Algorithms for Intelligent Systems
Series Editors: Jagdish Chand Bansal · Kusum Deep · Atulya K. Nagar

Rajesh Kumar
V. P. Singh
Akhilesh Mathur *Editors*

Intelligent Algorithms for Analysis and Control of Dynamical Systems

Springer

Algorithms for Intelligent Systems

This book series publishes research on the analysis and development of algorithms for intelligent systems with their applications to various real world problems. It covers research related to autonomous agents, multi-agent systems, behavioral modeling, reinforcement learning, game theory, mechanism design, machine learning, meta-heuristic search, optimization, planning and scheduling, artificial neural networks, evolutionary computation, swarm intelligence and other algorithms for intelligent systems.

The book series includes recent advancements, modification and applications of the artificial neural networks, evolutionary computation, swarm intelligence, artificial immune systems, fuzzy system, autonomous and multi agent systems, machine learning and other intelligent systems related areas. The material will be beneficial for the graduate students, post-graduate students as well as the researchers who want a broader view of advances in algorithms for intelligent systems. The contents will also be useful to the researchers from other fields who have no knowledge of the power of intelligent systems, e.g. the researchers in the field of bioinformatics, biochemists, mechanical and chemical engineers, economists, musicians and medical practitioners.

The series publishes monographs, edited volumes, advanced textbooks and selected proceedings.

More information about this series at http://www.springer.com/series/16171

Rajesh Kumar · V. P. Singh ·
Akhilesh Mathur
Editors

Intelligent Algorithms for Analysis and Control of Dynamical Systems

Springer

Editors
Rajesh Kumar
Department of Electrical Engineering
Malaviya National Institute of Technology
Jaipur, India

V. P. Singh
Department of Electrical Engineering
Malaviya National Institute of Technology
Jaipur, India

Akhilesh Mathur
Department of Electrical Engineering
Malaviya National Institute of Technology
Jaipur, India

ISSN 2524-7565 ISSN 2524-7573 (electronic)
Algorithms for Intelligent Systems
ISBN 978-981-15-8044-4 ISBN 978-981-15-8045-1 (eBook)
https://doi.org/10.1007/978-981-15-8045-1

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Preface

Intelligent Algorithms for Analysis and Control of Dynamical Systems is a collection of research articles and critical review articles presented in the International Conference on 'Advances in Systems, Control and Computing (AISCC-2020)', organized by the Department of Electrical Engineering and Department of Electrical Engineering of Malaviya National Institute of Technology, Jaipur, India, during February 27–28, 2020. This was a very unique conference which combined systems, control, and computing under one roof. Moreover, it is a matter of honor for MNIT Jaipur to learn that Springer was associated with AISCC-2020 as a major publication sponsor for the event. This edited book of this conference is published under Algorithms for Intelligent Systems (AIS).

Algorithms for Intelligent Systems (AIS) of Springer publishes research on the analysis and development of algorithms for intelligent systems with their applications to various real world problems. It covers research related to autonomous agents, multi-agent systems, behavioral modeling, reinforcement learning, game theory, mechanism design, machine learning, meta-heuristic search, optimization, planning and scheduling, artificial neural networks, evolutionary computation, swarm intelligence and other algorithms for intelligent systems.

The book series includes recent advancements, modification and applications of the artificial neural networks, evolutionary computation, swarm intelligence, artificial immune systems, fuzzy system, autonomous and multi agent systems, machine learning and other intelligent systems related areas. The material will be beneficial for the graduate students, post-graduate students as well as the researchers who want a broader view of advances in algorithms for intelligent systems. The contents will also be useful to the researchers from other fields who have no knowledge of the power of intelligent systems, e.g. the researchers in the field of bioinformatics, biochemists, mechanical and chemical engineers, economists, musicians and medical practitioners.

The chapters of Intelligent Algorithms for Analysis and Control of Dynamical Systems cover the problems of single/multivariable control systems and provide the necessary background for performing research in the fields of control/automation, systems, and computing. Aimed at helping readers understand the classical and

modern design of different intelligent automated systems, this book presents coverage on the control of linear and nonlinear systems, advances in systems, and advances in computing along with real-time control applications, etc.

The review committee has done an excellent job in reviewing the articles and approving the high quality research articles to be published in the conference proceedings. The editors are thankful to all of the faculty and students of these various committees for their dedication in making this a very successful conference and also to the editing and printing support staff of Springer for making the compilation possible. We sincerely hope that this volume will inspire researchers.

Prof. Rajesh Kumar
Professor, EED, MNIT, Jaipur, India

Dr. V. P. Singh
Assistant Professor, EED, MNIT, Jaipur, India

Dr. Akhilesh Mathur
Assistant Professor, EED, MNIT, Jaipur, India

Contents

Editors and Contributors

About the Editors

Dr. Rajesh Kumar received the Bachelor of Technology in Engineering degree with honors in Electrical Engineering from the Department of Electrical Engineering, National Institute of Technology, Kurukshetra, India, in 1994, Master of Engineering with honors in Power Engineering from the Department of Electrical Engineering, Malaviya National Institute of Technology, Jaipur, India, in 1997, and Ph.D. degree in Intelligent Systems from the Department of Electrical Engineering, Malaviya National Institute of Technology (MREC, University of Rajasthan), India, in 2005. He is currently Professor at the Department of Electrical Engineering; Adjunct Professor at Centre of Energy and Environment at Malaviya National Institute of Technology, Jaipur, India. He has been a Research Fellow (A) at the Department of Electrical and Computer Engineering at National University of Singapore from 2009 to 2011. He has been Reader from 2005 to 2009, Senior Lecturer from 2000 to 2005 and Lecturer from 1995 to 2000 at the Department of Electrical Engineering, Malaviya National Institute of Technology. He is the Founder of ZINE student innovative group. His background is in the fields of computational intelligence, artificial intelligence, intelligent systems, power and energy management, robotics, bioinformatics, smart grid and computer vision. Dr. Kumar research interests focus on intelligent systems, machine intelligence, power management, smart grid and robotics. Dr. Kumar has published over 500 research articles, has supervised 20 Ph.D. and more than 30 M.Tech. thesis. He has 12 patents to his name. He has delivered more than 100 expert talks in various conferences and workshops. He received 03 academic awards, 12 best paper awards, 06 best thesis awards, 04 professional awards and 25 student awards. He has received the Career Award for Young Teachers in 2002 from the Government of India. He is on 12 Journal Editorial Boards. He is an Associate Editor of IEEE Access, Associate Editor, Swarm and Evolutionary Computation, Elsevier, Associate Editor, IET Renewable and Power Generation, Associate Editor, International Journal of Bio Inspired Computing, and Associate Editor, CAAI

Transactions on Intelligence Technology, IET. Dr. Kumar is a Senior Member of IEEE (USA), Fellow of IET (UK), Fellow of IE (INDIA), Fellow of IETE, Life Member of CSI, Senior Member of IEANG and Life Member of ISTE. He was Honorary Secretary of IETE Rajasthan Chapter, Excom IEEE Delhi PES-IAS, and Vice Chairman of IEEE Rajasthan Subsection.

Dr. V. P. Singh graduated from Uttar Pradesh Technical University, Lucknow, India, in 2007 with B.Tech. degree in Electrical Engineering. He received M.Tech. degree in Control and Instrumentation and Ph.D. degree in Electrical Engineering in 2009 and 2013, respectively, from Motilal Nehru National Institute of Technology, Allahabad, India. Currently, he is working as Assistant Professor at Malaviya National Institute of Technology, Jaipur, India. Prior to joining Malaviya National Institute of Technology, Jaipur, India, he was Assistant Professor at National Institute of Technology Raipur, India. His research interests include system modeling, model order reduction and applications of optimization.

Dr. Akhilesh Mathur received the B.E. degree in Electrical Engineering from the Rajasthan University, Rajasthan, India, in 2007 and the M.Tech. degree in Electrical Engineering from IIT Roorkee, Roorkee, India, in 2010 with majors in power system. He received his Ph.D. degree at IIT Roorkee, Roorkee, India, in the stream of Electrical Engineering in 2018. He is currently working as an Assistant Professor in the Department of Electrical Engineering at MNIT Jaipur, Jaipur, India. His research interests include energy planning of smart grid, load flow and short-circuit analysis of unbalanced distribution systems with inverter-based distributed generation, optimal operation of distributed generations and planning of microgrids.

Contributors

Moina Ajmeri Department of Electrical Engineering, National Institute of Technology, Patna, India

Govind Anil National Institute of Technology, Warangal, Telangana, India

Raghav Ankur KPIT Technologies Ltd., Pune, India

Hemendra Arya Indian Institute of Technology Bombay, Mumbai, India

Arundhatti Bezbaruah Electrical Engineering Department, National Institute of Technology Kurukshetra, Kurukshetra, India

V. Bharath Kumar Vellore Institute of Technology - Andhra Pradesh (VIT-AP) University, Amaravati, Andhra Pradesh, India

Godavarthi Charan Vellore Institute of Technology-Andhra Pradesh (VIT-AP) University, Amaravati, Andhra Pradesh, India

Abrar Ahmed Chhipa College of Technology and Engineering, Udaipur, Rajasthan, India

M. Chidambaram Department of Chemical Engineering, National Institute of Technology, Warangal, India

Kishan S. Chowhan Indian Institute of Technology Bombay, Mumbai, India; Aeronautical Development Agency, Bangalore, India

Sudeepa Das National Institute of Technology, Raipur, India

Sujit Das National Institute of Technology Warangal, Warangal, India

Girish S. Deodhare Aeronautical Development Agency, Bangalore, India

Suhail Ganiny Mechanical Engineering Department, NIT Srinagar, Srinagar, Jammu and Kashmir, India

Ediga Chandramohan Goud Department of Chemical Engineering, National Institute of Technology, Warangal, India

Sandeep B. Hake Cummins Emission Solutions, Cummins Technical Centre India, CES, Pune, India

Ravi Kumar Jatoth National Institute of Technology, Warangal, Hanamkonda, India

R. R. Joshi College of Technology and Engineering, Udaipur, Rajasthan, India

Akhilesh Kr. Gupta Department of Electrical Engineering, MNNIT, Allahabad, Prayagraj, U.P., India

Bhavnesh Kumar Department of ICE, Netaji Subhas University of Technology, New Delhi, India

Deepak Kumar Department of Electrical Engineering, MNNIT, Allahabad, Prayagraj, U.P., India

Satyendra Kumar Department of Electrical Engineering, National Institute of Technology, Patna, India

Vinod Kumar College of Technology and Engineering, Udaipur, Rajasthan, India

Suhail Manroo Mechanical Engineering Department, NIT Srinagar, Srinagar, Jammu and Kashmir, India

Satanand Mishra AcSIR & CSIR-Advanced Materials and Processes Research Institute (AMPRI), Bhopal, India

Priya Misra Department of ICE, Netaji Subhas University of Technology, New Delhi, India

Jose Mrudula National Institute of Technology Warangal, Warangal, India

Vijaya Kumar Munagala National Institute of Technology, Warangal, Hanamkonda, India

Naresh Kumar Nagwani National Institute of Technology, Raipur, India

Divya Pandey Madan Mohan Malaviya University of Technology, Gorakhpur, U.P., India

Ritik Pandey National Institute of Technology Warangal, Warangal, India

Shivani Pandey AcSIR & CSIR-Advanced Materials and Processes Research Institute (AMPRI), Bhopal, India

Vijay V. Patel Aeronautical Development Agency, Bangalore, India

Y. V. Pavan Kumar Vellore Institute of Technology - Andhra Pradesh (VIT-AP) University, Amaravati, Andhra Pradesh, India

Bhanu Pratap Electrical Engineering Department, National Institute of Technology Kurukshetra, Kurukshetra, India

Ritika Roy National Institute of Technology, Raipur, India

Ajit Safai KPIT Technologies Ltd., Pune, India

Tirath Prasad Sahu National Institute of Technology, Raipur, India

Dasa Sampath Vellore Institute of Technology-Andhra Pradesh (VIT-AP) University, Amaravati, Andhra Pradesh, India

Paulson Samuel Department of Electrical Engineering, MNNIT, Allahabad, Prayagraj, U.P., India

K. Sandeep Rao Vellore Institute of Technology - Andhra Pradesh (VIT-AP) University, Amaravati, Andhra Pradesh, India

A. Seshagiri Rao Department of Chemical Engineering, National Institute of Technology, Warangal, India

Sakshi Sharma Electrical Engineering Department, National Institute of Technology Kurukshetra, Kurukshetra, India

Aditya Narayan Singh National Institute of Technology Warangal, Warangal, India

Chhabindra Nath Singh Department of Electrical Engineering, H.B. Technical University, Kanpur, U.P., India

V. N. Siva Praneeth Vellore Institute of Technology - Andhra Pradesh (VIT-AP) University, Amaravati, Andhra Pradesh, India

M. Siva National Institute of Technology, Warangal, Telangana, India

S. Srinath National Institute of Technology, Warangal, Telangana, India

G. Uday Bhaskar Babu National Institute of Technology, Warangal, Telangana, India

Shripati Vyas College of Technology and Engineering, Udaipur, Rajasthan, India

Nikhil K. Yadav Faculty of Electrical and Computer Engineering, Arba Minch Institute of Technology, Arba Minch University, Arba Minch, Ethiopia

Improving Transient Response of Cuk Converters Using Artificial Neural Network Based Controller

Godavarthi Charan, Dasa Sampath, V. N. Siva Praneeth, and Y. V. Pavan Kumar

1 Introduction

In many DC energy applications, the source DC voltage cannot be directly given to the load as in many cases, the load needs a high DC voltage, where source cannot deliver this directly. So, it is required to have a power electronic converter with high efficiency and benefits. There are three traditional DC-DC converters available, namely buck, boost, and buck–boost. Buck converter only decreases the voltage, and boost converter only increases voltage, whereas buck–boost converter does both. However, buck-boost converter requires pulsed current and gives output current as pulsed signal. But, most of the applications need continuous current flow in the system, so uses cuk converter. Cuk converter increases and decreases the input DC voltage and further ensures continuous output current, which varies it from the buck–boost converter. The operation of both buck–boost converter and cuk converter is same, but the energy transfers from input to output differ from each other. In buck–boost converter, the energy transfer takes place through an inductor, and in cuk converter, the energy transfer takes place through a capacitor. Further, cuk converter and buck–boost converter have same voltage ratio, but the main advantage in cuk

G. Charan · D. Sampath · V. N. Siva Praneeth · Y. V. Pavan Kumar (✉)
Vellore Institute of Technology-Andhra Pradesh (VIT-AP) University, Amaravati, Andhra Pradesh 522237, India
e-mail: pavankumar.yv@vitap.ac.in

G. Charan
e-mail: charan.18bec7075@vitap.ac.in

D. Sampath
e-mail: sampath.18bec7101@vitap.ac.in

V. N. Siva Praneeth
e-mail: praneeth.18bev7022@vitap.ac.in

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_1

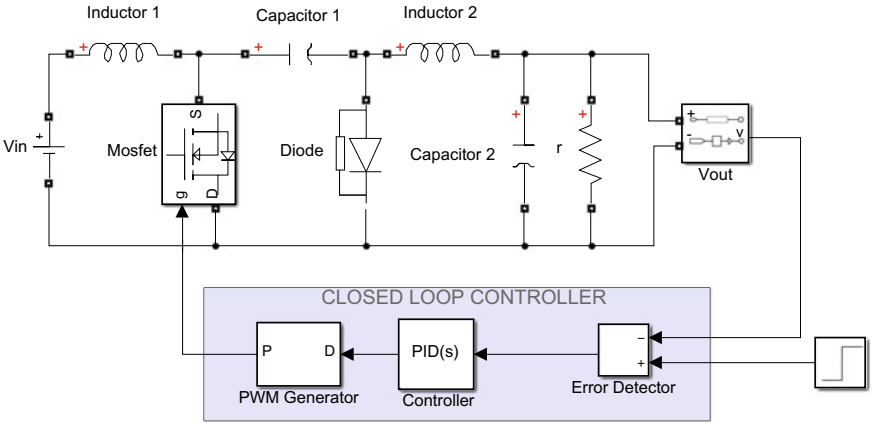


Fig. 1 Circuit diagram of the cuk converter with its control loop

converter is having lesser ripple content than a buck–boost converter, which improves its efficiency.

As shown in Fig. 1, cuk converter is designed with two inductors (inductor1, inductor2) and two capacitors (capacitor1, capacitor2) along with load resistor [1–4]. Based on the diode and MOSFET (acts like a switch) present in the circuit, converter works as either to increase or decrease the given input to get the required output. The input voltage flows through inductor1, which acts as a filter inductor. The operation of the circuit is as follows.

If MOSFET is in ON condition, the current passes through inductor1 and produces magnetic field in initial level. Due to this, the diode works in reverse bias and causes energy waste from elements, which helps to store in the output level.
If MOSFET is in OFF condition, the inductor1 helps to keep the current which is passing through it to collapse the magnetic field by turning the polarity and source current. Due to this, the energy will pass through the capacitor1, where the transformation of energy depends on the output level of circuit.

