Saeed, Bakhtiar I. and Mehrdadi, Bruce

Zero overshoot and fast transient response using a fuzzy logic controller

Original Citation

Saeed, Bakhtiar I. and Mehrdadi, Bruce (2012) Zero overshoot and fast transient response using a fuzzy logic controller. In: The 17th International Conference on Automation and Computing (ICAC 2011). IEEE, London, pp. 116-120. ISBN 9781467300001

Proceedings of the 17th International Conference on Automation & Computing, University of Huddersfield, Huddersfield, UK, 10 September 2011

Zero Overshoot and Fast Transient Response Using a Fuzzy Logic Controller

Bakhtiar I. Saeed & Bruce Mehrdadi
School of Computing and Engineering,
University of Huddersfield
Queensgate, Huddersfield HD1 3DH, UK
b.saeed@hud.ac.uk, b.mehrdadi@hud.ac.uk

Abstract— In some industrial process control systems it is desired not to allow an overshoot beyond the setpoint or a threshold, this could be a safety constraint or the requirement of the system. This study investigates the achievement of a zero overshoot step response using a fuzzy logic controller. A fuzzy PID controller is applied to stable, marginally stable and unstable systems and their step responses are compared with a PID controller. A comparative case study shows that the proposed fuzzy controller outperforms conventional PID controller in achieving zero overshoot response.

Keywords - fuzzy PD+I; PID controller; zero overshoot; scaling gain; tuning.

I. INTRODUCTION

An integral part in controller design and analysis is to achieve a satisfactory response in transient time and steady state. The characteristics of these states can be represented in parameters such as: overshoot, rise time, settling time and steady state error. In a stable system, the transient response exists for a short period of time. However, this might cause problems in some applications. For example, in some chemical processes, it is desired to have zero overshoot or an overshoot that does not exceed a specific threshold.

The conventional Proportional-Integral-Derivative (PID) controllers, which are the most popular feedback methods for their robustness and simplicity [1, 2] have some limitations, particularly when they are applied to obtain zero overshoot. These controllers can be tuned in several ways [3] to achieve zero overshoot if possible, but most of the time this is at the expense of rise time, and vice versa.

Some methods have been reported by researchers to find the values of PID gains to achieve zero overshoot [4-7]. By relating the step response overshoot to the positions of zeros and poles of a transfer function, a method has been derived to find the parameters of PID controller [6]. This method has been used to avoid overshoot in second order and lower order systems. A cascade sliding mode-PID controller has been proposed in literature [7].

On the other hand fuzzy logic controllers have been applied successfully in industrial processes and in some cases outperform PID controllers [8], in particular when the controlled system is complex or non-linear, as this is the case in many process control systems [9].

Nonetheless, designing fuzzy controllers is challenging. There is no systematic process for the design of fuzzy logic controllers that will produce a high-performance controller for a wide range of applications [10, 11]. For example, it is difficult to find the relation between selecting membership function type or rule base, and the controller performance such as better rise time or less overshoot. In addition, unlike conventional controllers, fuzzy controllers have several parameters that can be adjusted, such as membership function shape, rules and scaling gains. Furthermore, there is no general rule of tuning these parameters. However, some techniques applied in tuning conventional controllers can still be utilised to some extent [10].

In this study, a fuzzy PID controller is adopted and is applied to different second order systems. Initially the controller gains are fixed and then manually tuned to achieve zero overshoot with a short rise time and settling time. A case study has been used to compare the performance of the fuzzy PID and conventional PID controllers for a second order system. The results show that fuzzy controllers outperform conventional controllers in achieving zero overshoot and fast transient response.

The remainder of this paper is organised as follows: Section 2 presents an overview of the fuzzy controller. Design and synthesis of the fuzzy PD+I controller is illustrated in section 3 using MATLAB and Simulink. Simulation results are shown in section 4. Finally, some conclusions are drawn in section 5.

