PID-type Fuzzy Logic Controller for Active Magnetic Bearing System

Amin Noshadi, Juan Shi, WeeSit Lee, Akhtar Kalam, and Ali Zolfagharian
College of Engineering and Science
Victoria University
Melbourne, Australia
Amin.Noshadi@live.vu.edu.au, Juan.Shi@vu.edu.au, Weesit.Lee@vu.edu.au, Akhtar.Kalam@vu.edu.au,
Ali.Zolfagharyan@gmail.com

***Abstract*—This paper describes the design of a PID-type Fuzzy Logic Controller (PID-FLC) and its application on the stabilisation of an Active Magnetic Bearing System (AMB). The proposed PID-FLC is obtained by combining a PD-type Fuzzy Logic Controller (PD-FLC) and a PI-type Fuzzy Logic Controller (PI-FLC). A multi-objective Genetic Algorithm (MOGA) is used to determine the scaling factors of the inputs and outputs of the PID-FLC. The designed controller is then coded in C and implemented in real time on a Digital Signal Processor (DSP) card. The results from the PID-FLC are compared with those of a conventional lead-lag type controller and the system's on-board analogue controller. Designing controllers based on classical methods could become tedious, especially for systems with high order model. In contrast, PID-FLC controller design requires only tuning of some scaling factors in the control loop and hence is much simpler than classical design methods. The experimental results have also verified the enhanced performance and robustness of the system under proposed PID-FLC control in the presence of disturbance.**

Keywords—fuzzy logic control; PID-type fuzzy logic control; Active Magnetic Bearing system; multi-objective genetic algorithm

I. INTRODUCTION

Active Magnetic Bearing Systems (AMBs) can provide contactless suspension of rotors by attractive electromagnetic forces produced by electromagnets. Classical mechanical bearings can be replaced by active magnetic bearings, where the attraction force is provided by electric current. AMBs have various advantages over conventional mechanical and hydrostatic bearings. These advantages are zero frictional wear and efficient operation at extremely high speed. AMBs are ideal for clean environments where no lubrication is required. They can work in harsh environments such as high temperature, heavy load and high humidity. These special characteristics have attracted significant attention and interests amongst the researchers in the past decade. AMBs can be utilised in many industrial applications, where fast and precise operations are desired such as energy storage flywheels, artificial heart, high speed turbines and jet engines [1] and [2]. However, highly nonlinearity and open-loop unstable characteristics of these systems have hindered the widespread commercial and industrial application of AMBs. Since AMB is open-loop unstable, closed loop system identification is always required for model-based controller design purpose. In order to achieve high productivity and performance typically in terms of high speed and short response time in AMBs, it is necessary to have AMBs reliably stabilised. Even though the AMBs have existed for a long time, there are a lot of issues governing the systems that need to be explored. Some of these issues are finite bearing stiffness, rotor unbalance and easily excitable high frequency oscillations. Therefore, AMBs control system performance is demanding in terms of stability and robustness improvements.

In recent years, various types of controllers have been implemented on AMBs to address the challenges in modelling and control of AMBs. Reference [3] reported the design and implementation of a conventional lead compensator (PD controller) to stabilise a MBC500 magnetic bearing system. Different structures of H_∞ control, such as linear H_∞ loop-shaping, signal-based H_∞ control and two-degrees-of-freedom (2DOF) H_∞ control were implemented in [4], [5] and [6] to AMB system. The designed controllers based on H_∞ method were then compared with other control methods such as Linear-Quadratic-Gaussian (LQG) and PID. Generally, an accurate model of the system is crucial for obtaining a good controller for system stabilisation.