Here, the capacitor2 plays a role to keep the constant voltage at the output. Generally, in cuk converter, transformation of capacitive energy is higher than the transformation of inductive energy. The relation of the source and output voltages is given in (1), where D_c is the duty cycle of MOSFET, which is adjusted continuously to control the output voltage.

$$\frac{V_{out}}{V_{in}} = \frac{-D_c}{1 - D_c} \tag{1}$$

Cuk converter is used in many applications like renewable energy, electrical vehicles, DC motors, etc. It is always important to design a controller to resist disturbances and to get the output faster. There are many conventional PID control methods such

as open-loop transient response methods, error performance index methods, and sustained oscillations methods existing [5, 6]. But, it is hard to predict which method suits the best to a given system. Further, it is very tough to implement all methods and compare their effectiveness. Since all these are offline tuning methods and contain single-layered architecture, they may not be able to produce a good response when a disturbance occurs.

In order to overcome all these disadvantages with conventional methods, there is a dire need for introducing an artificial intelligence-based controller. There are many artificial intelligence techniques like fuzzy logic [7, 8], artificial neural network [9], model-based control, genetic algorithms [10, 11], etc., implemented in the literature for different applications. In line with that and based on the importance of power electronic converters in the present-day applications, this chapter proposes artificial neural network-based PID (ANN-PID) controller design for cuk converters. To implement the controller, the cuk converter transfer function is considered as given in (2) [12].

$$T(s) = \frac{-0.4904s^3 + 7923s^2 + (1.074 \times 10^7)s + (3.502 \times 10^9)}{(5 \times 10^{-6})s^4 + 1.001s^3 + 129.1s^2 + 410600s} \tag{2}$$

2 Implementation of the Proposed ANN-PID Controller

2.1 Architectures of ANN

ANN is used to process the information. Actually, it is motivated by the structure of biological nervous systems like information processing, brain, etc. ANN is interconnected with processing elements named as neurons. These neurons are arranged in the form of layers to become a network which executes equal and distributed computations. In Fig. 2a, simple architecture of ANN is shown. The ANN-based models have ability to recognize, learn the data patterns, and adapt changes in the environment like as human brain. In feed-forward network of multilayer perceptron, all are unidirectional connection and loops are absent in the network, which results in neuron links to successive layer neurons only. In Fig. 2b, simple architecture of feed-forwarding network is shown with inputs $\alpha_1, \alpha_2, \ldots, \alpha_m$ and output β. Weights that are connected to the hidden layer are denoted as w_{ij} and w_x, respectively, for output layer.

The arrangement of a periodic network is similar to a human brain network. These structures involve loops and backward links, which varies it from other conventional structures. Till the reach of an equilibrium point, the behavior of these networks is dynamic. Figure 2c shows a typical topology of recurrent networks. The radial-basis function (RBF) network is most used type of neural network after Elman recurrent

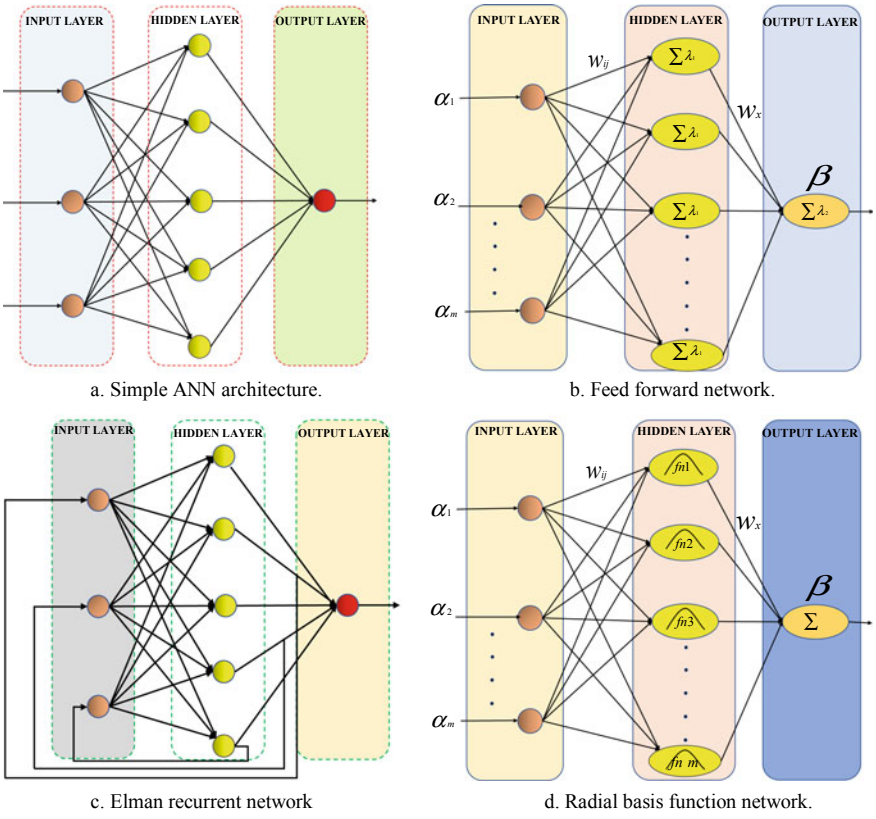


Fig. 2 General architectures (or) networks of ANN

networks, which is also used for recognizing patterns. Clustering, spline interpolation, and approximating functions are some other applications of RBF. Figure 2d represents the network structure of a two-layered RBF topology. This chapter uses feed-forward network structure to implement the proposed ANN-PID controller as it is simple and yet effective for controller applications when compared to other network structures.

3 Training Algorithms of ANN

There are mainly three algorithms used for training the ANNs, namely Bayesian regularization (BR), scaled conjugate gradient (SCG), and Levenberg–Marquardt (LM), out of which BR and SCG algorithms are considered as effective [13]. So, this paper uses those two algorithms to implement the proposed ANN-PID controller. SCG is one of the best conjugate gradient methods, which is widely used for the

feed-forward neural networks [14]. SCG uses the second-order approximation for step size and search direction, which varies it from general optimization strategy. Optimal distance is calculated by iterative approach. The distance to be moved is given by (3), and the present search direction is found by the line search. The search direction is done by the conjugate to the preceding search. Here, P_x is function of a_x which is a scalar quantity. It regulates the indefiniteness of the Hessian matrix. BR is much better algorithm than standard back-propagation nets [15]. Generally, all algorithms need to do lengthy cross-validation, but BR can reduce that process. BR can never be easily over-trained or over-fitted, and also, the irrelevant weights can be turned off. The basic objective function is given in (4), which can be modified by BR algorithm. The generalization capability of model will be improved by this modification. The objective function adds up another term which is sum of squares of network weights. Thus, the revised objective function becomes (5).

$$w_{x+1} = w_x + (a_x \times p_x) \tag{3}$$

$$f_s = E_d = \frac{1}{N}\sum_{i=1}^{N}(e_i)^2 \tag{4}$$

$$f_s = lE_d + mE_w \tag{5}$$

where l, m are the parameters that are to be maximized by Bayesian framework.

4 Implementation of ANN-PID Controller

Various steps involved in the implementation of ANN-PID controller are shown in Fig. 3. The very first step in implementing ANN-PID controller is data collection, and the more available data leads the better response from the system. In order to get best response, training data should be more accurate and precise. Here, in the proposed ANN-PID controller, this training data is PID gain parameter values that are obtained for different ranges of disturbances. To prepare this dataset, the conventional best

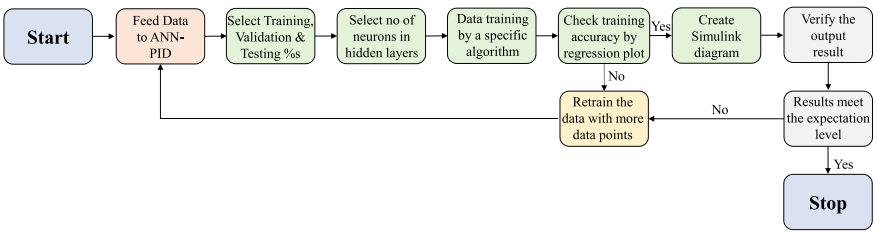


Fig. 3 Flowchart representing implementation of ANN-PID controller

method named, modified Ziegler–Nichols (MZN), is used as a reference. Further, the implemented BR and SCG based proposed ANN-PID controllers are compared with the conventional MZN-PID controller for understanding the importance of the proposed methods. Once data is prepared, the percentage of data used for validation, training, and testing should be defined. This chapter considers 75% of dataset values for training, 15% for validation, and 10% for testing. The number of neurons in hidden network layer is selected as 10.

This network is now trained with BR and SCG algorithms. The accuracy of training data can be checked from the regression plot, where the value of regression coefficient should be nearer to value 1. A brief overview on this regression coefficient and their responses are discussed in Sect. 3. This way, the required ANN trainer for feeding the PID controller gain values will be ready. The Simulink model of overall system is shown in Fig. 4, which includes the cuk converter system implemented with the conventional MZN-PID controller and the proposed BR- and SCG-based ANN-PID controllers for comparing their effectiveness. The ANN will feed the values of K_p, K_i, K_d based on the detected error signal as shown in Fig. 5, according to the obtained response and the set point.

This obtained or actual response of the converter is varied in accordance to the occurrence of the disturbances into the system. Table. 1 represents the training data used to train the ANN network, based on which the computations of K_p, K_i and K_d values will be done for certain range of disturbances. Even the disturbance is beyond

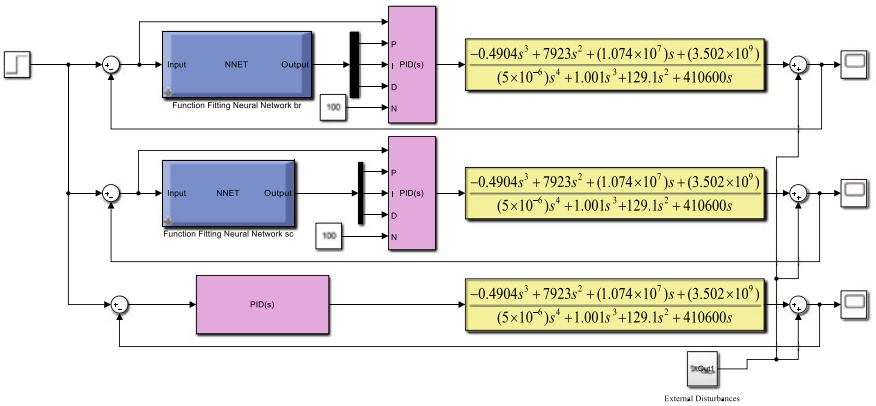


Fig. 4 System model with conventional and proposed controllers developed in Simulink

Fig. 5 Trained ANN block inputs and outputs representation

Table 1 PID controller gain parameters used to train the ANN

Voltage deviation from the reference value		PID parameter values		
Minimum	Maximum	Parameter	Lower	Upper
0.005	0.0097	K_p	0.0065	0.00445
		K_i	0.00084	0.00091
		K_d	$4.86e^{-05}$	$4.39e^{-05}$
0.0097	0.0388	K_p	0.0043	0.00209
		K_i	0.00092	0.00105
		K_d	$4.38e^{-05}$	$3.67e^{-05}$
0.0388	0.155	K_p	0.0021	0.0007
		K_i	0.00105	0.0022
		K_d	$3.60e^{-05}$	$2.84e^{-05}$
0.155	0.1705	K_p	0.0007	0.00065
		K_i	0.0022	0.0028
		K_d	$2.84e^{-05}$	$1.60e^{-05}$
0.1705	0.21375	K_p	0.00065	0.00052
		K_i	0.0028	0.0043
		K_d	$1.60e^{-05}$	$1.08e^{-06}$

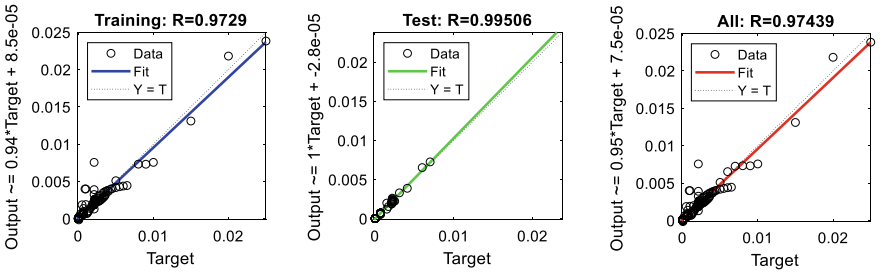
these ranges, the ANN will approximate the output values. This unique feature makes ANN-PID controller more efficient and robust than any other conventional method.

5 Results and Analysis

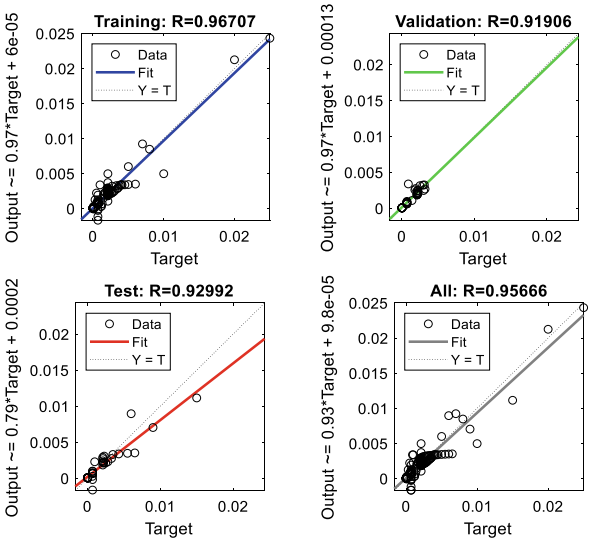
The effectiveness of the proposed ANN-PID controller over the conventional MZN-PID is assessed using training accuracy results, time-domain transient responses, and frequency-domain stability margins as given follows.

5.1 Analysis on Training Accuracy

Regression analysis is used to assess training accuracy. The regression plots of BR and SCG algorithms are shown in Fig. 6a, b respectively. Training, validation, and testing data are plotted by using regression plot. It indicates the linear relationship between targets and output by plotting mean square error of data. Here, the regression coefficient (R) values are nearer to 1, which shows that the relation between output and target is good, thereby estimating the best outputs. Further, it can be observed that the values of 'R' are much nearer to 1 with BR algorithm than SCG algorithm,



(a). Regression plot obtained with BR algorithm



(b). Regression plot obtained with SCG algorithm

Fig. 6 Regression plots obtained for the proposed ANN model

which means that the BR algorithm develops a good relation between input and outputs than SCG algorithm.

5.2 Analysis on System Response for Different Conditions

The time-domain response of the system obtained with conventional and proposed controller is shown in Fig. 7. This plot is analyzed in three parts: Fig. 8 shows the transient state response, Figs. 9 and 10 show system response when a disturbance of 5% of reference input is applied and ejected from the system, respectively. Table 2 shows the comparison with respect to various performance metrics. From this, the

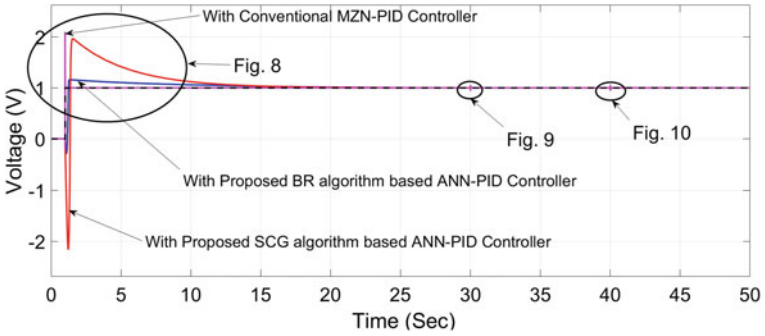


Fig. 7 System response with conventional and proposed controllers

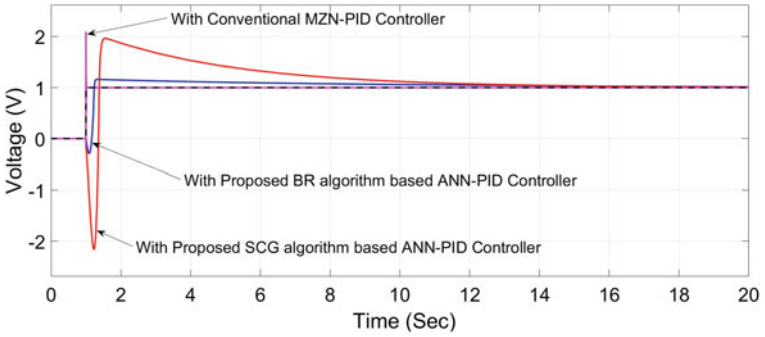


Fig. 8 System response during transient state with conventional and proposed controllers

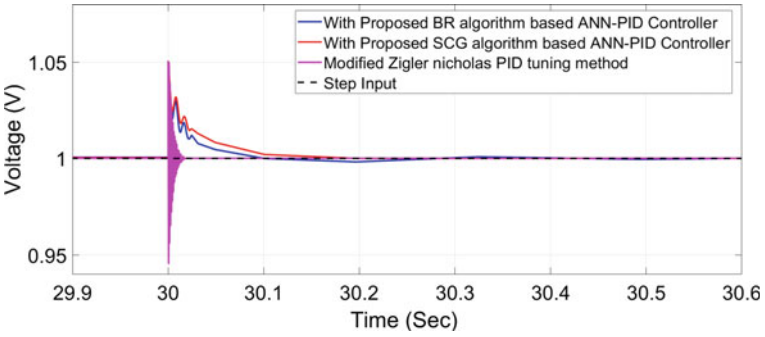


Fig. 9 System response with conventional and proposed controllers when disturbance is ON

better method is identified with respect to the response which possess lower peak time, peak overshoot, settling time, rise time, delay time, and oscillations.

