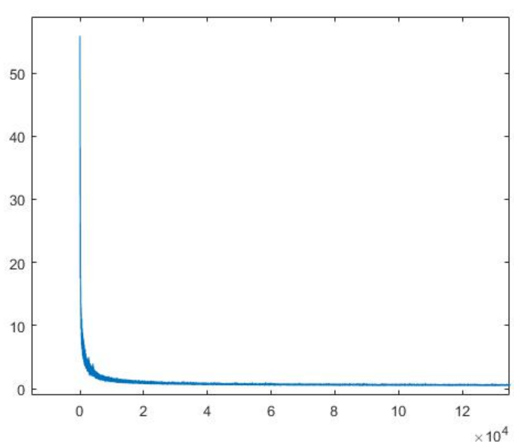

Fig. 3: Training process of loss function, the loss drops fast and SGDM find the global optimize point quickly.

by setting minibatch as 96. Learn rate was as 0.01 and drop 25% every 12 round. Learning method was settled Stochastic Gradient Descent(SGD). The training process is shown in Fig. 3. [20][21][22]

To test the result of optimizing PID parameters, we took the 2500 couples of input-output data as testing set. To evaluate the optimizing performance, we study three major dims of experiment outputs, stable time (ST), overshoot (OST) and stable error (OE), testing results are shown in Tab. 2.

Where the values in row ST express that how much time did our method reduced compared with artificially

Table 2

Experiment results

Sytem Order / Estimate optimized	2	3	4	5
ST	0.57s	0.33s	0.56s	0.82s
OST	99%	91%	81%	63%
OE	95%	96%	98%	92%
Average time cost	0.92s	0.88s	1.02s	1.11s
Average time cost (human)	1~10 minutes	5~25 minutes	20~40 minutes	30~60 minutes

modulated PID parameters in 5 steps. OST row shows the overshoot reduced of human modulated PID in five steps. OE row shows the mean stable reduced of random PID parameters. From the two bottom line of Tab. 2 we could conclude that our method could find an excellent PID parameter set for MBCS much faster and suitable than human.

V. CONCLUSION

This paper proposes a novel method for PID controller parameter modulation using deep neural network. The network is designed for optimizing PID coefficients by learning the system performance index and outputs the direction and value for PID adjustment. Experiment showed that our method is reasonable for MBCS PID controllers and performs much better than skilled engineers. We believe that our work is strong and novel for MBCS. In addition, our method could be utilized in many other practical scenes while the controller is PID and with more data, our method will work better.

REFERENCES

[1] Gerhard Schweitzer.Magnetic bearings:Theory,design,and application to rotating machinery[M].Springer-Verlag Berlin herdelberg,2009.

[2] Cao Guangzhong.Control Algorithms and Their Realization in the Magnetic Levitation System[M].Beijing,Tsinghua University Press,2013.

[3] Chen H.M,Lewis P.Pule-based damping control for magnetic bearing[J].*Proceedings of 3rd International symposium on magnetic bearing,1992:25-34.*

[4] Zhu Huangqiu,Sun Yukun.Application and study of optimal control the optic control system of active magnetic bearings[J].Control the Theory and Applications,2002,19:479-483.

[5] H.M.N.K.Balini,W.Scherer,Jasper Witte.Performance enhancement for AMB systems using unstable H∞ controllers[J].IEEE Transactions on control System Technology,2010,19:1479-1492.

[6] Lv Shenggang,Jiang Shuyun.Study of μ robust control of active magnetic bearing[J].Journal of mechanical engineering,2007,43:133-137.

[7] LeCun Y, Bengio Y, Hinton G. Deep learning[J]. nature, 2015, 521(7553): 436.

[8] Krizhevsky, A., Sutskever, I. & Hinton, G. ImageNet classification with deep convolutional neural networks. In Proc. Advances in Neural Information Processing Systems 25 1090–1098 (2012). This report was a breakthrough that used convolutional nets to almost halve the error rate for object recognition, and precipitated the rapid adoption of deep learning by the computer vision community.

[9] Farabet, C., Couprie, C., Najman, L. & LeCun, Y. Learning hierarchical features for scene labeling. IEEE Trans. Pattern Anal. Mach. Intell. 35, 1915–1929 (2013).

[10] Tompson, J., Jain, A., LeCun, Y. & Bregler, C. Joint training of a convolutional network and a graphical model for human pose estimation. In Proc. Advances in Neural Information Processing Systems 27 1799–1807 (2014).

[11] Szegedy, C. et al. Going deeper with convolutions. Preprint at http://arxiv.org/ abs/1409.4842 (2014).

[12] Jean, S., Cho, K., Memisevic, R. & Bengio, Y. On using very large target vocabulary for neural machine translation. In Proc. ACL-IJCNLP http://arxiv.org/ abs/1412.2007 (2015).

[13] Sutskever, I. Vinyals, O. & Le. Q. V. Sequence to sequence learning with neural networks. In Proc. Advances in Neural Information Processing Systems 27 3104–3112 (2014).

[14] Bottou, L. & Bousquet, O. The tradeoffs of large scale learning. In Proc. Advances in Neural Information Processing Systems 20 161–168 (2007).

[15] Duda, R. O. & Hart, P. E. Pattern Classification and Scene Analysis (Wiley, 1973).

[16] Schölkopf, B. & Smola, A. Learning with Kernels (MIT Press, 2002).

[17] Bengio, Y., Delalleau, O. & Le Roux, N. The curse of highly variable functions for local kernel machines. In Proc. Advances in Neural Information Processing Systems 18 107–114 (2005).

[18] Selfridge, O. G. Pandemonium: a paradigm for learning in mechanisation of thought processes. In Proc. Symposium on Mechanisation of Thought Processes 513–526 (1958).

[19] Rosenblatt, F. The Perceptron — A Perceiving and Recognizing Automaton. Tech. Rep. 85-460-1 (Cornell Aeronautical Laboratory, 1957).

[20] Werbos, P. Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences. PhD thesis, Harvard Univ. (1974).

[21] Parker, D. B. Learning Logic Report TR–47 (MIT Press, 1985).

[22] LeCun, Y. Une procédure d’apprentissage pour Réseau à seuil assymétrique in Cognitiva 85: a la Frontière de l’Intelligence Artificielle, des Sciences de la Connaissance et des Neurosciences [in French] 599–604 (1985).