However, for systems subject to external disturbances, nonlinearities, uncertainties and signal limits, obtaining a precise model of the system in various operating conditions is very difficult. Thus, model-free controllers have emerged recently in the literature [7]. Although the model of the system is required in the controller design stage, an exact model of the system is not required for controller synthesis. PD-type fuzzy controller (PD-FLC) and PI-type fuzzy controller (PI-FLC) are the most commonly reported methods in the literature. In [8] and [9], a PD-like fuzzy logic controller was designed for stabilisation of open-loop AMBs. The obtained results were compared with conventional lead compensator. The advantage of using the PD-FLC is that unlike conventional controller design methods, the exact model of the system is not required at the controller design stage. PD-FLC improves the transient response of the system. However, the steady-state error of the system remains large when PD-FLC is employed. Although PI-FLC control is known to be more practical than PD-FLC because it results in zero steady state error [10], due to the phase lag introduced by the integral action, its transient performance is poor for high order processes. While the three-term PID-FLC control is able to enhance the system

performance, its need for three inputs to the fuzzy controller ($N_i \times N_j \times N_k$ is the number of rules where N_i, N_j, and N_k is the number of membership functions of the inputs) [11]. An alternative approach to alleviate this problem is to add an integrator to the output of the PD-FLC and sum it to the PD part. However, it then produces a PD-FLC+I instead of a PID-FLC [12].

In this paper, in order to design a PID-type fuzzy logic controller, we combined a PI-FLC and a PD-FLC. This structure simplifies the configuration and makes the controller easier to implement because both rule-bases are two-dimensional. The problem then becomes one of designing a PI and a PD rule base, which reduces the complexity of the design [13]. After designing the fuzzy controller, fine tuning can be made to improve the performance of the controller. Tuning can be made either to the membership functions or to the scaling factors (note that it is common to use normalised inputs and outputs for fuzzy controller and hence scaling factors are required to normalise these inputs and outputs). However, as the rule-base conveys a general control policy, it is preferred to keep the rule-base unchanged and the tuning exercise is focused on the scaling factors. Multi-objective Genetic Algorithm (MOGA) was utilised to optimise the PID-FLC scaling factors in order to achieve the optimum trade-off between the desired objective functions. The final obtained controller was then implemented on the AMBs for system stabilisation analysis. The results from PID-FLC show a considerable performance enhancement in comparison with those of a conventional PD-FLC, a conventional lead-lag compensator and the on-board analogue controller. It should be noted that, although the carefully designed controller based on conventional method lead to a good performance, an extensive knowledge about the system behaviour is required.

II. Description of the Model and System Identification

The system under study is a shaft containing two pairs of magnetic bearings on each end to levitate the shaft on its equilibrium position. This can be represented as a four-input four-output system. Fig. 1 illustrates the top and front views of the experimental setup. The system employs four linear current-amplifier pairs (one pair for each radial bearing axis) and four internal analogue lead compensators to independently control the radial bearing axes. In this paper, we present a design where the on-board analogue controller will be replaced by digital controllers. A model of the MBC500 magnetic bearing system will be identified only for the purpose of dynamic simulation and conventional controller design as the design of FLCs does not require the model of the system. Since the magnetic bearing system is open-loop unstable, a closed-loop system identification procedure is required. A two-step closed-loop system identification procedure is employed to identify a model for the magnetic bearing system. The details of the frequency response experiment and the system identification procedure were described in Reference [14]. By using the experimental data collected, a model of the magnetic bearing was obtained by using the method described in [14]. The experimentally collected frequency response magnitude is also shown in Fig. 2. However, in this paper, only the model of the first channel is used for conventional controller design and AMB system dynamic simulation.

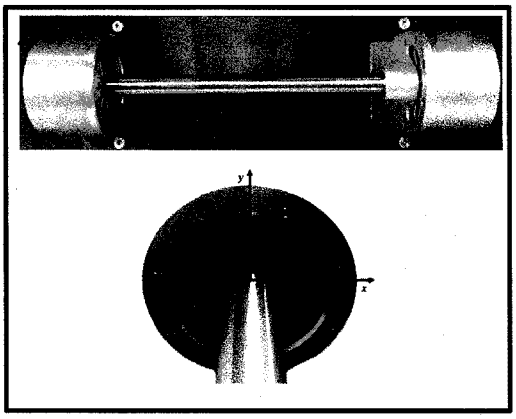

Fig. 1. Active magnetic bearing experiment.