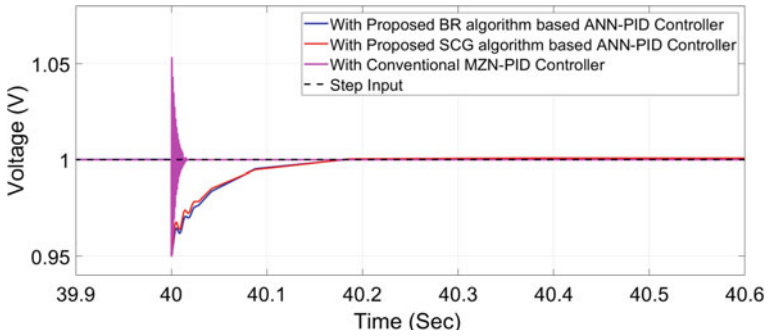


Fig. 10 System response with conventional and proposed controllers when disturbance is OFF

Table 2 Time-domain specifications computed for different methods

Method	Peak overshoot (%)	Peak time	Settling time	Rise time	Delay time	Oscillations
BR-PID	13	1.339	15.5	0.0537	1.2129	No
SCG-PID	93.6	1.545	15.595	0.0167	1.3727	No
MZN-PID	106	1.0005	1.0153	0.0001	1.00015	Yes

Further, Figs. 11 and 12 show the frequency response of the system with the proposed BR and SCG algorithms-based ANN-PID controller. These plots represent that the both systems are stable based on gain margin and phase margin computations.

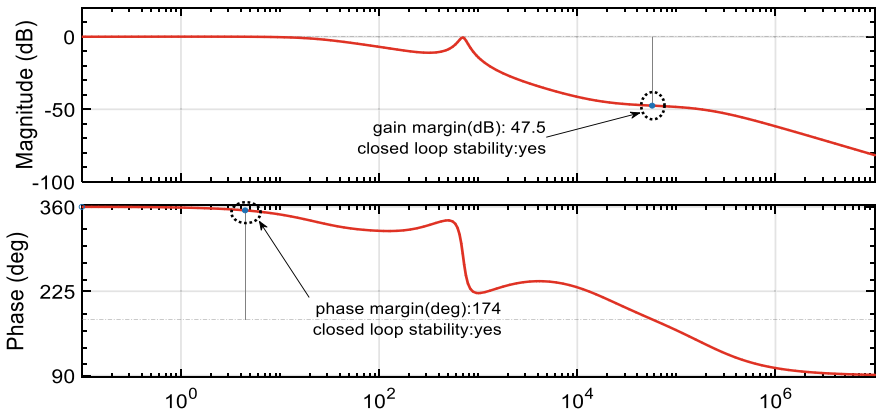


Fig. 11 System frequency response with the proposed BR algorithm-based ANN-PID controller

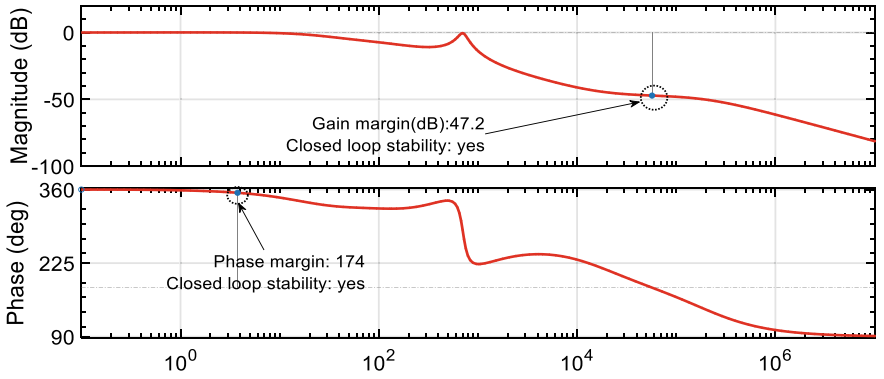


Fig. 12 System frequency response with the proposed SCG algorithm-based ANN-PID controller

6 Conclusion

This chapter addresses the drawback of classical offline PID tuning method. Hence, an ANN-PID controller is proposed for cuk converter application in this chapter. Further, two algorithms of ANN are evaluated with respect to regression plots and performance metrics. From the results, the following observations could be made.

- Conventional MZN-PID gives the fast response, but it produces high peak overshoot value with much oscillations, which may cause damage to the system.
- The proposed SCG algorithm-based ANN-PID has no oscillations, but it also produces a high peak overshoot value, which may cause damage to the system.
- The proposed BR algorithm-based ANN-PID gives the output without oscillations and possesses very less peak overshoot.

Thus, by considering all these factors, it is concluded that the proposed BR algorithm-based ANN-PID controller gives the better response for cuk converter application.

References

1. R. Pandey, B. Singh, A power factor corrected LLC resonant converter for electric vehicle charger using cuk converter, in *IEEE International Conference on Power Electronics, Drives and Energy System* (Chennai, India, 2018)
2. B.B. Tuvar, M.H. Ayalani, Analysis of a modified interleaved non-isolated cuk converter with wide range of load variation and reduced ripple content, in *3rd International Conference on Trends in Electronics and Informatics* (Tirunelveli, India, 2019)
3. A. Anand, B. Singh, Modified dual output cuk converter-fed switched reluctance motor drive with power factor correction. IEEE Trans. Power Electron. **34**(1), 624–635 (2019)

4. P.K. Moroti, P. Sanjeevikumar, P. Wheeler, F. Blaabjerg, M. Rivera M, Modified high voltage conversion inverting cuk DC-DC converter for renewable energy application, in *IEEE Southern Power Electronics Conference* (Puerto Varas, Chile, 2017)
5. H. Wu, W. Su, Z. Liu, PID controllers: design and tuning methods, in *9th IEEE Conference on Industrial Electronics and Applications* (Hangzhou, China, 2014)
6. V. Indragandhi, R. Selvamathi, T. Arunkumari, *Speed Control of a Switched Reluctance Motor Using PID Controller for PV Based Water Pumping Applications* (Innovations in Power and Advanced Computing Technologies, Vellore, India, 2017)
7. T.A.Srinivas, G. Themozhi, Fuzzy logic controlled cuk converter-inverter fed induction motor drive with reduced torque ripple, in *Second International Conference on Science Technology Engineering and Management* (Chennai, India, 2016)
8. N. ArunKumar, T.S. Sivakumaran, C. Shanmugasundaram, Simulation and real time implementation of fuzzy controller for CUK converter operating in buck mode, in *International Conference on Computing, Communication and Networking* (St. Thomas, VI, USA, 2008)
9. Q. Gao, K. Qi, Y. Lei, Z. He, An improved genetic algorithm and its application in artificial neural network training, in *5th International Conference on Information Communications and Signal Processing* (Thailand, 2005)
10. J. Wu, M. Liu, Improving generalization performance of artificial neural networks with genetic algorithms, in *IEEE International Conference on Granular Computing* (Beijing, China, 2005)
11. H. Zaeri, M.B. Poodeh, S. Eshtehardiha, Improvement of cuk converter performance with optimum LQR controller based on genetic algorithm, in *International Conference on Intelligent and Advanced Systems* (Malaysia, 2007)
12. B.K. Kushwaha, A. Narain, Controller design for cuk converter using model order reduction, in *2nd International Conference on Power, Control and Embedded Systems* (Allahabad, India, 2012)
13. K.S. Rao, V.N.S. Praneeth, Y.V.P. Kumar, D.J. Pradeep, Investigation on various training algorithms for robust ANN-PID controller design. Int. J. Sci. Technol. Res. **9**(2), 5352–5360 (2020)
14. S. Nayak, N. Kumar, B. Choudhury, Scaled conjugate gradient backpropagation algorithm for selection of industrial robots. Int. J. Comput. Appl. **7**(6), 92–101 (2017)
15. A. Payal, C.S. Rai, B.V.R. Reddy, Comparative analysis of bayesian regularization and levenberg-marquardt training algorithm for localization in wireless sensor network, in *15th International Conference on Advanced Communications Technology* (PyeongChang, South Korea, 2013)

Global Best Guided Crow Search Algorithm for Optimization Problems

Ritika Roy, Tirath Prasad Sahu, Naresh Kumar Nagwani, and Sudeepa Das

1 Introduction

In optimization problems, the objective function may have many local optima, but the ultimate goal is to find out global optima or the values near to global optima. Conventional methods are not capable of finding the global optimal solution as it may stuck with local optima. So, there is a need to develop other optimization techniques which is capable of handling local optima and finds global best solution for the given problem. In recent years, metaheuristic techniques are widely used to solve optimization problems as it has flexibility and adaptability characteristics to deal with the environment. Metaheuristic techniques are categorized into evolutionary-based, physics-based, bio-inspired, swarm-based and nature-inspired algorithms [1]. Evolutionary algorithm (EA) [2] is introduced so as to survive naturally in next generation using mutation and crossover. Genetic algorithm (GA) [2], differential evolution (DE) [3], genetic programming (GP) [4] are the most preferred EAs. The agents are modernized according to the physics phenomenon such as thermodynamics and heat exchange in physics-based algorithms (PBA). Simulated annealing (SA) [5], harmony search (HS) [6], gravitational search algorithm (GSA) [7] are mostly preferred PBAs. Bio-inspired algorithms (BIAs) are inspired from the biological

R. Roy (✉) · T. P. Sahu · N. K. Nagwani · S. Das
National Institute of Technology, Raipur 492010, India
e-mail: ritika.roy1234@gmail.com

T. P. Sahu
e-mail: tpsahu.it@nitrr.ac.in

N. K. Nagwani
e-mail: nknagwani.cs@nitrr.ac.in

S. Das
e-mail: sudeepa.das1992@gmail.com

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_2

behavior of living creatures. Artificial immune system (AIS) [8], bacterial foraging optimization (BFO) [9], dendritic cell algorithm (DCA) [10] belong to this category. Swarm-based algorithm (SBA) is generally inspired from the social behavior of swarms for finding food and settlers. Ant colony optimization (ACO) [11], artificial bee colony (ABC) [12], particle swarm optimization (PSO) [13] belong to this category. Nature-inspired algorithm (NIA) is inspired from the social relationship of organisms with each other for surviving in the environment. Firefly algorithm (FA) [14], bat algorithm (BA) [15], crow search algorithm (CSA) [16] are mostly preferred NIAs.

Crows are the most intelligent birds in the bird's fraternity. They have a very large brain compared to their body. Crows hide their food in undoubted location for the next season. But it is not that much easy for a crow to hide the food in a single place because some opponent crows always try to follow them [16]. Various applications of CSA are already introduced. However, there is still an insufficiency in CSA algorithm regarding the solution search equation. It is well known that the main objective of optimization technique is to counterbalance between exploration and exploitation. While we observed that the solution search equation of CSA is good in exploration but poor in exploitation. So, in order to improve the exploitation, we modify the solution search equation by applying the global best guided solution to CSA which is named as G-CSA.

The rest of the paper is structured as follows: Sect. 2 illustrates crow search algorithm (CSA); Sect. 3 describes the global best guided CSA (G-CSA). Section 4 shows results and experiments, and Sect. 5 concludes the paper and contains directions for future work.

2 Crow Search Algorithm (CSA)

Crows are one of the most intelligent birds among the bird fraternity. Recent study shows that crow's brain is slightly smaller than human brain. Abilities like self-awareness, communication in advanced ways, food storing and food hiding from other birds are the special characteristics of crows [17].

The crow has the capability to hide the overabundance food and to observe movement of the other crows which help the crow to steal the food of the targeted crow. From this experience, the crow learns to avoid becoming an easy target and tries to make fool to the opponent crow by taking that crow far from the hiding place. Askarzadeh [16] has proposed crow search algorithm (CSA) by formulating the behavior of crow mathematically which considers the following features.

- Crows lives in groups.
- The memory of crows to remember the hiding place of food.
- Crows observe and follow one another to rustle food.

Assuming the problem with d-dimension and number of crows in one group is N_p. The population of crows (X) in the search space is expressed mathematically in

Eq. (1).

$$X = \begin{bmatrix} X_1^1 & X_2^1 & \cdots & X_D^1 \\ X_1^2 & X_2^2 & \cdots & X_D^2 \\ \vdots & \vdots & \vdots & \vdots \\ X_1^{N_p} & X_2^{N_p} & \cdots & X_D^{N_p} \end{bmatrix} \tag{1}$$

The memory of the crows (MC) is the food hiding place of crows and can be mathematically expressed as in Eq. (2).

$$\mathrm{MC} = \begin{bmatrix} \mathrm{MC}_1^1 & \mathrm{MC}_2^1 & \cdots & \mathrm{MC}_D^1 \\ \mathrm{MC}_1^2 & \mathrm{MC}_2^2 & \cdots & \mathrm{MC}_D^2 \\ \vdots & \vdots & \vdots & \vdots \\ \mathrm{MC}_1^{N_p} & \mathrm{MC}_2^{N_p} & \cdots & \mathrm{MC}_D^{N_p} \end{bmatrix} \tag{2}$$

In CSA, two phases are considered in terms of the awareness of the crow. The two phases are described as follows.

Phase 1: In this phase, the jth crow is unaware about the ith crow $r_1 \geq AP$ by which ith crow will be benefitted by stealing the food of jth crow. The mathematical expression of this behavior of the crow is defined in Eq. (3) in which ith crow alters its position by concerning the memory of the jth crow.

$$X_{\text{new}}^i = X_{\text{old}}^i + \text{rand} \times \text{fl}^i \times \left(\text{MC}^j - X_{\text{old}}^i\right); \quad \text{if } r_1 \geq \text{AP} \tag{3}$$

where AP and fl are the awareness probability and flight length, respectively, rand and r_1 are two different uniform random numbers in a range of [0, 1].

Phase 2: In this phase, jth crow is aware of the ith crow $r_1 < \text{AP}$ by which jth crow tries to make fool to the ith crow. The jth crow misguides the ith crow by taking far away from the food hiding place. The mathematical expression of this phase is defined in Eq. (4).

$$X_{\text{new}}^i = \text{LB} + (\text{UB} - \text{LB}) \times \text{rand}; \quad \text{if} r_1 < \text{AP} \tag{4}$$

3 Global Best Guided CSA (G-CSA)

Generally, the performance of the population-based algorithm is measured by the capability of the algorithm to maintain the balance between exploration and exploitation. In Phase 1 of the CSA, the position of the ith crow is updated by concerning the memory of jth crow. And this may enhance the exploration capability of the algorithm. But it also enhances the probability to be trapped into local optima point. So,

this algorithm has better exploration capability rather than exploitation. This dilemma of CSA is reduced by introducing global best guided technique in Eq. (3). The position of the *i*th crow is updated by concerning the optimal point of corresponding iteration as described in Eq. (5).

$$X_{\text{new}}^{i} = X_{\text{old}}^{i} + \text{rand} \times \text{fl}^{i} \times \left(\text{MC}^{j} - X_{\text{old}}^{i}\right) + \text{rand} \times \left(X_{\text{Best}} - X_{\text{old}}^{i}\right) \quad \text{if } r_1 \geq \text{AP} \tag{5}$$

where X_{Best} is the global best value of corresponding iteration. By including the global best value of corresponding iteration, the exploitation capability of the algorithm is enhanced by guided toward the optimal point. In CSA, the exploration capability was sufficient enough, and by introducing global best guided technique, the exploitation capability can also be improved (Fig. 1).

4 Experimental Results

In this work, novel G-CSA is opted for its precise benefits such as easy implementation and less assumptions. Here, G-CSA is introduced for enhancing the exploitation capability. DE, PSO, SCA, CSA algorithms are opted to validate G-CSA algorithms to contribute optimum solution of some benchmark equations. All the algorithms are accomplished separately with 50 numbers of populations for 100 iterations as stopping criterion.