II. STRUCTURE OF THE FUZZY CONTROLLER

The most widely used fuzzy controllers are fuzzy proportional-derivative controllers (FPD) and they act on two inputs: error and change in error (derivative of error) signals; therefore, designing rule base for these controllers is well understood and a straightforward procedure. This configuration exhibits a good performance at the transient response of the system, while encountering problems at the steady state when the error is close to zero [3, 12, 13].

To enhance the steady state performance of a system, integral action is required [3]. Thus the controller becomes fuzzy proportional-integral controller (FPI). Although these controllers have good performance, at the steady state they suffer from a slow response [12, 13].

Improving both the transient state and the steady state requires a controller that includes both derivative and integral actions. A fuzzy controller with this capability is

known as fuzzy proportional-derivative-integral controller (FPID).

In literature, various structures have been proposed to design FPID controllers [3, 13-15].

Fig.1 shows a simple design proposed in [3] and was adopted as the fuzzy PID controller in this study.

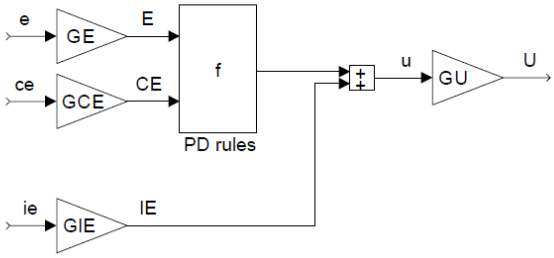

Figure 1: Fuzzy PD+I controller (FPD+I) [3]

The controller consists of a normal FPD with added integral action; therefore it is known as FPD+I. As the controller has three inputs: error, derivative of error and integral of error, it can provide all the benefits of conventional PID controllers, but still has some disadvantages such as derivative kick and integrator windup. Additionally, there is only one rule base with two inputs; therefore, designing the rule base is less complex than the structure proposed in [15] which has three input rule base. Furthermore, some techniques applied in tuning conventional PID controllers can still be utilised to some extent [3, 10].

III. SIMULATION OF CONTROLLERS

A. *Fuzzy PD+I*

Matlab (v7.9) and Simulink were used to build and simulate the model. Fig. 2 shows the FPD+I controller in a closed-loop feedback system.

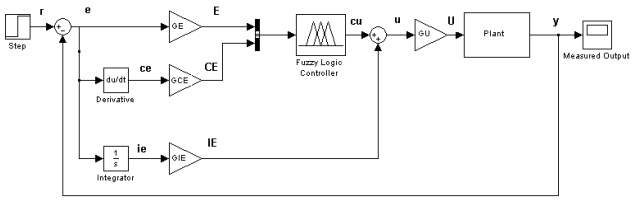

Figure 2: Simulink model of Fuzzy PD+I controller in a closed-loop control structure.

The plant block represents the desired transfer function to be controlled.

The fuzzy PD+I controller was formed by adding the integrator to the output of the fuzzy PD controller. The controller has three input signals: error (e), change in error (ce) and integral of error (ie). The error signal is obtained from the difference between the setpoint (r) and the measured plant output (y), the change in error signal and the integral of error signals are produced by passing the signals through derivative and integral blocks respectively. The inputs have scaling gains: gain of error (GE), gain of change in error (GE) and gain of integral error (GIE). These gains along with the output gain (GU) can be tuned to achieve better performance [3, 10, 15-17]. Adjusting these gains is used frequently for tuning fuzzy controllers and it has been regarded as an effective approach [12, 15, 16, 18-21]. First of all, they have a global effect on the performance of the controller; their effects can be easily observed [18]. Secondly, there are few parameters to tune, thus the tuning process is computationally efficient in contrast with other methods, where there are several parameters to tune. Finally, they can be considered as conventional controller gain parameters; therefore, they are convenient to tune and some ideas from conventional controller tuning can be borrowed [12, 16, 18-21].

The controller output (u) is formed by adding the integral of error (ie) to the output of the fuzzy PD controller (cu).