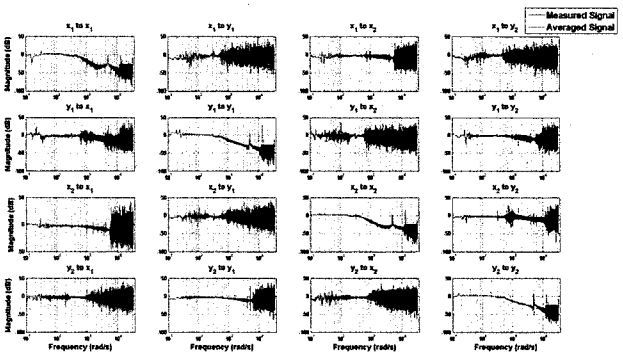

Fig. 2. Frequency response of MIMO system.

Two resonances can be seen in the frequency response of the first channel (Fig. 2). Since the total model order of four is required to model the two resonant modes, it can be deduced from Fig. 3 that models with order less than five will fail to accurately model the actual plant. Furthermore, the higher the order model, the more difficult to design the conventional controller. The ultimate goal here is to find a model of the real system that is as simple as possible and yet capable of capturing all of the essential characteristics of the plant. Therefore, a 6th order model is chosen in order to model both the rigid-body and bending-body modes with a second order and fourth order transfer function, respectively.

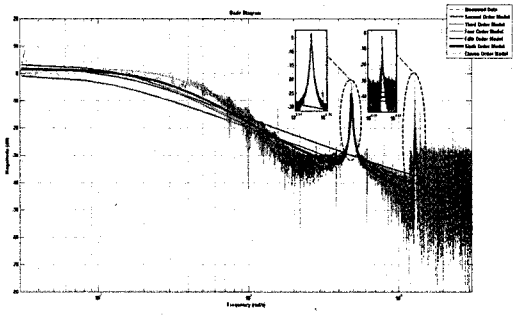

Fig. 3. Frequency responses of the fitted models vs experimental data.

A sixth order model of the first channel is obtained and follows:

$$G_{x1}(s) = \frac{-0.0026695(s+3.595e4)(s-1200)(s^2-1240s+2.127e07)(s^2-1140s+1.631e08)}{(s+389)(s-212.5)(s^2+46.56s+2.344e07)(s^2+49.37s+1.673e08)} \quad (1)$$

In order to reduce the number of rules that is used in regular three-input PID-type fuzzy controller, two Sugeno-type FLC structures shown in Fig. 4 are used in this paper. In fact, the PID action is separated into a PI-FLC part and a PD-FLC part. The outputs from these two fuzzy controllers are then added to produce a PID-FLC. As it can be seen from Fig. 4, four parameters need to be tuned using optimisation methods. Namely, GE is the input error scaling factor, GCE is the input rate of change of error scaling factor, GCU is the PI-FLC output scaling factor, and GU is the PD-FLC output scaling factor respectively.

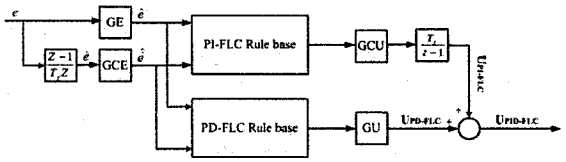

Fig. 4. Structure of the PID-FLC Controller.

The inputs to both the PI-FLC and the PD-FLC parts are scaled error ($\hat{e}(t)$) and the scaled rate of change of error ($\hat{\dot{e}}(t)$). The linguistic variables "error", "rate of change of error," and "control output" will take on the following linguistic values:

"NL" = Negative Large
"NM" = Negative Medium
"NS" = Negative Small
"ZO" = Zero
"PS" = Positive Small
"PM" = Positive Medium
"PL" = Positive Large

The size of inputs and output membership functions is chosen to be seven. The membership functions of the input variables for both the PI-FLC and the PD-FLC to be employed are of the triangular type and they are defined as shown in Figs. 5 and 6. The membership function of the output variable for both the PI-FLC and the PD-FLC are singletons as depicted in Fig. 7.