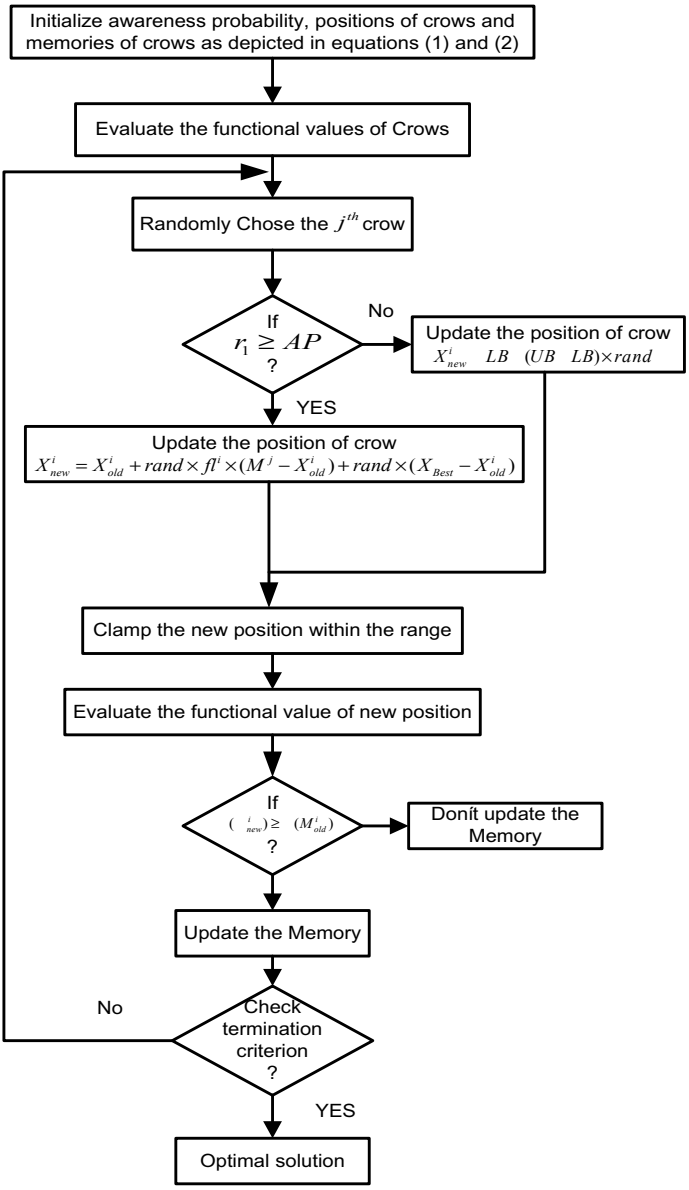


Fig. 1 Flowchart of G-CSA

Algorithm 1G-CSA Algorithm

Input : Engineering design problem consists of objective function, decision variables and constraints.

Auxillary : Set the values of *m, AP, fl,* $iter_{max}$ and *d*.

output : Optimized value for objective function.

while it<$iter_{max}$

  for $i \leftarrow 1$ **to** Np (all N crows of the group)

    Select a crow randomly to follow (j^{th} crow).

    Evaluate the best crow (X_{Best})

    if $r_1 \geq AP$

      Update the position of crow.

$$X_{new}^{i} = X_{old}^{i} + rand \times fl^{i} \times \left(MC^{j} - X_{old}^{i}\right) + rand \times \left(X_{Best} - X_{old}^{i}\right)$$

    else

      X_{new}^{i}=LB+(UB-LB)×rand

    end

  end

  Evaluate the fitness values of new crows.

  Update the memory of crows with better fitness values.

  Obtain the optimal solution.

End

it : Current iteration

itermax : Maximum no. of iteration

The benchmark equations opted for this work are tabulated in Table 1 with their expected results. The whole realization is graded based on the best value, mean and standard deviation of the solution. The performance parameters of different algorithms are tabulated in Table 2. The convergence of all algorithms is illustrated in Fig. 2. From Fig. 2 and Table 2, G-CSA contribute superior solution of benchmark equations than DE, PSO, SCA, CSA algorithms. Logistic regression and SMO performed well in all aspects of the career recommendation system.

5 Conclusion and Future Work

In this chapter, global best guided technique is implemented in CSA to enhance the efficacy of CSA. This technique is used to improve the exploitation capability of CSA. The experimental results tested on benchmark functions show that G-CSA with appropriate parameter outperforms the CSA algorithm. The graph also converges soon in the proposed algorithm. In future work, G-CSA function parameter ranging from 0 to 1 can be adaptive (as per iteration) to carry out better results.

Table 1 Benchmark equations opted for this work

No.	Functions	Range	D	Formulation	$f_{\min}$
1	Easom	$[-100, 100]$	2	$f(x) = -\cos(x_1)\cos(x_2) \exp\left((x_1-\pi)^2 - (x_2-\pi)^2\right)$	-1
2	Sphere	$[-100, 100]$	30	$f(x) = \sum_{i=1}^{D} x_i^2$	0
3	Rosenbrock	$[-30, 30]$	30	$f(x) = \sum_{i=1}^{D-1} 100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2$	0
4	Griewank	$[-600, 600]$	30	$f(x) = \frac{1}{4000}\left(\sum_{i=1}^{D}(x_i - 100)^2\right) - \left(\prod_{i=1}^{D}\cos\left(\frac{x_i - 100}{\sqrt{i}}\right)\right) + 1$	0
5	Ackley	$[-32, 32]$	30	$f(x) = -20\exp\left(-0.2\sqrt{\frac{1}{\pi}\sum_{i=1}^{D} x_i^2}\right) - \exp\left(\frac{1}{\pi}\sum_{i=1}^{D} cos(2\pi x_i)\right) + 20 + \mathrm{e}$	0
6	Schwefel2.22	$[-10, 10]$	30	$f(x) = \sum_{i=1}^{D}\lvert x_i\rvert + \prod_{i=1}^{D}\lvert x_i\rvert$	0
7	Rastrigin	$[-5.12, 5.12]$	30	$f(x) = \sum_{i=1}^{D}\left(x_i^2 - 10\cos(2\pi x_i) + 10\right)$	0
8	Colville	$[-10, 10]$	4	$f(x) = 100\left(x_1^2 - x_2\right)^2 + (x_1-1)^2 + (x_3-1)^2 + 90\left(x_3^2 - x_4\right)^2 + 10.1(x_2-1)^2 + (x_4-1)^2 + 19.8(x_2-1)(x_4-1)$	0
9	Booth	$[-10, 10]$	2	$f(x) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2$	0
'10	Michalewicz2	$[0, \pi]$	2	$f(x) = -\sum_{i=1}^{D}\sin(x_1)\left(\sin\left(ix_i^2/\pi\right)\right)^{20}$	-1.8013

Table 2 Comparison of DE, PSO, SCA, CSA and G-CSA

Functions	Performance measures	DE	PSO	SCA	CSA	G-CSA
F_1	BV	−4.6079e-6	0	−0.9997	−1	**−1**
	AVG	−0.2315	5.2361e−8	−0.9997	−0.7836	**−0.9246**
	SD	0.0534	5.2327e−7	0	0.3803	**0.2392**
F_2	BV	5.9501e4	8.3855e04	14.399	363.4292	**9.6804**
	AVG	4.2859e4	7.8516e4	14.399	2.4549e3	**948.2942**
	SD	1.1426e5	1.2708e4	0	4.5881e3	**2.8126e3**
F_3	BV	2.0937e8	2.5730e08	4.2306e4	2.0621e4	**1.8976e3**
	AVG	6.1891e7	2.9355e08	4.2306e4	2.6949e6	**4.5219e5**
	SD	4.0560e7	7.5156e07	0	1.2091e7	**2.8222e6**
F_4	BV	580.4083	644.0151	**0.0171**	2.4211	0.1975
	AVG	311.4130	696.3616	0	21.1084	**15.1263**
	SD	68.9855	114.1588	0.0171	45.1315	**31.8740**
F_5	BV	20.7132	21.0517	20.0550	5.7321	**3.7806**
	AVG	19.7470	20.8945	20.0550	9.1004	**7.2202**
	SD	0.2704	0.2518	0	3.3730	**3.4523**
F_6	BV	9.6309e9	2.5325e11	**0.3464**	9.7050	2.8853
	AVG	9.6424e8	7.9666e18	0.3464	3.2489e5	**91.1179**
	SD	2.6309e9	9.3511e19	0	2.3680e6	**782.5462**
F_7	BV	143.0740	170.7604	0.5726	1.0273	**0.0577**
	AVG	65.3510	217.7415	0.5726	6.6533	**3.9177**
	SD	32.9164	36.0415	0	12.6659	**9.4078**
F_8	BV	2.2280e3	4.1676e04	0.1730	4.2452e−4	**0.0094**
	AVG	310.8610	2.3433e05	0.1730	23.9487	**4.7451**
	SD	501.9932	2.2914e5	0	128.2247	**10.1640**
F_9	BV	8.2932	1.1805e3	1.3933e−4	3.5946e−10	**1.9676e-10**
	AVG	0.3243	317.2821	1.3933e−4	0.0445	**0.0156**
	SD	1.1751	374.6901	0	0.1940	**0.0806**
F_{10}	BV	−1.5777	−0.0080	−1.8010	−1.8013	**−1.8013**
	AVG	−1.7619	−0.3634	−1.8010	−1.7813	**−1.8005**
	SD	0.0665	0.4366	0	0.0737	**0.0029**

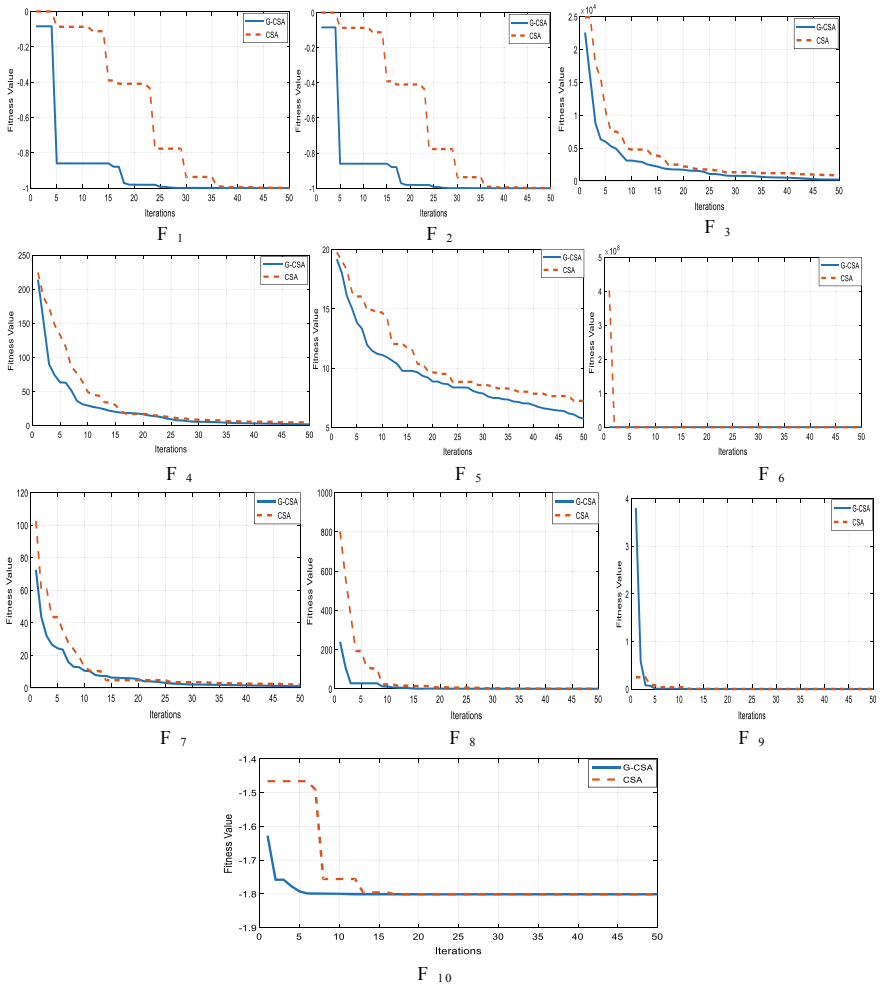


Fig. 2 Convergence plots

References

1. G. Dhiman, V. Kumar, Spotted hyena optimizer: a novel bio-inspired based metaheuristic technique for engineering applications. Adv. Eng. Softw. **114**, 48–70 (2017)
2. D.E. Goldberg, Genetic Algorithms in Search. Optimization, and Machine Learning (1989)
3. K.V. Price, Differential evolution: a fast and simple numerical optimizer, in *Proceedings of North American Fuzzy Information Processing* (IEEE. 1996), pp. 524–527
4. J.R. Koza, Genetic programming as a means for programming computers by natural selection. Stat. Comput. **4**(2), 87–112 (1994)
5. S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, Optimization by simulated annealing. Science **220**(4598), 671–680 (1983)
6. Z.W. Geem, J.H. Kim, G.V. Loganathan, A new heuristic optimization algorithm: harmony search. Simulation **76**(2), 60–68 (2001)
7. E. Rashedi, H. Nezamabadi-Pour, S. Saryazdi, GSA: a gravitational search algorithm. Inf. Sci. **179**(13), 2232–2248 (2009)
8. D. Dasgupta, Advances in artificial immune systems. IEEE Comput. Intell. Mag. **1**(4), 40–49 (2006)
9. S. Das, A. Biswas, S. Dasgupta, A. Abraham, Bacterial foraging optimization algorithm: theoretical foundations, analysis, and applications. Foundations Comput. Intell. **3**, 23–55 (2009)
10. J. Greensmith, U. Aickelin, J. Twycross, Articulation and clarification of the dendritic cell algorithm, in *International Conference on Artificial Immune Systems* (Springer, Berlin, Heidelberg. 2006), pp. 404–417
11. M. Dorigo, M. Birattari, T. Stutzle, Ant colony optimization. IEEE Comput. Intell. Mag. **1**(4), 28–39 (2006)
12. D. Karaboga, B. Akay, A comparative study of artificial bee colony algorithm. Appl. Math. Comput. **214**(1), 108–132 (2009)
13. J. Kennedy, R. Eberhart, Particle swarm optimization, in *Proceedings of ICNN'95-International Conference on Neural Networks*, vol. 4 (IEEE, 1995), pp. 1942–1948
14. X.S. Yang, X. He, Firefly algorithm: recent advances and applications. https://arXiv.org/1308.3898 (2013)
15. X.S. Yang, A new metaheuristic bat-inspired algorithm, in *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)* (Springer, Berlin, Heidelberg 2010), pp. 65–74
16. A. Askarzadeh, A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm. Comput. Struct. **169**, 1–12 (2016)
17. G.I. Sayed, A.E. Hassanien, A.T. Azar, Feature selection via a novel chaotic crow search algorithm. Neural Comput. Appl. **31**(1), 171–188 (2019)

Artificial Intelligent-Based Sliding Mode Controller for State Feedback Discrete-Time System

Nikhil K. Yadav

1 Introduction

Real-world systems are engrossed with in-built nonlinearities, multivariable, uncertainty in parametric variations, and external environmental threats. Numerous nonlinear control problems assume non-affine-in-control nonlinear systems. First time the variable structure control was designed [1]. In [2], the researcher developed the variable structure control with sliding mode control method. Linear matrix inequality (LMI) approach to sliding mode control is designed for class of uncertainty time-delay system in [3]. The sliding mode control technique is used in adaptive control strategies [4, 5]. Robust backstepping is used for fault estimation [6]. Discrete-time variable structure controller is used in computer numerical control (CNC) servomechanism [7]. Theory of fuzzy control has been described [8]. Stability is described for discrete-time fuzzy system with varying time-delay [9]. First time, fuzzy system was introduced [10].

Amalgamated technique is developed by fuzzy sliding control approaches and this technique generated better performances. Sudden variation in velocity of state on the sliding surface is described by researchers [11]. Therein, reference [12] described intelligence based sliding mode controller. The fuzzy logic controller is being designed on human being intelligence, i.e., heuristically gained knowledge of individual and the proposed work prove to be more stable when the uncertainties are occurring and that has been shown by simulated system. Unstructured systems are highly complex for a mathematical modeling. Fuzzy formulation will be more effective for resolving problems to obtain the desired solutions [13, 14]. Jae [15], described robust fuzzy control for nonlinear systems with parametric uncertainties.

N. K. Yadav (✉)
Faculty of Electrical and Computer Engineering, Arba Minch Institute of Technology, Arba Minch University, Arba Minch, Ethiopia
e-mail: drnyadav8@gmail.com

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_3

In [16], stability and stabilization of fuzzy system are described. Continuous-time system is sampled and quantized by incorporation of sampling time. It will convert continuous–time into discrete-time system with train of pulses. It breaks the continuation of signal in the pulses at sampling rate. The development of discrete-time technique in which the sampling process will be incorporated for the continuous-time systems to generate discrete-time systems. It will provide faster responses, fast calculations, reducing parallax error which leads to saving of time, cost factor and facilitates for better performance. Development of silicon material-based semiconductor technology for very large-scale integrated circuits (VLSI) reduces the size and cost of components. Discrete-time-designed microcontrollers and microprocessors increase the functional speed of operating system.