To represent the values of the e, ce and cu, five symmetric triangle shape membership functions (except trapezoid for the two at the extreme ends for e and ce) with 50% of overlap were chosen [3, 10]. Although the choice of membership function shape and width is subjective, triangular shapes were chosen, because they are more popular and convenient [10, 11]. The interval of [-1, 1] was used for the universes of discourse of the input variables, while [-2, 2] was used for the output variable. The linguistic descriptions of the input and output membership functions are negative large (NL), negative small (NS), zero (ZE), positive small (PS) and positive large (PL). These are shown in Fig. 3 and Fig. 4 respectively.

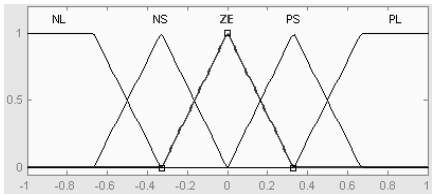

Figure 3: Error and change of error membership functions

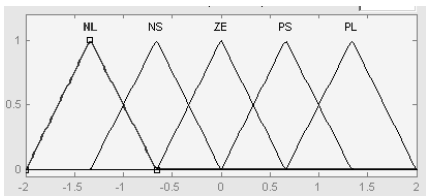

Figure 4: Output membership functions

The minimum operator was selected as an implication method, and the most popular and standard method of defuzzification process known as the centre of gratify (CoG) was selected.

The fuzzy rule-base is a mapping between the inputs, e and ce and the output, cu. A sample of the rule has the following form:

If error is PL *and change in error is* PL, *then control signal is* PL

The rule implies that if the error is positive large (measured output far away from the set point) and the change of error is positive large, then the control signal should be positive large to return back the output near the setpoint.

As there are 5 linguistic variables for each input, 25 rules were created using Fuzzy Logic Toolbox, Table I shows the rules.

TABLE I. RULE-BASE FOR THE FUZZY CONTROLLER

e \ ce	NL	NS	ZE	PS	PL
NL	NL	NL	NS	NS	ZE
NS	NL	NS	NS	ZE	PS
ZE	NS	NS	ZE	PS	PS
PS	NS	ZE	PS	PS	PL
PL	ZE	PS	PS	PL	PL

Three standard closed-loop performance criteria were chosen as design specifications to measure the performance of the controller [2]: maximum percentage overshoot (*Mp*), rise time (*tr*) and settling time (*ts*)

Finally, a script code was developed to simulate the model, calculate the *Mp*, the *tr* and the *ts* and generate the required plots.

B. PID controller

In order to compare the performance of the FPD+I controller with a conventional PID controller, a simulation model of a PID controller with auto tuning capability was created, this is shown in Fig. 5.

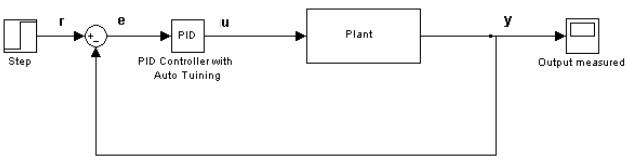

Figure 5: Simulink model of conventional PID controller in a closed-loop control structure.

The model contains a plant block that represents the desired transfer function to be controlled and a PID controller block with auto tuning capability. Also for this design, the same performance measures were chosen and a script code was developed to simulate the model.

IV. RESULTS AND DISCUSSIONS

A. Tests

In order to evaluate the design, three different systems: stable, marginally stable and unstable were simulated using the FPD+I and the conventional PID models. The transfer functions of these systems are provided in Table II.

TABLE II. DIFFERENT SYSTEM TRANSFER FUNCTIONS

System	Transfer Function	Stability
1	$\frac{1}{S^2+0.5S+1}$	Stable
2	$\frac{1}{S^2+1}$	Marginally stable
3	$\frac{1}{S^2-15S+1}$	Unstable

Initially, the gains of the FPD+I controller GE, GCE, GIE and GU are set to 1, and then tuned to achieve zero overshoot with a fast rise time and short settling time, the tuned values are shown in Table III.