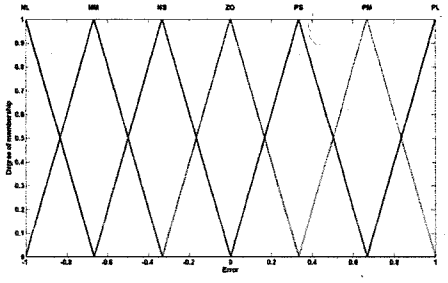

Fig. 5. Membership functions of the input error $\hat{e}(t)$ in the PI-FLC and the PD-FLC.

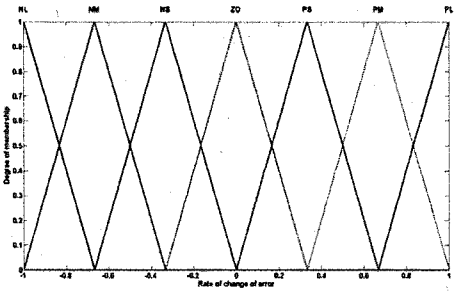

Fig. 6. Membership functions of the input rate of change of error $\hat{\dot{e}}(t)$ in the PI-FLC and the PD-FLC.

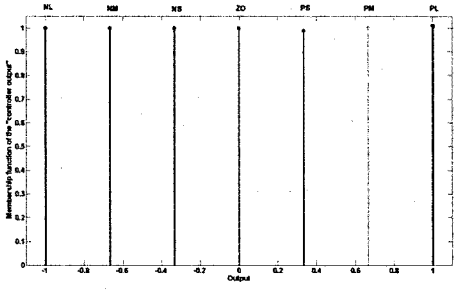

Fig. 7. Membership functions of the control output in the PI-FLC and the PD-FLC.

The controller surface for the PI-FLC and the PD-FLC are depicted in Figs. 8 and 9.

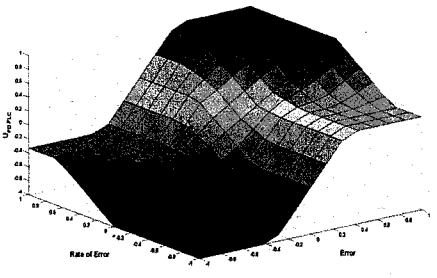

Fig. 8. Control surface of the PD-FLC.

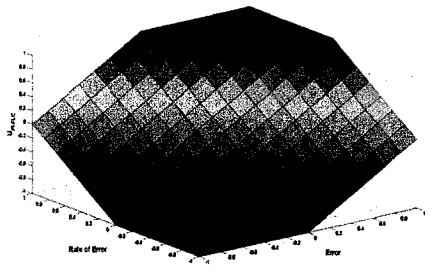

Fig. 9. Control surface of the PI-FLC.

For the two configurations (PD-FLC and PI-FLC) used in Fig. 4, two complete rule matrices of size 7 * 7 are defined separately in Table. I and II.

TABLE I. RULE-BASE FOR THE PI-FLC

"Control Output" $U_{PI\text{-}FLC}$		error (e)						
		NL	NM	NS	ZO	PS	PM	PL
rate of change of error ($\dot{e}$)	PL	ZO	PS	PM	PL	PL	PL	PL
	PM	NS	ZO	PS	PM	PL	PL	PL
	PS	NM	NS	ZO	PS	PM	PL	PL
	ZO	NL	NM	NS	ZO	PS	PM	PL
	NS	NL	NL	NM	NS	ZO	PS	PM
	NM	NL	NL	NL	NM	NS	ZO	PS
	NL	NL	NL	NL	NL	NM	NS	ZO

TABLE II. RULE-BASE FOR THE PD-FLC

"Control Output" $U_{PD\text{-}FLC}$		error (e)						
		NL	NM	NS	ZO	PS	PM	PL
rate of change of error ($\dot{e}$)	PL	NS	NS	NS	PS	PL	PL	PL
	PM	NS	NS	NS	PS	PL	PL	PL
	PS	NM	NS	NS	PS	PM	PL	PL
	ZO	NL	NM	NS	ZO	PS	PM	PL
	NS	NL	NL	NM	NS	PS	PS	PM
	NM	NL	NL	NL	NS	PS	PS	PS
	NL	NL	NL	NL	NS	PS	PS	PS

Since both the PD-FLC and the PI-FLC parts share the same inputs, the number of rules has been reduced to $7 \times 7 + 7 \times 7$ instead of regular three input PID-FLC controller with (7×7×7) rules. Note that the universe of discourse for both the PI-FLC and the PD-FLC is normalised between [-1, 1]. Thus, four scaling factors namely GE, GCE, GCU and GU (see Fig. 4) need to be designed and tuned using MOGA. This will be discussed in the following section.