In this chapter, the digital fuzzy sliding mode control algorithm has been designed for state feedback system to generate stabilization of system. System output states are feedback and actuating discrete-time error signal is generated. Sliding manifold is designed on error dynamics. Designed sliding manifold is asymptotically stabilized. Designed discrete-time fuzzy sliding mode controller will help to obtain stabilization of unstructured systems. Stability of unstructured system is very complex task. This proposed algorithm will facilitate to improve the performance of unstructured system. This chapter has been organized as follows:

Section 2 describes the problem statement and system description, Sect. 3 illustrates the digital signal processing of the system, Sect. 4 illustrates the sliding mode controller design, and Sect. 5 proposed fuzzy logic controller. Section 6 describes the simulation results and finally Sect. 7 leads to describe the conclusion.

2 Problem Statement and System Description

Consider the nonlinear plant below:

$$\dot{x}(t) = f(x(t), u(t), t)) \tag{1}$$

$$\dot{x}(t) = Ax(t) + Bu(t) + Bf(t) \tag{2}$$

where $x(t) \in \Re^n$ and $u(t) \in \Re^m$ are the system's state and control input vector, respectively.

Assumption 1: Pair $(A,\ B)$ is completely controllable.

3 Discretization of System

The discrete-time system is obtained as follows:

$$x(k+1) = f(x(k), u(k), f(k), k) \tag{3}$$

$$x(k+1) = \Omega x(k) + \Theta u(k) + \Theta f(k) \tag{4}$$

where $x(k) \in \Re^n$ and $u(k) \in \Re^m$ are the system's state and control input vector in discrete-time domain by discretizing the continuous-time system. $\Omega \in \Re^{nxn}$ and $\Theta \in \Re^{mx1}$ are the system's state matrix and control vector, respectively. Input in phased disturbance $f(k) \in \Re^{mx1}$ incurred in the system. k is the sampling period of the system discretization. Input disturbance contains all unknown and undesired inputs which come from external disturbances, parametric uncertainties, parameter variations, and ill-modelled dynamics. The objective of this chapter is to design a sliding mode controller to guarantee asymptotic stability of the system.

4 Sliding Mode Controller Design

Sliding mode controller is designed based on knowledge of the system. Discrete-time system is consisted in phased input uncertainty.

$$x(k+1) = \Omega x(k) + \Theta u(k) + \Theta f(k) \tag{5}$$

The following assumption holds here as below:

Assumption 2: The variation of uncertainty $f(k)$ is bounded bandwidth of positive constant w. The bandwidth of uncertainty must satisfy the given condition $|f(k) - f(k-1)| \leq w$. Sliding mode conditions are exhibited as follows:

$$S(k) \leq 0 \tag{6}$$

where $S(k)$ is sliding manifold

$$S(k+1) \leq 0 \tag{7}$$

$$S(k+1) < S(k) \leq 0 \tag{8}$$

error dynamics of the system is generated to design sliding manifold as below

$$S(k) = \sigma e(k) \approx \sigma(x(k) - x_{\text{ref}}(k)) \tag{9}$$

error dynamics $e(k)$ is computed as follows:

$$e(k) = (x(k) - x_{\text{ref}}(k))$$

$x_{ref}(k) \in \Re^n$ is reference trajectory. $\sigma = \begin{bmatrix} \sigma_1 & \sigma_2 \end{bmatrix}$ is designing coefficient. $\sigma \in \Re^n$ row vector, σ determines the slopes of sliding manifold. The control law of discrete-time sliding mode control (DSMC) is designed as below

$$u(k) = \hat{f}(k) + (\sigma B)^{-1}[\sigma x_{\text{ref}}(k) - \sigma \Omega x(k) + qS(k) - \eta S(k)] \tag{10}$$

$$\hat{f}(k) = \hat{f}(k-1) + [S(k) - qS(k-1) + \eta(S(k-1))] \tag{11}$$

where q is reaching speed parameter on the sliding manifold. η is switching manifold gain. $\hat{f}(k)$ is estimated uncertainty. Estimated error uncertainty is calculated as follows:

$$\gamma(k) = f(k) - \hat{f}(k) \tag{12}$$

Sliding manifold stability is calculated below:

$$\begin{aligned} S(k+1) &= \sigma e(k) \approx \sigma(x(k+1) - x_{\text{ref}}(k+1)) \\ S(k+1) &= \sigma[\Omega x(k) + \Theta u(k) + \Theta f(k)] - \sigma x_{\text{ref}}(k+1) \\ S(k+1) &= \sigma \Omega x(k) - \sigma \Theta \hat{f}(k) - \sigma \Theta q(k) \\ &\quad + \sigma \Theta f(k) - \sigma x_{\text{ref}}(k) \\ &\quad + (\sigma x_{\text{ref}}(k+1) - \sigma \Omega x(k) + qS(k) - \eta \text{sat}(S(k))) \\ &= qS(k) - \eta \text{sat}(S(k)) + \sigma \Theta \gamma(k) - \sigma \Theta q(k) \end{aligned} \tag{13}$$

Lemma 1 [15] Given constant matrices U, V, W and a symmetric constant matrix Y of appropriate dimensions, the following inequality holds:

$$Y + UVW + V^T W^T U^T < 0, \tag{14}$$

where W satisfies, $W^T W \leq N$ if and only if for some factor $\zeta > 0$

$$Y + \begin{bmatrix} \zeta^{-1} V^T & \zeta U \end{bmatrix} \begin{bmatrix} N & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \zeta^{-1} V \\ \zeta U^T \end{bmatrix} < 0 \tag{15}$$

Theorem 1 [9] If P, Q and Z are symmetric positive definite matrices. Scalar $\lambda > 0$ such that reduced order discrete-time system is asymptotically stable via sliding manifold (13)

$$\begin{bmatrix} X - N & * & * & * \\ 0 & -X & * & * \\ A_{11}N + A_{12}Z & Z & \overline{X} & * \\ W_1 N + W_2 Z & W_3 N + W_4 Z & 0 & -\lambda I \end{bmatrix} \prec 0 \tag{16}$$

where $X = P^{-1}QP$, $N = P^{-1}$, $Z = NP^{-1}$, $\bar{X} = - + \lambda UU^T$ and * denotes the transposed elements in the symmetric positions.

5 Designing of Fuzzy Controller

Unstructured systems are amalgamation of numerous coexist set of linear systems. Discrete-time unstructured system is represented by the following Takagi and Sugeno (T–S) fuzzy model.

Plant Rule i:

IF $\theta_1(k)$ is $M_{i1}(k)$ and …and $\theta_s(k)$ is $M_{is}(k)$.

THEN

$$x(k+1) = A_i x(k) + B_i u(k) \tag{17}$$

here $i \in \Re = \{1, 2, \cdots\cdots r\}$ where r is the number of IF–THEN rules; $M_{is}(k)$ is the fuzzy sets; $x(k) \in \Re^n$ is the state vector; $u(k) \in \Re^m$ is input vector. $\theta(k) = [\theta_1(k), \theta_2(k), \ldots, \theta_s(k)]$ are the premise variable.

Assume that in discrete-time fuzzy system, the premise variables independent input variable $u(k)$.

$$x(k+1) = \sum_{i=1}^{r} h_i(\theta(k))\{A_i x(k) + B_i u(k)\} \tag{18}$$

where the fuzzy basis functions are given by

$$h_i(\theta(k)) = \frac{\mu_i(\theta(k))}{\sum_{i=1}^{r} \mu_i(\theta(k))} \tag{19}$$

with $\mu_i(\theta(k)) = \prod_{j=1}^{s} M_{ij}(\theta_i(k))$, where $M_{ij}(\theta_i(k))$ is the grade of membership $\theta_i(k)$ in M_{ij}. Membership functional value.

$$\mu_i(\theta(k)) \geq 0, i \in \Re \text{ and } \sum_{i=1}^{r} \mu_i(\theta(k)) > 0 \tag{20}$$

for all k. $h_i(\theta(k)) \geq 0, i \in \Re$ and $\sum_{i=1}^{r} h_i(\theta(k)) = 1$.

Discrete-time T-S fuzzy model of the state given by

$$x(k+1) = \overline{A}(k)x(k) + \overline{B}u(k) \tag{21}$$

$$\overline{A}(k) = \sum_{i=1}^{r} h_i(\theta(k))A_i \tag{22}$$

$$\overline{B}(k) = \sum_{i=1}^{r} h_i(\theta(k))B_i. \tag{23}$$

Fuzzy control technique is more feasible for unstructured systems. Fuzzy controllers faced drawback of performance and stability fuzzy logic system performs a mapping from $i \in \Re$, $U \subset \Re^n$ to $V \subset \Re^n$. Let $U_i = U_1 \times \cdots \times U_n$, where $U_i \subset R^n \quad i = 1, 2, 3, \ldots, n$.

Fuzzyfier maps a crisp point in U_n into a fuzzy sets U_i. Fuzzy rule base consists of IF-THEN condition for igniting the rules:

$R^{(l)}$: IF $e_1(k)$ is F_1^l, and $e_2(k)$ is F_2^l, and … $e_n(k)$ is F_n^l,

THEN y_j is G_j^l, $\quad j = 1, 2, \ldots, p$.

Assume that all states are available for feedback of the system. Designed state feedback controller is an asymptotically stable. The state feedback fuzzy controller is represented by the following rules:

5.1 Controller Rule i

$$\text{IF } \theta_1(k) \text{ is } M_{i1}(k) \text{ and} \ldots \text{and } \theta_s(k) \text{ is } M_{is}(k). \tag{24}$$

Then

$$u(k) = K_i x(k) i \in \Re \tag{25}$$

where $K_i \in \Re^{n \times n}$, $i \in \Re$ is the local control gain. Thus, the controller $u(k)$ (25) expressed in following input–output fuzzy form.

$$u(k) = \sum_{i=1}^{r} h_i(\theta(k))K_i x(k) = \overline{K}(k)x(k) \tag{26}$$

substituting (20) into (15) and creating the closed-loop fuzzy system

$$x(k+1) = \left[\overline{A}(k)x(k) + \overline{BK}(k)x(k)\right] \tag{27}$$

The designed state feedback fuzzy controller is to determine the feedback gain K_i such that the closed-loop fuzzy system into (26) is asymptotically stable.

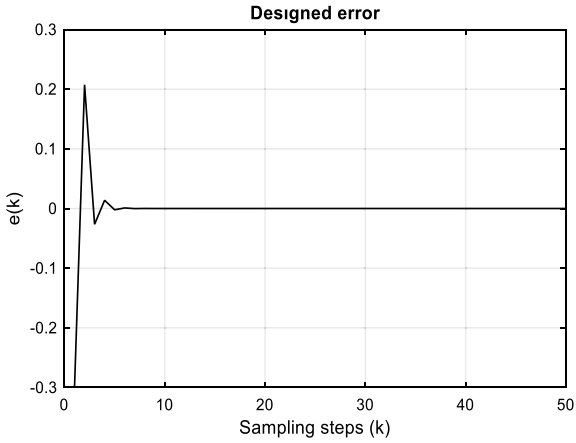


Fig. 1 Designed error of the system

6 Simulation Results

6.1 Designed Error Signal

Designed error is computed between the plant state and the reference model state of the system. The error magnitudes are reaching peak value of maximum 0.21 units in positive direction and -0.02 units in reverse direction. Tracking error $e(k)$ becomes stabilized after five sampling steps, i.e., 0.25 s with sampling period 0.05 s. Stabilization of tracking error exhibits the accurate desired set point. Control surfaces require lesser amount of force on actuator to follow the command input signal by the system performance index are quickly obtain the desired performance. In discrete-time domain, the error $e(k)$ becomes zero very quickly at five steps. The designed technique creates tracking error zero at faster rate (Fig. 1).

6.2 Sliding Manifold

An asymptotically stable sliding manifold is designed with a design coefficient parameter. On variation of the design coefficient parameter, the nonlinear functional value will be changed. On increasing the design coefficient parameter, the initial force required very large magnitude of 0.035 units in positive direction and 0.0780 units in reverse direction. The sliding manifold has the bandwidth of span of 0.113 units variation. This reveals that the actuating device needs a large travel efforts and hence, the device may go for the rated value and it will reduce the operating life. It is first overshoot value and very faster rate after 7 sampling steps, i.e., 0.35 s the sliding motion of sliding manifold becomes stabilized. Sliding manifold will remain

consistently zero position to meet the sliding motion characteristics for all future time of operation of system (Fig. 2).

6.3 Stable Output Response by Fuzzy Controller

The controller applies a control effort to the control surfaces 1180.00 units initially and becomes constant after six steps, i.e., 0.018 s. It seems that pierce force is reaching on close to rated peak value. This peak condition applied heavy stroking force on actuator. Bounded bandwidth should be developed to safe zone of actuator in the future course of time. The nonlinear function decreases exponentially on the increase of tuning parameter. The applied control effort will be a consistently constant magnitude of 1090.00 units for all time after 0.018 s for future operation of system (Fig. 3).

7 Conclusion

Practical nonlinear systems are facing numerous threats and ill-modelled. Proposed artificial intelligent sliding mode controller is made on the intelligence heuristic knowledge of human being. Error is computed by the controller upto peak magnitude of 0.21 units and stabilized after 7 sampling step, i.e., 0.35 s. Designed sliding manifold sliding motion in positive and negative direction with bandwidth of 0.11 units with peak positive 0.032 unit and negative peak -0.078 unit. The control efforts realized by the actuator peak magnitude of 1180 units and stabilized the control

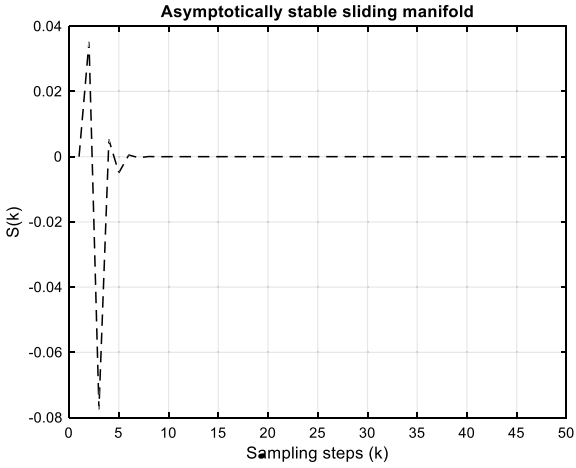


Fig. 2 Asymptotically stable sliding manifold

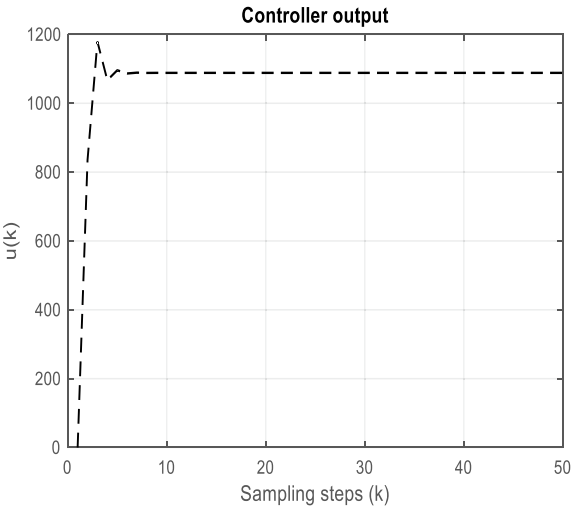


Fig. 3 Stable output response by proposed controller

efforts 1110 units after 6 sampling steps, i.e., 0.3 s. The proposed intelligent algorithm is more compatible even if in the variation of environmental condition such as disturbances and uncertainties. The simulated results are showing desirable stable performance after 0.30 s. This technique will more suited to highly unstable flying machineries. Future development can incorporate bounded bandwidth to reduce the closure peak rated value of control efforts.