TABLE III. F PD+I CONTROLLER TUNED GAIN VALUES

System	GE	GCE	GIE	GU
1	1	0.6	0.25	5
2	1	0.4	0.15	12
3	0.8	0.025	0.01	2800

For the conventional PID controller, the Matlab PID auto tuner was used to obtain the values of PID gains (P, I and D), these values are shown in Table IV.

TABLE IV. CONVENTIONAL PID CONTROLLER GAIN VALUES

System	P	I	D
1	11.74	0.85	8.85
2	12.04	1.09	12.088
3	3273.19	6345.72	325.43

The step responses of the three systems for the FPD+I and the conventional PID are combined together and shown in the Fig. 5 - Fig. 7.

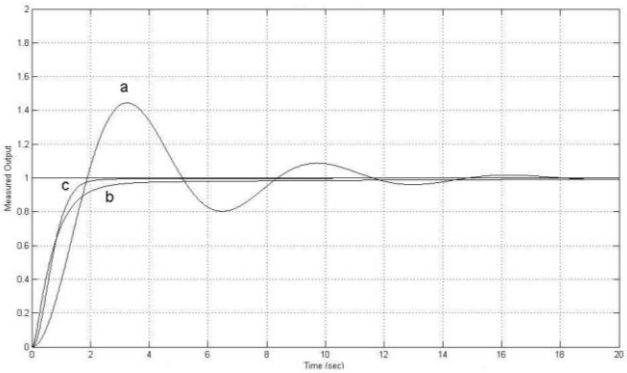

Figure 6: Simulation results for the first system: (a) open-loop. (b) Conventional PID (c) Fuzzy PD+I.

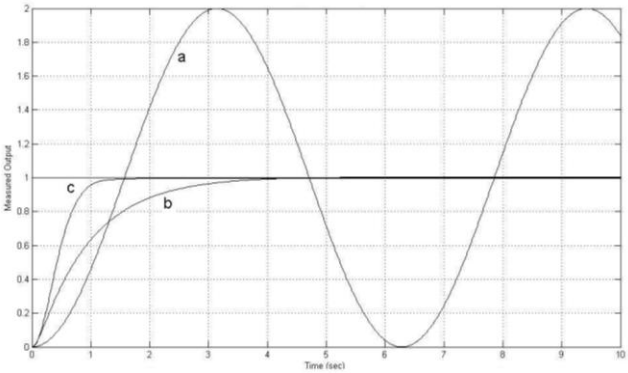

Figure 7: Simulation results for the second system: (a) open-loop. (b) Conventional PID (c) Fuzzy PD+I.

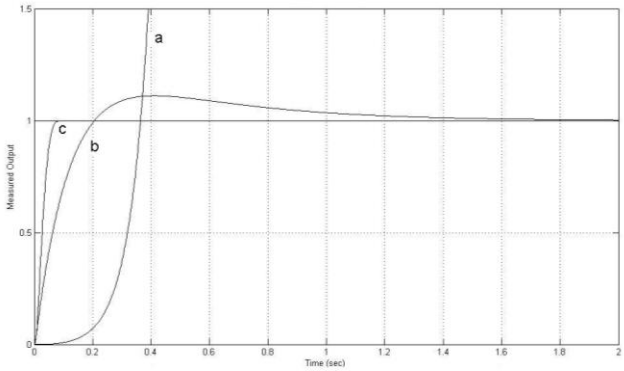

Figure 8: Simulation results for the third system: (a) open-loop. (b) Conventional PID (c) Fuzzy PD+I.

The performance measures of each controller are shown in Table V.

The performance measures of each controller and systems.

System	Performance measure	Open-loop	Conventional PID	Fuzzy PD+I
1	M_p	% 44.43	% 0	% 0.0
	t_r	1.25	1.63	1.10
	t_s	14.11	5.92	1.97
2	M_p	% 100.0	% 0.43	% 0.0
	t_r	1.01	1.97	0.67
	t_s	Not known	3.42	1.16
3	M_p	% 1.28 e+19	% 11.12	% 0.0
	t_r	0.13	0.14	0.041
	t_s	Not known	1.22	0.06

The results obtained from the fuzzy PD+I clearly indicate substantial improvements in transient response of systems have been achieved.