III. TUNING OF THE SCALING FACTORS USING MOGA

The overall output of the PID-FLC controller $u_{PID-FLC}$ can be obtained as the sum of the PI-FLC output and the PD-FLC output (see Fig. 4).

$$u_{PID-FLC} = u_{PI-FLC} + u_{PD-FLC} \tag{2}$$

The output from PI-FLC controller can be written as:

$$u_{PI-FLC} = GCU \times \int \left(\hat{e}(t) + \hat{\dot{e}}(t)\right) dt \tag{3}$$

$$u_{PI-FLC} = GCU \times \int (GE \times e(t) + GCE \times \dot{e}(t))\, dt \tag{4}$$

$$u_{PD-FLC} = GU \times (\hat{e}(t) + \hat{\dot{e}}(t)) \tag{5}$$

$$u_{PD-FLC} = GU \times (GE \times e(t) + GCE \times \dot{e}(t)) \tag{6}$$

Substituting (4) and (6) into (2) results in:

$$u_{PID-FLC} = (GCU \times GCE + GU \times GE)e(t) + (GCU \times GE) \int e(t)dt + (GU \times GCE)\hat{\dot{e}}(t) \tag{7}$$

Comparing (7) with the output equation of a traditional PID controller results in:

$$K_p = GCU \times GCE + GU \times GE \tag{8}$$

$$K_I = GCU \times GE \tag{9}$$

$$K_D = GU \times GCE \tag{10}$$

Thus, the initial value of the scaling factors in (8) to (10) (GCU, GU, GE, and GCE) can be obtained using available PID-tuning methods such as Ziegler-Nichols method. Furthermore, fine tuning can be made by employing optimisation methods. MOGA has been chosen as an optimization approach due to its remarkable properties compared to its counterparts [15]. The main idea of using MOGA is to obtain the optimum solution when there are multiple objectives need to be optimised at the same time. MOGA using fitness sharing technique is adopted in this study due to its versatile characteristic for dealing with various conflict objectives and their constraints. MOGA based on fitness sharing has been successfully implemented in other control engineering problems [16] and [17]. In MOGA, fitness sharing technique is utilised to give confidence in the search toward the true Pareto optimal set while maintaining diversity in the population. The basic idea of fitness sharing is that all the individuals within the same region (called a niche) share their fitness. In fitness sharing method a niche count is obtained from the Euclidean distance between every pair of solutions first and then the fitness of each solution is ranked from the best individual to the worst. Details of this method can be found in [18].

Given the system $G(s)$ (see Fig. 3) with unity feedback, controller $K(s)$, disturbance d, measurement noise n, and reference input r, the closed-loop model of the system can be shown as:

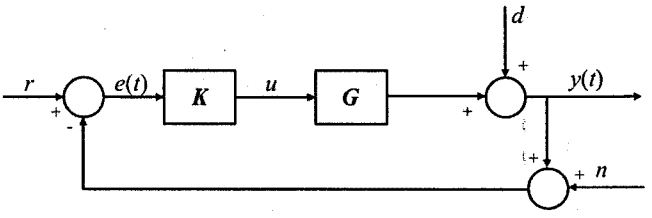

Fig. 10. Closed-loop model of the system.

The integral of absolute magnitude of the error (IAE) and the maximum overshoot of the AMB system were chosen as performance indices.

$$J_1 = \int_0^t |e(t)|\, dt \tag{11}$$

$$J_2 = \max(y(t)) \tag{12}$$

Convergence of the objective functions using MOGA is depicted in Fig. 11. Optimal values of GCU, GU, GE, and GCE were found to be 0.18, 0.0048, 9.5 and 111.55, respectively.