References

1. S.V. Emelyanov, *Variable Structure Control Systems* (Nauka, Moscow, 1967)
2. V.I. Utkin, Variable structure systems with sliding modes. IEEE Trans. Autom. Control **22**, 212–222 (1977)
3. H.H. Choi, An LMI approach to sliding mode control design for a class of uncertain time-delay systems, in *Proceedings of European Control Conference (Karlsruhe)* (pp. 1022–1028) (1999)
4. M. Basin, P. Yu, Y. Shtessel, Hypersonic missile adaptive sliding mode control using finite-and fixed-time observers. IEEE Trans. Ind. Electron. **65**(1), 930–941 (2018)
5. H. Li, D. Yao, Adaptive sliding-mode control of Markov jump nonlinear systems with actuator faults. IEEE Trans. Autom. Control **62**(4), 1933–1939 (2017)
6. F. Chen, R. Jiang, K. Zhang, B. Jiang, G. Tao, Robust backstepping sliding-mode control and observer-based fault estimation for a quadrotor UAV. IEEE Trans. Ind. Electron. **63**(8), 5044–5056 (2016)
7. K. Zhang, H. Su, K. Zhuang, J. Chu, Comments on 'Discrete-time variable structure controller with a decoupled disturbance compensator and its application to a CNC servomechanism. IEEE Trans. Control Sys. Technol. **11**(1), 156–157 (2003)
8. K. Tanaka, *A Theory of Advanced Fuzzy Control* (Kyouritsu, Tokyo, Japan, 1994)

9. H. Gao, X. Liu, J. Lam, Stability analysis and stabilization for discrete-time fuzzy systems with time-varying delay. IEEE Trans. Syst. Man Cybern. Part B **39**(2), 306–317 (2009)
10. L.A. Zadeh, Fuzzy sets. Inf. Control **8**, 338–353 (1965)
11. C.Y. Dong, W. Yan, S. Huishe, Method to design Time-varied sliding surface. J. Shanghai Jiaotong Univ. **32**(6), 72–77 (1998)
12. A. Kumar, R.K. Singh, N.K. Yadav, New control algorithm based on artificial intelligence. Int. J. Inf. Technol. Knowl. Manage. **2**(1), 49–53 (2009)
13. S. Zhou, T. Li, Robust stabilization for delayed discrete-time fuzzy systems via basis-dependent Lyapunov-Krasovskii function. Fuzzy Sets Syst. **151**(1), 139–153 (2005)
14. Y.Y. Cao, P.M. Frank, Analysis and synthesis of nonlinear time-delay systems via fuzzy control approach. IEEE Trans. Fuzzy Syst. **8**(2), 200–211 (2000)
15. L.H. Jae, P.J. Bae, C. Guanrong, Robust fuzzy control of nonlinear systems with parametric uncertainties. IEEE Trans. Fuzzy Syst. **9**(2), 329–340 (2001)
16. H. Gao, X. Liu, J. Lam, Stability analysis and stabilization for discrete-time fuzzy system with time-varying delay. IEEE Trans. Syst. Man Cybern. Part B Cybern. **39**(2), 306–317 (2009)

A Comparative Study of Four Genetic Algorithm-Based Crossover Operators for Solving Travelling Salesman Problem

Aditya Narayan Singh, Jose Mrudula, Ritik Pandey, and Sujit Das

1 Introduction

Genetic algorithm (GA), a subclass of evolutionary algorithms, has been successfully applied by the researchers to solve various optimization problems including travelling salesman problems (TSP). Generally, evolutionary algorithms are part of heuristic search algorithms which do not always guarantee to provide the exact optimal solutions, but they will definitely help to find better optimal solutions as compared to other one within less amount of time. GAs are by nature adaptive optimization algorithms that mimic the process of natural selection and human genetics. Exploitation and exploration are two key concepts used in GA, where exploitation uses the existing knowledge to get better solutions and exploration is used to consider new solutions. Crossover and mutation are used to execute exploitation and exploration, respectively. In crossover, a pair of chromosomes are randomly selected for exchanging information between them which in turn generates two new chromosomes which are often found to be better individuals. In order to improve the functionality of genetic algorithms, it is important to use specific crossover operator. After reproduction or the selection process, the population consists of better individuals compared to old ones. Since the crossover operator is primarily responsible

A. N. Singh · J. Mrudula · R. Pandey · S. Das (✉)
National Institute of Technology Warangal, Warangal 506004, India
e-mail: sujit.das@nitw.ac.in

A. N. Singh
e-mail: adityans1923@gmail.com

J. Mrudula
e-mail: josemrudula@gmail.com

R. Pandey
e-mail: ritikpandey25@gmail.com

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_4

for improvements in fitness values of the individuals, this chapter aims to compare four significant crossover operators. Here, we present a comprehensive comparison among the high performing crossover operators, CX2, PMX, ERX, and OX in order to review their performance critically. We have observed the performance of these crossover operators for 3000 generations on 200 iterations.

Remaining sections of this chapter are organized as follows. Section 2 presents the related work for better understanding the paper content. We discuss the four crossover operators considered for comparison in Sect. 3 followed by a detailed comparative study in Sect. 4. Finally, we conclude in Sect. 5.

2 Related Work

Many researchers have contributed to solve TSP using GA. A brief presentation of the significant studies are given below. Khan and Khan [1] found the optimal solution for this problem is using binary matrix and new fittest criteria. This has been a very effective procedure which is applicable for both symmetric and asymmetric types of TSP. Philip [3] contributed to TSP by generating some number of random candidate routes (chromosomes) and then calculated their fitness values. Consequently, two best routes are selected and crossover is performed which in turn replaced the worst two routes. Their approach showed good results but run-time increases with an increase in population size. Ahmed [2] solved the TSP by developing a new crossover namely sequential constructive crossover using the representation given in [3]. The offspring was constructed by considering the minimum cost node as legitimate node which resulted in a quality solutions and reduced the burden of searching for the best chromosome. In [5], Jaya Lakshmi et al. developed a new hybrid crossover namely edge recombination crossover (ERX). Here, they maintained an edge map and the resultant offspring obtained the characteristics of the parents at the maximum extent which was not seen in many other crossover operations which are mentioned in this section. They constructed the offspring of a generation by considering the nearest neighbor of a current city which is the modified version of sequential constructive crossover. This crossover operator gives only one new offspring while all the others give two new offsprings. Modified cycle crossover was proposed by Hussain and Muhammad [7], where they modified the version of cycle crossover. The authors selected two parents and formed a cycle by sequentially searching the position of a city in both of the parents. This algorithm gave better results when compared with the basic ordered crossover. Xu [6] proposed a new crossover technique "random cross mapping method" (RCMX) based on random crossover which increased the diversity of population. They used randomly generated cross point and crossover section length (generally equal to half of the original sequence length) to perform crossover which was not used earlier. The next generation was created based on these two parameters-crossover point and crossover section length. In [9], Otman and Jaafar proposed a new crossover, namely "Partially Mapped Crossover" (PMX), where two random points are chosen and the portion between the points, one parent's string is

mapped to the other and the left-over remaining string's information is exchanged. This operator behaves like a mix of both cycle crossover and random cross mapping operator. Akter [8] solved TSP by introducing a type of crossover by combining the information of two parents following the approach of crossover points which has been used in partially mapped crossover and random cross mapping method. In their approach, the two crossover points are considered to be fixed and thus offsprings are generated. Repeated nodes are deleted and the missing nodes are added accordingly. Sallabi [4] developed the improved version of GA to solve TSP which included a new crossover operator and multi-mutation operator. The comparison of this algorithm with the previous related algorithms in the same area has shown better results for the solution in lesser time. Thus, the crossover operates use the concepts of cross points, partial mapping, and cyclic mapping.

3 Crossover Operators

In this section, we discuss the four specific crossovers which are considered for comparative study.

3.1 Random Cross Mapping Crossover (RCMX) [6]

RCMX mainly involves the concept of crossover points. It makes use of the single-point crossover and increases the diversity of population. By this technique, the offsprings obtain most of the parent's characteristics which is supposed to be an appropriate solution. In this method, a point is selected as a crossover point randomly between 0 to $n/2$, where $n =$ actual length of a parent. From the chosen crossover point (say k), the portion from k to $(k + (n/2))$ of both parents is swapped.

Example 1 Consider $P1 = (9\ 1\ 4\ 5\ 6\ 7\ 8\ 3\ 2)$ and $P2 = (6\ 8\ 1\ 2\ 3\ 9\ 5\ 4\ 7)$ be two chromosomes. Select the crossover point as 2. Then, swap the portion between the position 2 and 6 $(2 + 4)$. Hence, offsprings' are obtained as $O1 = (*\ *\ 1\ 2\ 3\ 9\ *\ *\ *)$ and $O2 = (*\ *\ 4\ 5\ 6\ 7\ *\ *\ *)$. Next final result is obtained as $O1 = (4\ 5\ 1\ 2\ 3\ 9\ 6\ 7\ 8)$ and $O2 = (8\ 1\ 4\ 5\ 6\ 7\ 2\ 3\ 9)$.

3.2 Edge Recombination Crossover (ERX) [8]

This crossover operator is the slightly modified version of the crossover operator proposed by Whitley [10]. This technique ensures that offspring should inherit the characteristics of the parent chromosomes as much as possible. It keeps track of the

information about the neighboring connections of a city by storing in edge map and thus the offspring is created. Here, a city is initially taken as "current city" from either of the two parents randomly. The current is always marked as visited. Then the entry related to this current city from the edge map is removed. If the current city has any common neighbor in both parents, it is added to the resultant offspring and this common city becomes the current city and the process is continued. Otherwise, the nearest city to the current city is identified from both parents and made as current city and this process is continued till all the cities are marked as visited.

Example 2 Consider $P1 = (5\ 2\ 1\ 4\ 3)$ and $P2 = (3\ 2\ 1\ 5\ 4)$ be two chromosomes and Dist (a, b) be the distance between the city a and city b. Assume that randomly selected current city is 2. Since 5, 1 and 3, 1 are the neighbors of 2 in $p1$ and $p2$, respectively, therefore 1 is selected. Now, 4 and 5 are remaining neighbors of 1. Since Dist(1, 4) < Dist(1, 5), 4 is selected. Similarly, we obtain offspring O = (2, 1, 4, 3, 5).

3.3 Partially Mapped Crossover (PMX) [9]

This crossover operator uses two randomly chosen cross over points to generate the offsprings, where the strings between two cross over points in parent chromosomes are mapped. Here, the offsprings mostly obtain the characteristics of the parent chromosomes. In PMX, initially, two crossover points are fixed for both the parents. Then, middle part between the crossover points is swapped and other bits from each parent which have no conflict are filled and then the remaining conflict bits are filled with the mapped one.

Example 3 Consider $p1 = (3\ 5\ 1\ 4\ 7\ 6\ 2\ 8)$ and $p2 = (4\ 6\ 5\ 1\ 8\ 3\ 2\ 7)$ be two chromosomes, and assume randomly chosen point are 3 and 6. The mapping here is $1 \leftrightarrow 5, 4 \leftrightarrow 1, 7 \leftrightarrow 8, 6 \leftrightarrow 3$. Now, middle part is swapped to obtain offsprings $o1 = (* * 5\ 1\ 8\ 3 * *)$ and $o2 = (* *1\ 4\ 7\ 6* *)$. Then, we can fill other bits (from respective parent) as $o1 = (* * 5\ 1\ 8\ 3\ 2 *)$ and $o2 = (* * 1\ 4\ 7\ 6\ 2 *)$. Now 6 comes in 1st position in $o1$ and 4 comes in 2nd position in $o1$ since $3(p2)$ is mapped to $6(p1)$ and $5(p2)$ is mapped to $1(p1)$ but 1 is already present in $o1$ therefore the mapping of $1(p2)$ is checked and the process is repeated until no conflict occurs which gives $4(p1)$ and similarly $o1 = (6\ 4\ 5\ 1\ 8\ 3\ 2\ 7)$ and $o2 = (5\ 3\ 1\ 4\ 7\ 6\ 2\ 8)$ are obtained.

3.4 Modified Cycle Crossover (CX2) [7]

This crossover operator is the improved version of basic cycle operator. The offsprings are generated using a cycle from the parents till the last bit. The bits

are arranged in circular order which results in less repetition. Thus, the offsprings inherit the maximum genes from the parents.

In this approach, initially, two parents are chosen. The 1st bit (say $k1$) of the second parent is selected as the 1st bit of the first offspring ($o1$). Now $k1$ in first parent is searched and same position bit (say $k2$) in second parent is selected. Again $k2$ is searched in $p1$ and same position bit (say $k3$) is selected as 1st bit of 2nd offspring ($o2$) Now $k3$ is searched in $p1$ and bit ($k4$) located at same position in $p2$ becomes next bit of $o1$. Thus, this process is continued until both the offsprings are filled by taking a single hop for first offspring and two hops for second offspring.

Example 4 Consider $p1 = (a\ b\ c\ d\ e\ f\ g\ h)$ and $p2 = (b\ g\ e\ h\ d\ a\ f\ c)$ be two parent chromosomes. In the process mentioned in [7], the offsprings are computed as $o1 = (b\ *\ *\ *\ *\ *\ *\ *)$, $o2 = (f\ *\ *\ *\ *\ *\ *\ *)$ $[b \rightarrow g$ and $g \rightarrow f]$, then $o1 = (b\ a\ *\ *\ *\ *\ *\ *)$ $[f \rightarrow a]$, $o2 = (f\ g\ *\ *\ *\ *\ *\ *)$ $[a \rightarrow b$ and $b \rightarrow g]$. Finally, the results are obtained as $o1 = (b\ a\ f\ g\ e\ c\ h\ d)$ and $o2 = (f\ g\ b\ a\ h\ d\ e\ c)$.

4 Experimental Result

In this section, we present our experimental observations implemented in Python 3.7. The various parameters and their values used for the experimentation purpose are: population size (P) = 50 for smaller dataset and 100 for larger dataset, generation count (G) = 500, crossover probability = 0.9, and mutation probability = 0.07. Here, we give an analysis of various results obtained while comparing four different crossover operators, which are RCMX, ERX, CX2, PMX on 10 benchmark datasets available in TSPLIB. Number of evaluations were set as 200 and results are shown in tables below. In Table 1, we show the average value of solution obtained after 200 iterations, whereas Table 2 shows the average time taken by each crossover.

Table 1 Average value generated by the respective crossover operator after 200 iteration (in km)

Dataset	RCMX	ERX	CX2	PMX
dantzig42	975.6	886.4	1146.0	1044.8
fri26	1082.1	997.2	1173.3	1171.9
gr17	2096.066	2088.467	2099.6	2128.867
attt48	41,998.3	41,735.1	42,645.033	42,710.1
bcl380	2956.46	3357.38	3599.85	3268.35
dj38	9016.6	8913.39	9190.65	9141.12
qa194	24,683.84	27,462.72	30,568.10	26,885.26
rw1621	40,146.24	45,249.34	48,895.19	43,916.67
nu3496	126,532.47	198,547.10	256,812.85	243,326.98
xqf131	830.92	1006.24	1280.94	1153.69

Table 2 Average time taken by the respective crossover after 200 iteration (in sec)

Dataset	RCMX	ERX	CX2	PMX
dantzig42	1.399	3.642	3.35	1.561
fri26	1.688	4.791	4.945	2.257
gr17	0.298	0.738	0.765	0.386
attt48	1.664	4.297	4.116	1.792
bcl380	21.353	46.859	33.079	23.096
dj38	3.10	9.905	9.322	4.215
qa194	21.648	41.65308	31.51	13.7901
rw1621	113.564	189.781	93.338	97.7901
nu3496	280.791	453.782	323.56	297.56
xqf131	8.783	20.081	17.371	7.538

We observe that quality of initial population makes an impact on the results. The quality of initial population can make the result more accurate and even sometimes gives optimal solution after few generations. In Table 1, we have shown average value of solution calculated over 200 iterations where we can observe that ERX performs better than other crossover operators for small size of datasets, for instance as we can see for dantzig42, fri26, gr17, att48, dj38 datasets, ERX performs better than the others, but for large datasets like ali535, bc380, xqf131, nu3496, RCMX performs better than ERX and other crossover operators, because ERX partially follow nearest neighbor which may work when there is small number of nodes, but for larger number of nodes, obtaining optimal solution is different if same concept is followed. Therefore, for small datasets, ERX is preferred and for large datasets, RCMX is preferred. Table 2 shows the average time taken by each crossover operator implemented on i7 7th gen processor and 8 GB RAM, where we find that RCMX is about 2 times faster than ERX in obtaining the solution. In Fig. 1, we have shown the graphical analysis of solutions obtained with respect to number of generations for each of the crossover operator for dataset xqf131.

5 Conclusion

In this chapter, we have compared four different crossover operators, i.e., RCMX, ERX, CX2, and PMX in the context of travelling salesman problem. Analyzing the behavior of these four crossover operators on ten different datasets, we have explored the applicability of various crossover operators on variable sized datasets. For comparative purpose, we have considered average value of solution and average time taken by each crossover operator over two hundred iterations. According to our comparative study, we conclude that RCMX outperforms other mentioned crossover operators in terms of performance and obtaining the best result for larger datasets

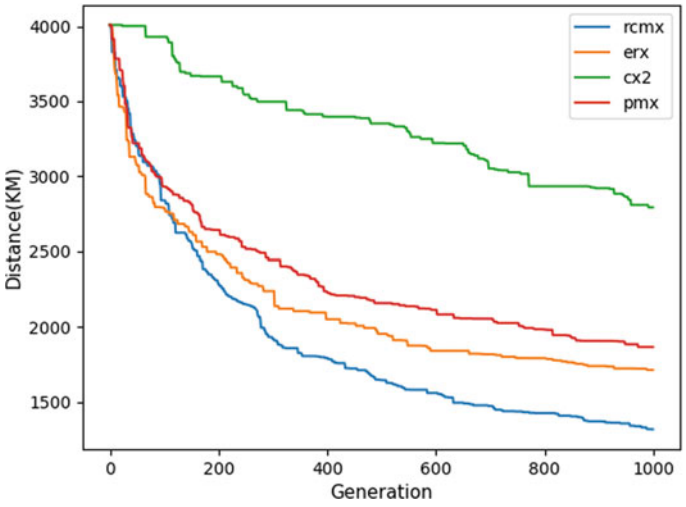


Fig. 1 Distance versus generation for dataset xqf131

whereas ERX perform better only for comparatively smaller sized datasets (less than hundred nodes). In the future, one can use the comparative analysis for developing new crossover operators using GA.