B. *Case study*

The transfer function of a chemical process shown in (1) has been used by other researchers [6] to achieve zero overshoot in the closed-loop response. Accordingly, the parameters of the PID controller were calculated (P=7.2, I=0.972 and D=6.99).

$$G(s) = \frac{1}{(1+S)(1+10S)} \qquad (1)$$

Three tests were conducted on the above system: the FPD+I controller with tuned gains, a conventional PID controller using the parameters proposed by the method in [6] and the conventional PID controller using Matlab auto tuner. The parameters of the three controllers were as follows: FPD+I (GE = 1, GCE = 0.7, GIE = 0.04, GU = 20), PID controller using the method in [6] (P = 7.2, I = 0.72, D = 6.99) (The original values were for KC, Ti and Td, they were converted to the values of P, I and D to be used within the setting of Matlab PID controller) and for the conventional PID controller using Matlab auto tuner (P = 1.74, I = 0.20, D = -4.58).

The closed-loop step responses of the three controllers along with the open-loop step response are shown in Fig. 9.

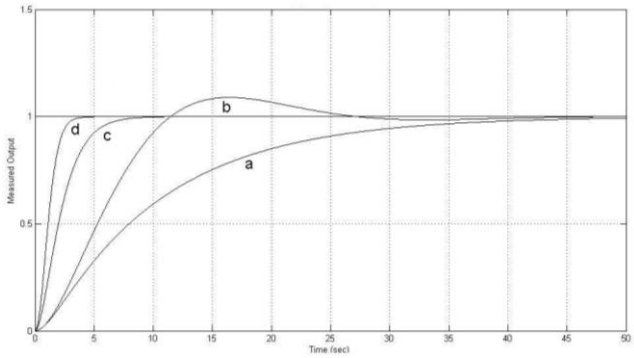

Figure 9: Simulation results: (a) open-loop. (b) Conventional PID (parameters found using Matlab auto tuner). (c) Conventional PID (parameters found using the method in [6]. (d) FPD+I.

The performance measures of the controller are shown in Table VI.

The performance measures of each controller.

PM	Open-loop	Conventional PID (parameters found using the method in [6])	Fuzzy PD+I	Conventional PID (parameters found using Matlab auto tuner)
Mp	% 0.0	% 0.0	% 0.0	% 8.92
tr	22.14	3.96	1.77	7.8546
ts	40.17	6.90	3.05	24.53

C. Discussions

It is evident from the results, that the FPD+I controller has achieved zero overshoot with faster rise time and shorter settling time compared to the conventional PID controller. Although the conventional PID controller has achieved zero overshoot for the first and second systems, it has been at the expense of the rise time and the settling time. Additionally, in the third system and in the case study the performance of the conventional PID controller was degraded as the response resulted with some overshoot.

V. CONCLUSIONS

A fuzzy PID controller was applied to a stable, marginally stable and unstable second order system. The results showed that fuzzy PID controller outperformed the conventional PID controllers in achieving zero overshoot and produced a faster transient response. This is an ongoing research and the next phase of the work will encompass the ability to fine tune the fuzzy gains automatically. Inclusion of predictive and intelligent agents in the fuzzy algorithms will reduce the tuning time.

REFERENCES

[1] K. J. AstrolÌˆm and B. r. Wittenmark, *Computer-controlled systems*. Upper Saddle River, N.J: Prentice Hall, 1997.

[2] R. C. Dorf and R. H. Bishop, *Modern control systems*. Upper Saddle River: Prentice Hall, 2001.

[3] J. Jantzen, *Foundations of fuzzy control*. Chichester: Wiley, 2007.