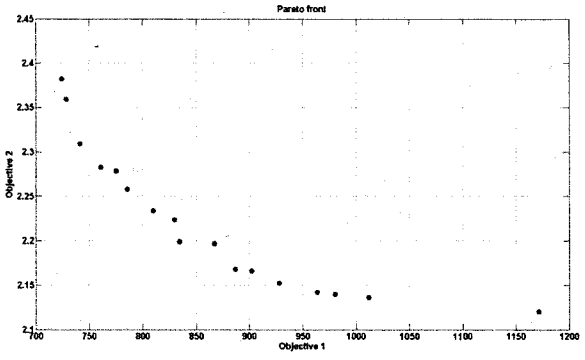

Fig. 11. Pareto front of the two objectives using MOGA.

A flowchart of MOGA is illustrated in Fig. 12.

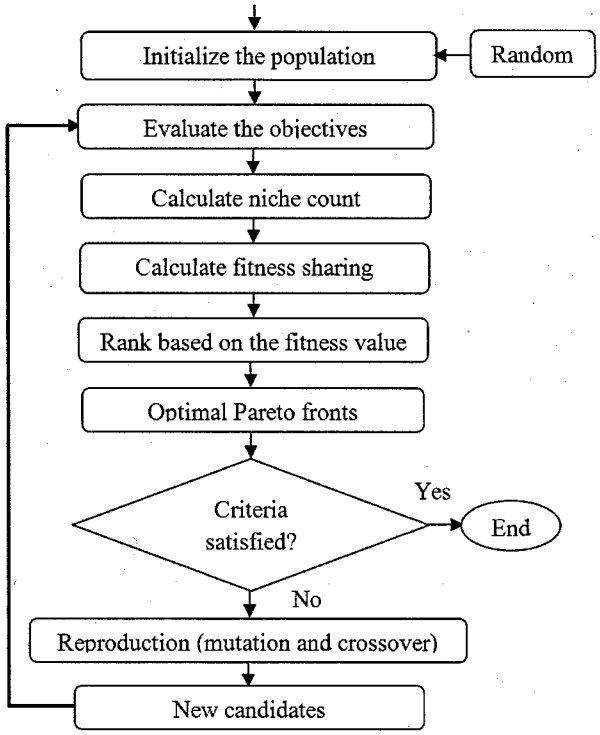

Fig. 12. Flowchart of MOGA.

IV. Implementation Of The Designed Fuzzy Controller

After designing the PID-FLC controller and obtaining the required scaling factors using MOGA, a dSPACE DS1104 processor board, MATLAB, and dSPACE Control Desk are employed to implement the controller in real time. The controller is coded in C with a sampling frequency of 20 KHz. The inputs and output of the controller are connected to the MBC500 system via the ADC and the DAC of the DS1104 processor board. Performance of the designed PID-FLC controller is first compared with a carefully designed conventional lead-lag controller. The results demonstrate that the performance of the system using the PID-FLC is better than the lead-lag type compensator. It should be noted that in the conventional controller, the designed lead-lag compensator should be cascaded to two notch filters in order to attenuate the effect of the structural resonance frequencies (at approximately 775Hz and 2059Hz). The final conventional controller should also be cascaded with a low-pass filter in order to attenuate the high-frequency measurement noise and the destabilising effect of high-frequency model uncertainty. In contrast, the designed PID-FLC can be implemented on the AMBs without employing any notch filters and low-pass filters.

The FLC was tested extensively to ensure that it can operate in a wide range of conditions. A unit step disturbance is introduced to the first channel after approximately 1.5 seconds and the results from PID-FLC, lead-lag compensator and the on-board controller are depicted in Figs. 13 and 14.