References

1. F.H. Khan, N. Khan, S. Tullah, S.T. Nizami, Solving TSP problem by using Genetic Algorithm. Int. J. Basic Appl. Sci. **9**(10), 79–88 (2009)
2. Z.H. Ahmed, Genetic algorithm for the TSP using sequential constructive crossover operator. Int. J. Bio-Metrics Bio-Inf. **3**(6), 96–105 (2010)
3. A. Philip, A.A. Taofiki, O. Kehinde, A genetic algorithm for solving travelling salesman problem. Int. J. Adv. Comput. Sci. Appl. **2**(1), 26–29 (2011)
4. O.M. Sallabi, An improved genetic algorithm to solve the travelling salesman problem. World Acad. Sci. Eng. Technol. **52**, 530–533 (2009)
5. G.A. JayaLakshmi, S. SathiaMoorthy, R. RajaRam, A hybrid genetic algorithm—a new approach to solve travelling salesman problem. Int. J. Comput. Eng. Sci. **2**(02), 339–355 (2001)
6. J. Xu, L. Pei, R-Z. Zhu, Application of a genetic algorithm with random crossover and dynamic mutation on the travelling salesman problem, in *Proceedings of 8th International Congress of Information and Communication Technology 2018*, pp. 937–945
7. A. Hussain, Y.S. Muhammad, N. Sajid, Genetic algorithm for the traveling salesman problem with the modified cycle crossover operator. Comput. Intell. Neurosci. **7430125** (2017)
8. S. Akter, N. Nahar, M.S. Hossain, K. Andersson, A new crossover technique to improve genetic algorithm and its application to TSP, in *Proceedings of International Conference on Electrical, Computer and Communication Engineering*, 7–9 Feb 2019
9. A. Otman, A. Jaafar, A comparative study of adaptive crossover operators for genetic algorithms to resolve the traveling salesman problem. Int. J. Comput. Appl. **31**(11), 0975–8887 (2011)

10. D. Whitley, T. Startweather and D'Ann Fuquay, Scheduling problems and traveling salesman: the genetic edge recombination operator, in *Proceedings of 3rd International conference on genetic algorithm*, pp. 133–140 (1989)

Modified Eigen Permutation-Based Model Simplification of LTI Systems Using Evolutionary Algorithm

Akhilesh Kr. Gupta, Chhabindra Nath Singh, Deepak Kumar, and Paulson Samuel

1 Introduction

The simplification of complex higher-order physical systems is necessary to comprehend its dynamic behaviour and to project a controller for various engineering applications. In this simplification process, the reduced order model (ROM) replicates the complex higher-order model (HOM) to ease the design and implementation of controllers and digital filters. For an effective model order reduction (MOR) technique, the essential requirements are (i) the minimization of approximation error (ii) preservation of properties such as stability of the HOM (iii) less computational effort. In general, the MOR techniques may be broadly categorized into time and frequency domains. Some conventional time domain techniques are aggregation method [1], balanced truncation [2], balanced singular perturbation approximation [3] and optimal Hankel norm approximation [4]. The frequency domain approach [5] may further be sub-classified as stability criterion method (SCM), stability preservation method (SPM) and classical reduction method (CRM). The conventional approximation theories are used for the development of CRM [6–8]. Although these methods provide computational simplicity however, in some cases, unstable ROMs may be

A. Kr. Gupta · D. Kumar · P. Samuel
Department of Electrical Engineering, MNNIT, Allahabad, Prayagraj 211004, India
e-mail: akhilesh_ree5213@mnnit.ac.in

D. Kumar
e-mail: deepak_kumar@mnnit.ac.in

P. Samuel
e-mail: paul@mnnit.ac.in

C. N. Singh (✉)
Department of Electrical Engineering, H.B. Technical University, Kanpur 208002, India
e-mail: cnsinghhbti7@gmail.com

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_5

produced even for a stable HOM. To deal the instability issue, SPM [9–14] is developed. The SCM employs a mixed approach [15–17] in which the denominator of ROM is obtained by one of the SPM and its numerator is determined by either CRM or optimization techniques. It is observed that the SCM improves the performance of ROMs in low-frequency ranges. The effectiveness of the optimization algorithms has already been ascertained for MOR problems as presented in [15–17]. Moreover, it is observed that these MOR methods save computational time and retains the original dynamic behaviour with high accuracy. Further, Singh and Chandra proposed various MOR techniques [13, 19–21]for interval systems. In this chapter, development of a new mixed MOR method is presented by combining MEP and BBBC optimization algorithm [18] by minimization of approximation error. Due to its fast convergence and simple operation, BBBC optimization algorithm is also used for the simplification of complex HOM [22–24].

The poles of ROM are synthesized by MEP approach which is developed for an rth order system by modification in the eigen permutation (EP) approach [25], whereas BBBC optimization is used to obtain the numerator coefficients of ROMs. The proposed scheme ensures the stability of ROM. This chapter is ordered into five sections: the problem statement is recited in Sect. 2 while Sect. 3 offers the proposed approach to compute the approximants. The effectiveness of the proposed approach is discussed in Sect. 4 by comparing the existing techniques [17, 24–31] with presented approach. Further, conclusive remarks are incorporated in Sect. 5.

2 Problem Formulation

For nth order complex LTI (MIMO or SISO) systems, the objective is to obtain an rth order ($r < n$) ROM which closely approximates the characteristics of the assumed HOM.

3 Proposed Approach

The proposed method generalizes the eigen permutation algorithm [25] to find the denominator of ROM while the BBBC optimization algorithm [18] is employed to figure the numerator polynomial.

3.1 Determination of ROM Denominator Polynomial

The proposed method generalizes the eigen permutation algorithm [25] to obtain the denominator of ROM. Singh et al. method [25] can only find a ROM of the second order; whereas by using the proposed generalization, it is possible to obtain a ROM

of any order. The proposed method is based on the dominant eigenvalue retention in ROM which is simple, effective and computer-oriented. Moreover, the proposed approach can also be extended to the approximation of HOM having complex roots. For a given HOM, the steps to find the ROM's denominator are summarized below:

Step 1 [Cluster formation] Compute the poles of HOM and form the clusters of poles, where real and complex poles of HOM are $-a_i, \quad i = 1, 2, \ldots, n_1$ and $X_m = -(\alpha_m \pm j\beta_m)$, $m = 1, 2, \ldots, n_2$, $n_1 + 2n_2 = n$, respectively.

Step 2 [Retention and Sorting] After retention of the dominant poles, sort all other poles in increasing manner such as $|a_1|<|a_2|<\cdots<|a_i|, i = 1, 2, \ldots n_1$, and $|\alpha_1|<|\alpha_2|<\cdots<|\alpha_m|, m = 1, 2, \ldots n_2$.

Step 3 [Repetition] Find the unique pole, i.e. a step is made to identify the repeated poles such that

Case 1: If the poles of HOM are repeated, then the poles of ROM are

(a) For real roots

$$a_{ey} = |a_1|, \quad y = 1, 2, \ldots, r_1 \tag{1}$$

(b) For complex roots

$$\alpha_{ez} = \alpha_1; \beta_{ez} = \frac{\sum_{t=1}^{m} (\beta_t)}{2\beta_m(z-1)}, \quad z = 2, 3, \ldots r_2 \tag{2}$$

where $-a_{ey}$, $y = 1, 2, \ldots, r_1$ and $R_r = -(\alpha_{ez} \pm j\beta_{ez})$, $z = 2, 3, \ldots, r_2$, $r_1 + r_2 = r$ are real and complex poles of the ROM, respectively.

Case 2: If the repeated real poles are not in existence, then the poles subsequent to retained pole (a_1) are calculated as:

For Real Poles

$$a_{ey} = \frac{\sum_{b=1}^{n} (a_b)}{a_n(y-1)}, \quad y = 2, 3, \ldots, r_1 \tag{3}$$

The complex poles (i.e. real and imaginary parts) of the ROM are determined by Eqs. (4) and (5) as below

$$\alpha_{ez} = \frac{\sum_{t=1}^{m} (\alpha_t)}{\alpha_m(z-1)}, \quad z = 2, 3, \ldots, r_2 \tag{4}$$

$$\beta_{ey} = \frac{\sum_{t=1}^{m} (\beta_t)}{2\beta_m(z-1)}, \quad z = 2, 3, \ldots, r_2 \tag{5}$$

where $r_1 + 2r_2 = r$.

Step 4 [Computation] The ROM denominator polynomial $\mathrm{p}_r(s)$ is figured as

$$p_r(s) = (s + ae_1) \cdots (s + aer_1)(s + (\alpha e_1 \pm j\beta e_1)) \cdots (s + (\alpha er_2 \pm j\beta er_2)) \quad (6)$$

3.2 Determination of ROM Numerator Polynomial

In this section, the BBBC optimization algorithm [18] is explained to determine the numerator polynomial of ROM. The flow chart of BBBC algorithm is shown in Fig. 1.

The BBBC operation is performed in two phases:

The big bang phase is followed by the big crunch phase. The first phase is the generation of an initial random population which is similar to the genetic algorithm

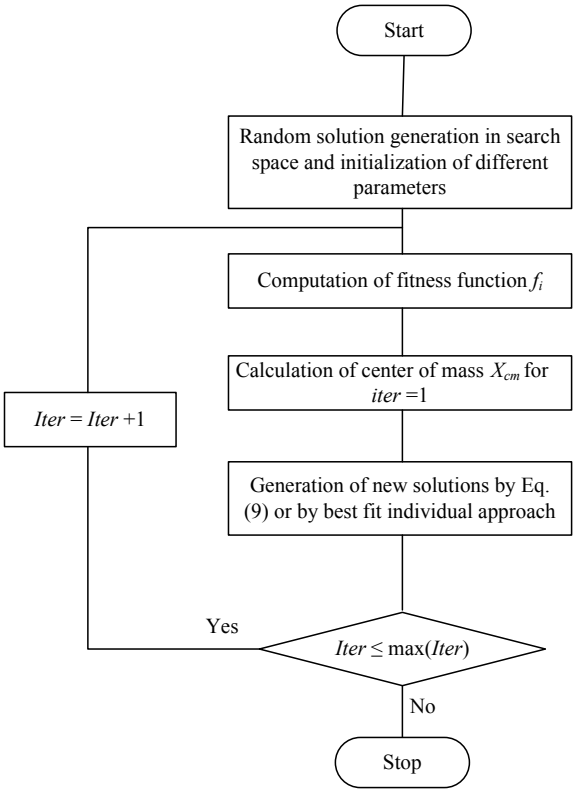


Fig. 1 Flow chart of BBBC algorithm

[15]. A contraction procedure of the random solutions is applied in the big crunch phase to create a centre of mass for the big bang phase until an optimization criterion is met.

The centre of mass (COM), X_{cm}, is calculated as

$$X_{\mathrm{cm}} = \frac{\sum_{i=1}^{K} \frac{X_i}{f_i}}{\sum_{i=1}^{K} \frac{1}{f_i}} \tag{7}$$

where f_i is the fitness function corresponding to point X_i in n-dimensional search space with population size of K. For the computation of ROM, the fitness function is considered as

$$\mathrm{f}_i = \sum_{\mathrm{i}=0}^{\mathrm{Q}} -1[\mathrm{h}(\mathrm{i}\Delta\mathrm{t}) - \mathrm{h_r}(\mathrm{i}\Delta\mathrm{t})]^2 \tag{8}$$

where $h(i\Delta t)$ and $h_r(i\Delta t)$ are the HOM and ROM unit step responses at $t = \Delta\, t$. The COM or best-fit individual is used as a centralized point for the next big bang phase to generate the random solutions. The new points around the COM are given by

$$\mathrm{X_i^{new}} = \mathrm{X_{cm}} + \frac{\mathrm{l_l} * \mathrm{rand}}{\mathrm{iter}} \tag{9}$$

where X_{cm} is the COM, l_l is the length of search space, and iter is the number of iteration with a random number rand.

4 Case Studies

Case 1: The transfer function of eighth order SISO LTI system [25, 29] is given as

$$T_8(s) = \frac{40320 + 185760s + 222088s^2 + 122664s^3 + 36380s^4 + 5982s^5 + 514s^6 + 18s^7}{40320 + 109584s + 118124s^2 + 67284s^3 + 22449s^4 + 4536s^5 + 546s^6 + 36s^7 + s^8}$$

By suggested approach, the ROM's denominator polynomial can be synthesized as $D_2(s) = 4.5 + 5.5s + s^2$. Similarly, by the procedure discussed in Sect. 3 and after DC gain matching, the ROM's numerator polynomial is found as $N_2(s) = 13.79s + 4.5$.

Finally, the TF of the approximant by proposed scheme is determined as $T_2(s) = \frac{13.79s+4.5}{s^2+5.5s+4.5}$.

Step response shows the transient as well as steady-state behaviour of the system. A step response plot of ROMs obtained by the proposed and other well-known approaches [17, 25] are shown in Fig. 2. The step responses of original system and ROM obtained by proposed approach are very close to each other in transient as well as steady-state behaviour. Therefore, it can be easily concluded that the proposed approach provides a ROM which shows a close resemblance in its behaviour to the original model.

Further, the performance attributes such as ISE, ITSE, IAE and ITAE are given in Table 1. These attributes are the representation of closeness of ROM behaviour to the original higher-order system. Further, less we get the magnitude of performance indices, better will be ROM. Table 1 shows that the proposed approach resulted lower values of the performance indices in comparison with other well-known approaches [17, 25, 27–29]. Therefore, the proposed approach is an efficient way to reduce the HOM into a ROM with retention of original system's characteristics.

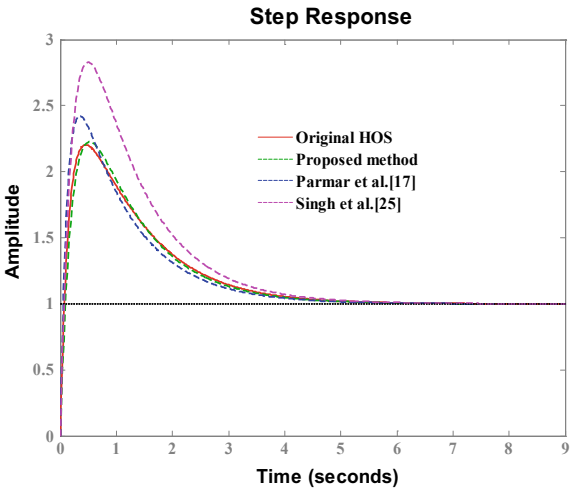


Fig. 2 Step responses of reduced models for Case 1

Table 1 Analysis of errors for Case 1

Methods	ISE	IAE	ITSE	ITAE
Proposed approach	9.689×10^{-3}	0.122	3.604×10^{-3}	0.147
Parmar et al. [17]	4.81×10^{-2}	0.3007	2.35×10^{-2}	0.389
Singh et al. [25]	1.008×10^{-2}	0.128	4.77×10^{-3}	0.138
Mittal et al. [27]	2.689×10^{-1}	0.8098	1.91×10^{-1}	1.210
Mukherjee and Mishra [28]	2.689×10^{-1}	0.8098	1.91×10^{-1}	1.210
Mukherjee et al. [29]	5.69×10^{-2}	0.4574	7.48×10^{-2}	0.948

Case 2: Let us deliberate a sixth order MIMO system from the existing articles [24, 30, 31]

By proposed algorithm, the approximant is computed as

$$T_2(s) = \frac{1}{s^2 + 3.05s + 2.05}\begin{bmatrix} 1.2169s + 2.05 & 0.8574s + 0.82 \\ 0.5395s + 1.025 & 1.5807s + 2.05 \end{bmatrix}$$

The step responses of ROMs evaluated by the proposed and prevailing approaches [24, 26] are depicted in Fig. 3(a–d), which clearly indicates the closeness of the proposed ROM with the HOM. From the step response, it is evident that the transient as well as steady-state behaviour of the proposed ROM is similar to the original HOM.