[4] Y. Aiping, "A Fast generalized predictive control algorithm with non-overshoot," in *Intelligent Control and Automation (WCICA), 2010 8th World Congress on*, 2010, pp. 3572-3575.

[5] Y.-S. Lu and C.-M. Cheng, "Design of a Non-Overshooting PID Controller with an Integral Sliding Perturbation Observer for Motor Positioning Systems," *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, vol. 48, pp. 103-110, 2005.

[6] A. Rachid and C. Scali, "Control of overshoot in the step response of chemical processes," *Computers & Chemical Engineering*, vol. 23, pp. S1003-S1006, 1999.

[7] Q. P. H. Thanh H. Tran, and Hung T. Nguyen, "Robust Non-Overshoot Time Responses Using Cascade Sliding Mode-PID Control," *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol. 11, 2007.

[8] S. R. Vaishnav and Z. J. Khan, "Design and Performance of PID and Fuzzy Logic Controller with Smaller Rule Set for Higher Order System," in *Proceedings of the World Congress on Engineering and Computer Science*, San Francisco, USA, 2007, pp. 24-26.

[9] R. Babuska and E. Mamdani, "Fuzzy Control," http://www.scholarpedia.org/article/Fuzzy_control, 2008.

[10] K. M. Passino and S. Yurkovich, *Fuzzy control*. Menlo Park, Calif: Addison-Wesley, 1998.

[11] I. H. Altas and A. M. Sharaf, "A generalized direct approach for designing fuzzy logic controllers in Matlab/Simulink GUI environment," *International Journal of Information Technology and Intelligent Computing*, 2007.

[12] H. Y. Chung, B. C. Chen, and J. J. Lin, "A PI-type fuzzy controller with self-tuning scaling factors," *Fuzzy Sets and Systems*, vol. 93, pp. 23-28, 1998.

[13] B. Subudhi, B. A. Reddy, and S. Monangi, "Parallel structure of fuzzy PID controller under different paradigms," in *Industrial Electronics, Control & Robotics (IECR), 2010 International Conference on*, 2010, pp. 114-121.

[14] O. Karasakal, M. Guzelkaya, I. Eksin, and E. Yesil, "Online rule weighting of fuzzy PID controllers," in *Systems Man and Cybernetics (SMC), 2010 IEEE International Conference on*, pp. 1741-1747.

[15] C. X. Yung C. Shin, *Intelligent Systems: Modeling, Optimization, and Control*: CRC Press. Taylor & Francis Group, 2009.

[16] X. Jie, L. Liyun, Derong, C. Yanbo, and W. Shiyu, "Fuzzy gain based adaptive fuzzy logic controller for BLDCM drive," in *Control Conference, 2008. CCC 2008. 27th Chinese*, 2008, pp. 159-163.

[17] L.-X. Wang, *Adaptive fuzzy systems and control*. Englewood Cliffs, N.J: PTR Prentice Hall, 1994.

[18] S. Chopra, R. Mitra, and V. Kumar, "Auto Tuning of Fuzzy PI Type Controller Using Fuzzy Logic," *INTERNATIONAL JOURNAL OF COMPUTATIONAL COGNITION (HTTP://WWW. IJCC. US)*, vol. 6, 2008.

[19] M. N. S. Melba Mary.P, Albert Singh. N., "Design of Intelligent Self-Tuning Temperature Controller for Water Bath Process," *INTERNATIONAL JOURNAL OF IMAGING SCIENCE AND ENGINEERING (IJISE)*, vol. VOL.1, 2007.

[20] M. Murad, K. C. Cheok, and M. Das, "Methodology to simplify the tuning process of self-organizing fuzzy logic controllers," in *Intelligent Engineering Systems, 2009. INES 2009. International Conference on*, 2009, pp. 57-60.

[21] J. Victor and A. Dourado, "Adaptive scaling factors algorithm for the fuzzy logic controller," in *Fuzzy Systems, 1997., Proceedings of the Sixth IEEE International Conference on*, 1997, pp. 1021-1026 vol.2.