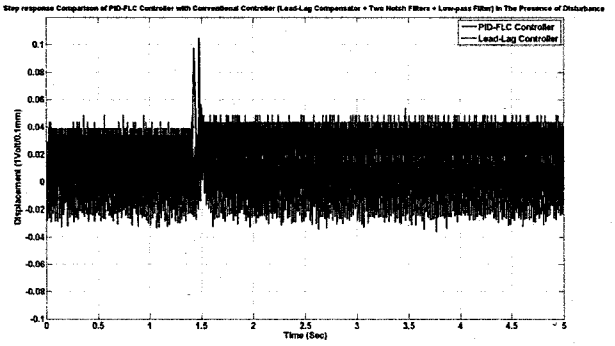

Fig. 13. Step response of the system using the PID-FLC controller and the lead-lag compensator in the presence of disturbance.

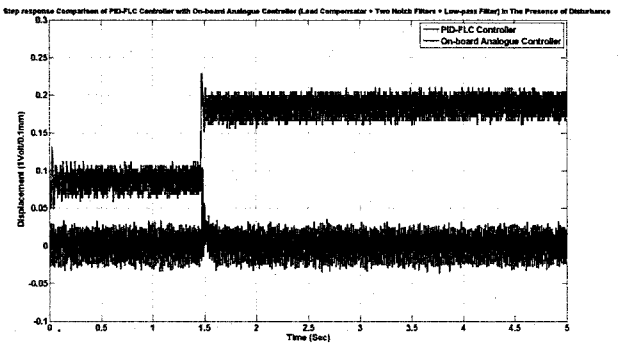

Fig. 14. Step response of the system using the PID-FLC controller and the on-board analogue controller in the presence of disturbance.

One advantage of using the proposed PID-FLC scheme is that by only replacing the scaling factor GCU (see Fig. 4) to zero, a PD-FLC can be achieved without any further adjusments or modifications. The results obtained from the PD-FLC is compared with the on-board analogue controller (on-board analgue controller was a lead-type compensator in serier with two notch filters and a low-pass filter) in Fig. 15. The resutls clearly show that the PD-FLC (without any notch filters and low-pass filters) that was obtained by setting GCU to zero has a very similar behaviour to the on-board lead compensator.

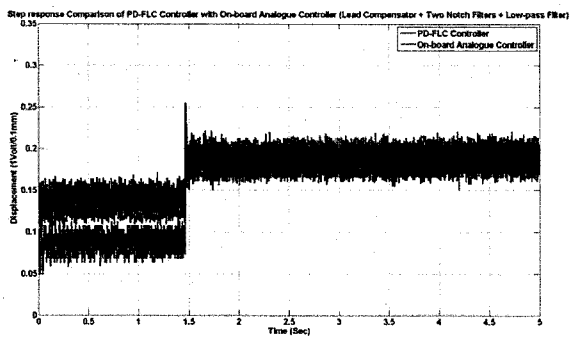

Fig. 15. Step response of the system using the PD-FLC controller and the on-board analogue controller in the presence of disturbance.

V. CONCLUSION

A PID-FLC controller was designed by combining a PI-FLC and a PD-FLC. Multi-objective Genetic Algorithm (MOGA) was used to secure an optimal trade-off between multiple conflicting performance measures such as steady-state error, overshoot, settling time and rise time. The optimal PID-FLC controller was then implemented on the system in real time. The results were compared with those of a conventional lead-lag compensator and the on-board analogue controller in the presence of disturbance. The results show that the system's performance was dramatically improved by using the PID-FLC compared to the on-board analogue controller. It is worth noting that the designed lead-lag type compensator also worked well. However, two notch filters and a low-pass filter had to be added to the designed lead-lag compensator before closed-loop stability becomes a reality. The PID-FLC, on the other hand, was successfully implemented without the need of any notch filters or low-pass filters. This is an advantage over conventional lead-lag controllers where notch filters are generally needed for cancelling the resonant modes and low-pass filter is required to attenuate the high-frequency measurement and the destabilising effect of high-frequency model uncertainty in AMBs.

REFERENCE

[1] J. X. Shen, K. J. Tseng, D. M. Vilathgamuwa, and W. K. Chan, "A novel compact PMSM with magnetic bearing for artificial heart application," *IEEE Transactions on Industry Applications,* vol. 36, pp. 1061-1068, 2000.

[2] K. C. Lee, Y. H. Jeong, D. H. Koo, and H. J. Ahn, "Development of a radial active magnetic bearing for high speed turbo-machinery motors," in *the proceedings of SICE-ICASE International Joint Conference*, 2006, pp. 6090-6095.

[3] J. Shi and J. Revell, "System identification and re-engineering controllers for a magnetic bearing system," in *the proceedings of IEEE Conference on Computers, Communications, Control and Power Engineering*, 2002, pp. 1591-1594 vol.3.

[4] R. P. Jastrzebski, K. M. Hynynen, and A. Smirnov, "H∞ control of active magnetic suspension," *Mechanical Systems and Signal Processing,* vol. 24, pp. 995-1006, May 2010.

[5] R. P. Jastrzebski, A. Smirnov, O. Pyrhönen, and A. K. Piłat, "Discussion on Robust Control Applied to Active Magnetic Bearing Rotor System," in *Challenges and Paradigms in Applied Robust Control*, ed: InTech, 2011.

[6] A. Smirnov, R. P. Jastrzebski, K. Hynynen, and O. Pyrhonen, "Comparison of suboptimal control methods in magnetic levitation system," in *the Proceedings of 15th European Conference on Power Electronics and Applications*, 2013, pp. 1-10.

[7] J. Y. Hung, "Magnetic bearing control using fuzzy logic," *IEEE Transactions on Industry Applications,* vol. 31, pp. 1492-1497, Nov-Dec 1995.

[8] J. Shi and W. S. Lee, "An Experimental Comparison of a Model Based Controller and a Fuzzy Logic Controller for Magnetic Bearing System Stabilization," in *the Proceedings of IEEE International Conference on Control and Automation*, 2009, pp. 379-384.

[9] J. Shi and W. S. Lee, "Design and Implementation of Conventional and Advanced Controllers for Magnetic Bearing System Stabilization," in *Magnetic Bearings, Theory and Applications*, ed: Bostjan Polajzer (Ed.), 2010.

[10] H. X. Li and H. B. Gatland, "Conventional fuzzy control and its enhancement," *IEEE Transactions on Systems Man and Cybernetics Part B-Cybernetics,* vol. 26, pp. 791-797, Oct 1996.

[11] H. C. Chen, "Optimal fuzzy PID controller design of an active magnetic bearing system based on adaptive genetic algorithms," in *the proceedings of International Conference on Machine Learning and Cybernetics*, 2008, pp. 2054-2060.

[12] G. K. I. Mann, B. G. Hu, and R. G. Gosine, "Analysis of direct action fuzzy PID controller structures," *IEEE Transactions on Systems Man and Cybernetics Part B-Cybernetics,* vol. 29, pp. 371-388, Jun 1999.

[13] P. J. Escamilla-Ambrosio and N. Mort, "A novel design and tuning procedure for PID type fuzzy logic controllers," in *the proceedings of First International IEEE Symposium Intelligent Systems*, 2002, pp. 36-41.

[14] A. Noshadi, J. Shi, W. S. Lee, P. Shi, and A. Kalam, "Genetic Algorithm-based System Identification of Active Magnetic Bearing System: A Frequency-domain Approach," in *To be published in the proceedings of International Conference of Control and Automation*, Taiwan, 2014.

[15] K. Deb, *Multi-objective optimization using evolutionary algorithms* vol. 2012: John Wiley & Sons Chichester, 2001.

[16] V. V. Silva, P. J. Fleming, J. Sugimoto, and R. Yokoyama, "Multiobjective optimization using variable complexity modelling for control system design," *Applied Soft Computing,* vol. 8, pp. 392-401, 2008.

[17] A. Zolfagharian, A. Noshadi, M. R. Khosravani, and M. Z. M. Zain, "Unwanted noise and vibration control using finite element analysis and artificial intelligence," *Applied Mathematical Modelling,* 2013.

[18] C. M. Fonseca and P. J. Fleming, "An Overview of Evolutionary Algorithms in Multiobjective Optimization," *Evolutionary Computation,* vol. 3, pp. 1-16, Spr 1995.