Further, the effectiveness of proposed approach is justified by the comparison of performance indices which is an important attribute to establish the closeness between HOM and ROM. A significant improvement in integral square error (ISE)

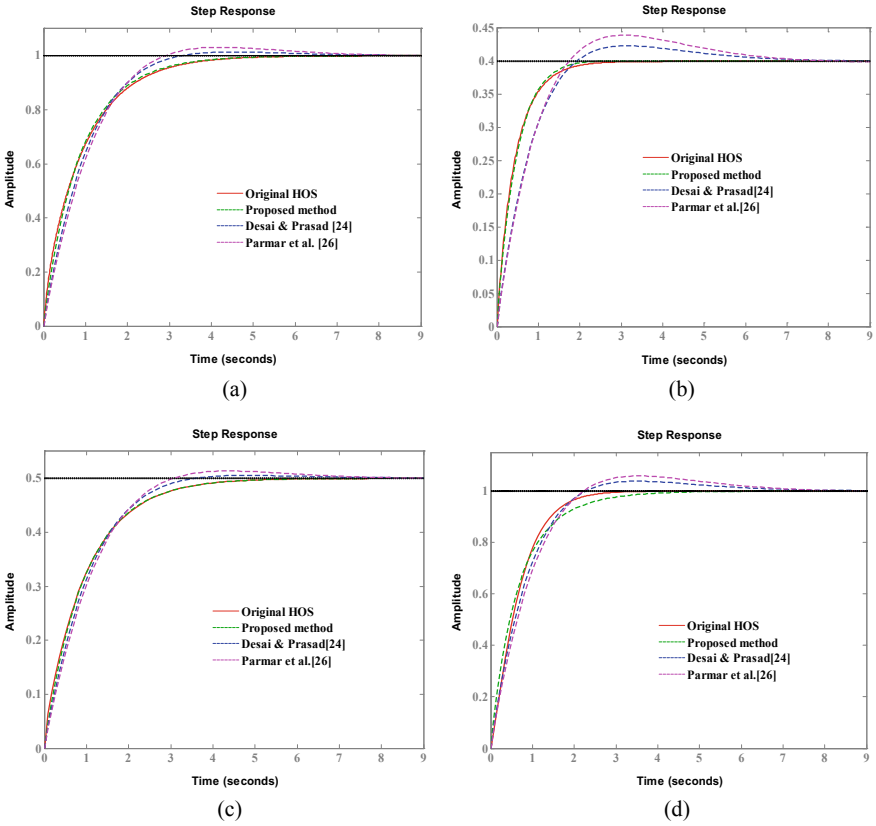


Fig. 3 **a–d** Step responses of ROMs for Case 2

Table 2 ISE comparison of ROMs for Case 2

Methods	R_{11} ($\times 10^{-3}$)	R_{12} ($\times 10^{-3}$)	R_{21} ($\times 10^{-3}$)	R_{22} ($\times 10^{-3}$)
Proposed approach	0.7696	0.07641	0.08607	3.72
Desai and Prasad [24]	7.026	6.031	1.024	6.367
Parmar et al. [26]	14.5	8.75	2.539	15.74
Prasad and Pal [30]	136.5	2.446	40.29	67.9
Prasad et al. [31]	30.69	2.56	262	21.68

as presented in Table 2 endorses the supremacy of proposed approach over prevailing techniques [24, 26, 30, 31]. Therefore, it is concluded that the proposed approach offers a better approximant in comparison with other well-known approaches. Further, the proposed model preserves the key characteristics of original HOM.

5 Conclusion

A simple and effective approach based on BBBC and modified eigen permutation is presented in this chapter for model reduction of HOM. The ROM's denominator is determined by the MEP, whereas the numerator polynomial is obtained by ISE minimization approach. The resulting ROM preserves the stability of HOM and a substantial improvement in the error indices like ISE, IAE, ITSE and ITAE are also observed. Further, the suggested technique can also be employed for the simplification of discrete time models.

References

1. M. Aoki, Control of large-scale dynamic systems by aggregation. IEEE Trans. Automat. Control **13**(3), 246–253 (1968)
2. B.C. Moore, Principal component analysis in linear systems: controllability, observability and model reduction. IEEE Trans. Automat. Control **26**(1), 17–32 (1981)
3. K.V. Fernando, H. Nicholson, Singular perturbation model reduction of balanced systems. IEEE Trans. Automat. Control **27**, 466–468 (1982)
4. K. Glover, All optimal Hankel norm approximation of linear multivariable systems and their L∞ L∞ error bounds. Int. J. Control **39**(6), 1115–1193 (1984)
5. D.K. Chaturvedi, *Modeling and Simulation of Systems Using MATLAB and Simulink* (CRC Press, 2010)
6. Y. Shamash, Stable reduced order models using Padè type approximation. IEEE Trans. Automat. Control **19**(5), 615–616 (1974)
7. Y. Shamash, Continued fraction methods for reduction of constant linear multivariable systems. Int. J. Syst. Sci. **7**, 743–758 (1976)
8. V.P. Singh, D.P.S. Chauhan, S.P. Singh, T. Prakash, On time moments and Markov parameters of continuous interval systems. J. Circ. Syst. Comput. **26**(3), 1750038 (2017)
9. Y. Shamash, Linear system reduction using Pade approximation to allow retention of dominant modes. Int. J. Control **21**(2), 257–272 (1975)

10. R.K. Appiah, Padè methods of Hurwitz polynomial approximation with application to linear system reduction. Int. J. Control **29**(1), 39–48 (1979)
11. T.C. Chen, C.Y. Chang, K.W. Han, Stable reduced-order Padè approximants using stability equation method. Electron. Lett. **16**(9), 345–346 (1980)
12. V.P. Singh, P. Chaubey, D. Chandra, Model order reduction of continuous time systems using pole clustering and Chebyshev polynomials, *in IEEE Students Conference on Engineering and Systems (SCES)* (Allahabad, India, 2012)
13. V.P. Singh, D. Chandra, Routh-approximation based model reduction using series expansion of interval systems, in *IEEE International Conference on Power, Control and Embedded Systems (ICPCES)* (Allahabad, India, 2010). https://doi.org/10.1109/ICPCES.2010.5698658
14. T.N. Lucas, Factor division: a useful algorithm in model reduction. IEE Proc. **130**(6) (1983)
15. Satakshi, S. Mukherjee, R. C. Mittal, Order reduction of linear discrete systems using a genetic algorithm. Appl. Math. Model. **29**, 565–578 (2005)
16. Z. Gallehdari, M. Karrari, O.P. Malik, Model order reduction using PSO algorithm and its application to power systems, in *Proceedings of IEEE International Conference on Electric Power and Energy Conversion Systems* (Sharjah, 2009), pp. 1–5
17. G. Parmar, R. Prasad, S. Mukherjee, System reduction using factor division algorithm and eigen spectrum analysis. Appl. Math. Model. **31**(11), 2542–2552 (2007)
18. O.K. Erol, Eksin: new optimization method: big bang-big crunch. Adv. Eng. Softw. **37**(2), 106–111 (2006)
19. V.P. Singh, D. Chandra, Model reduction of discrete interval system using clustering of poles. Int. J. Model. Identification Control **17**(2), 116–123 (2012)
20. V.P. Singh, D. Chandra, Analysis and adjustment of steady state response of rational systems associated with interval systems, in *IEEE Students Conference on Engineering and Systems (SCES)* (Allahabad, India, 2012)
21. V.P. Singh, D. Chandra, Model reduction of discrete interval system using dominant poles retention and direct series expansion method, in *IEEE International power engineering and optimization conference (PEOCO)* (Malaysia, 2011)
22. B. Philip, J. Pal, An evolutionary computation based approach for reduced order modeling of linear systems, in *Proceedings of IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)* (Coimbatore, India, 2010), pp. 1–8
23. S.R. Desai, R. Prasad, A new approach to order reduction using stability equation and big bang big crunch optimization. Syst. Sci. Control Eng. **1**(1), 20–27 (2013)
24. S.R. Desai, R. Prasad, A novel order diminution of LTI systems using big bang big crunch optimization and Routh approximation. Appl. Math. Model. **37**, 8016–8028 (2013)
25. J. Singh, K. Chatterjee, C.B. Vishwakarma, System reduction by eigen permutation algorithm and improved Pade approximations. Int. J. Math. Comput. Phys. Electr. Comput. Eng. **8**(1) (2014)
26. G. Parmar, R. Prasad, S. Mukherjee, Order reduction of linear dynamic systems using stability equation method and GA. Int. J. Electr. Comput. Energetic Electron. Commun. Eng. **1**(2), 26–32 (2007)
27. A.K. Mittal, R. Prasad, S.P. Sharma, Reduction of linear dynamic systems using an error minimization technique. Int. J. Electr. Comput. Energetic Electron. Commun. Eng. **1**(2), 201–206 (2007)
28. S. Mukherjee, R.N. Mishra, Order reduction of linear systems using an error minimization technique. J. Franklin Inst. **323**(1), 23–32 (1987)
29. S. Mukherjee, Satakshi, R. C. Mittal, Model order reduction using response matching technique. J. Franklin Inst. **342**, 503–519 (2005)
30. R. Prasad, J. Pal, Use of continued fraction expansion for stable reduction of linear multivariable systems. J. Inst. Eng. India IE (I) J. EL **72**, 43–47 (1991)
31. R. Prasad, J. Pal, A.K. Pant, Multivariable system approximation using polynomial derivatives. J. Inst. Eng. India IE J. EL **76**, 186–188 (1995)

A Mixed Approach for Model Reduction Using Differential Evolution and Eigen Permutation

Chhabindra Nath Singh, Akhilesh Kr. Gupta, Deepak Kumar, and Paulson Samuel

1 Introduction

Mathematical modeling of a system is essential to understand its dynamic behavior and to design a controller for closed-loop operation. Modern controllers such as H∞ and LQG possess the same order or even higher order than the order of plant. Therefore, the implementation of such controllers poses a great challenge for closed-loop systems. Hence, a ROM which provides identical characteristics to the original system is desired. Despite the availability of many methods, no approach provides the best result for all problems. Some conventional frequency domain methods are continued fraction expansion [1], moment matching [2] and Padè approximation [3]. However, these methods face the issue of instability for some cases. Therefore, to overcome this issue, stability guaranteed methods [4–6] are employed. In addition to this, several combinatorial approaches [7–10] are presented by integrating the attributes of stability guaranteed methods and conventional methods for the synthesis of a ROM. Some evolutionary algorithms [11–14] are also used for model order reduction (MOR) problems in many articles [15–20] by minimizing ISE. Recently, fuzzy C-means clustering-based methods [21, 22] as well as factor division with

C. N. Singh (✉)
Department of Electrical Engineering, H.B. Technical University, Kanpur, U.P. 208002, India
e-mail: cnsinghhbti7@gmail.com

A. Kr. Gupta · D. Kumar · P. Samuel
Department of Electrical Engineering, MNNIT, Allahabad, Prayagraj, U.P. 211004, India
e-mail: akhilesh_ree5213@mnnit.ac.in

D. Kumar
e-mail: deepak_kumar@mnnit.ac.in

P. Samuel
e-mail: paul@mnnit.ac.in

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_6

improved pole clustering-based mixed approaches [23] are presented for MOR of continuous and discrete-time systems. Further, various MOR techniques [35–38] are presented in the recent past for the simplification of interval systems.

It is observed that still there is a large scope for effective improvement in reduction error between the higher-order system (HOS) and ROM. Therefore, a novel mixed approach is presented for MOR of LTI continuous-time systems by combining EP [24] and DE algorithm [25] which always preserves the stability of HOS in ROM. The proposed approach is computationally simple and computer-oriented. Further, the approach proposed in this chapter is compared with existing approaches [16, 18–20, 24, 26–34] to substantiate its efficacy. Other sections of this chapter are organized in the following order. The problem statement is narrated in Sect. 2, and the proposed method is described in Sect. 3. To show the supremacy of the proposed method over other approaches, case studies are considered in Sect. 4. The conclusion of the present work is incorporated in Sect. 5.

2 Problem Statement

For the given nth order LTI SISO and MIMO systems expressed in the transfer function (TF) form, the objective is to acquire a rth order ($r < n$) reduced model which closely approximates the key characteristics of the given HOS.

3 Proposed Method

Since last few decades, an extensive research has been carried out to approximate the order of HOS by merging two methods. In this chapter, the attributes of the two techniques are integrated to further minimize the approximation errors. The proposed method is described as follows:

3.1 Computation of Denominator Polynomial

The dominant eigenvalues retention-based EP method was proposed by Singh et al. [24]. This approach also determines the imaginary roots for ROM. For a given *nth* order system with real poles $-R_i, i = 1, 2, \ldots, n_1$ and complex poles $C_j = -(x_j \pm jy_j),\ j = 1, 2, \ldots, n_2$ where $n_1 + n_2 = n$, the following steps are followed to find the poles of ROM:

Step 1 Arrange the real poles and real parts of the complex poles lying in the left half s-plane as $|R_1| < |R_2| < \cdots < |R_i|, i = 1, 2, \ldots, n_1$ and $|x_1| < |x_2| < \cdots < |x_j|, j = 1, 2, \ldots, n_2$.

Step 2 For the real poles, retain the most dominant pole and sort the other non-dominant poles in ascending order. Check the recurrence of real poles. In case of repetition, both poles of ROM are equal to the dominant pole. If it is not so, then the pole other than dominant pole is evaluated as $a_{e2} = \frac{1}{R_{n_1}}\left[\sum_{k=1}^{n_1} R_k\right]$.

Step 3 Similarly, check the repetition of real parts of the complex poles. If these are repetitive, then the real parts of ROM poles are given as. If it is not so, then the real and imaginary parts of the poles of ROM are evaluated by the formula given below

$$x_{e1} = \frac{1}{\mathrm{Re}(C_{n_2})}\left[\sum_{k=1}^{n_2} \mathrm{Re}(C_k)\right]; \quad y_{e1} = \frac{1}{2\mathrm{Im}(C_{n_2})}\left[\sum_{k=1}^{n_2} \mathrm{Im}(C_k)\right].$$

Step 4 The denominator polynomial of ROM is computed as

$$p_r(s) = \begin{cases} (s + a_{e1})(s + a_{e2}), & \text{for real poles} \\ (s + x_{e1} \pm j y_{e1}), & \text{for complex poles} \end{cases} \tag{1}$$

where a_{e1} is the retained pole.

3.2 Computation of Numerator Polynomial

Once the denominator of ROM is determined, the numerator polynomial is obtained by DE using the minimization of fitness function which is discussed as below:

Differential Evolution
Storn [25] presented DE in which the weighted differences between randomly generated solutions are utilized to change the population. The DE algorithm is operated by three operations: mutation, crossover and selection. The implementation of this algorithm consists of the following steps:

a. **Initialization**
The initial parameters are randomly generated with upper and lower bound values. Let $x_{i,j}^k$ be a solution among N number of candidates (i.e.$i = 1, 2, \ldots, N$) of D design variables (i.e. $j = 1, 2, \ldots, D$) at kth iteration which is represented by

$$x_{i,j}^k = x_L + \mathrm{rand}(0,\ 1)^*(x_U - x_L) \tag{2}$$

where rand(0, 1) are the uniformly distributed random numbers between 0 and 1. Each variable j in a i^{th} solution candidate is bounded between the lower bound x_L and upper bound x_U.

b. **Mutation**
The mutation is the addition of a weighted difference of two randomly selected vectors to a third randomly selected vector. These three randomly selected vectors $x_{r_1,k}$, $x_{r_2,k}$ and $x_{r_3,k}$ are chosen in a manner such that their indices r_1, r_2 and $r_3 (r_1, r_2, r_3 \in [1, N])$ are distinct. The weighted addition of two vectors generates a third vector v_i^{k+1} called mutant vector or donor vector as $v_i^{k+1} = x_{r_1}^k + F\left(x_{r_2}^k - x_{r_3}^k\right)$, $F \in [0, 2]$.
where $0.5 < F < 1$ is the mutation factor.

c. **Crossover**
The crossover is a recombination process which is performed by combining the mutant vector v_i^{k+1} with the target vector $x_{i,j}^k$, and the result comes in the form of a new vector called trial vector $u_{i,j}^{k+1}$. This trial vector is selected in the crossover operation by the expression given below

$$u_{i,j}^{k+1} = \begin{cases} v_{i,j}^{k+1} \text{ if } \text{rand}_{i,j} \leq \text{CR or } j = I_{\text{rand}} \\ x_{i,j}^k \text{ if } \text{rand}_{i,j} > \text{CR or } j \neq I_{\text{rand}} \end{cases} \tag{3}$$

where $i = 1, 2, 3, \ldots, N$; $j = 1, 2, 3, \ldots, D$; $I_{\text{rand}} = 1, 2, \ldots, D$ is a random integer and $CR \in (0, 1)$ is crossover probability.

d. n**Selection**
The target and trial vectors are promoted to the next generation after a comparison on the basis of the better fitness function value. The selection procedure is represented by

$$x_i^{k+1} = \begin{cases} u_i^{k+1} & \text{if} \quad f\left(u_i^{k+1}\right) \leq f\left(x_i^k\right) \\ x_i^k & \text{otherwise} \end{cases} \tag{4}$$

e. **Termination**
If the algorithm reaches to stopping criterion, the optimum solution is displayed; otherwise, it goes to step (b) to step (d) again.

4 Numerical Examples

Case 1: Let us consider an eighth order SISO LTI system from the existing literature [24, 32].

The poles of the HOS are determined as

$$a_1 = -1, a_2 = -2, a_3 = -3, a_4 = -4, a_5 = -5, a_6 = -6, a_7 = -7, a_8 = -8.$$

By recommended method, the poles of ROM are found as $a_{e1} = -1, a_{e2} = -4.50$.

Hence, the ROM denominator polynomial is worked out as $D_2(s) = s^2 + 5.5s + 4.5$.

By DE algorithm as described in Sect. 3.2, and matching of DC gain, the numerator polynomial of the approximant is evaluated as $N_2(s) = 13.8490s + 4.5$.

Eventually, the ROM of the second order is found as $R_2(s) = \frac{13.8490s+4.5}{s^2+5.5s+4.5}$.

Figures 1 and 2, respectively, show the time and frequency responses of eighth order HOS and ROM acquired by proposed and existing approaches [24, 32].The comparison of performance indices with existing methods [20, 24, 30–32] is presented in Table 1. From this table, it is perceived that there is a noteworthy improvement in the error indices for the suggested technique over prevailing methods [20, 24, 30–32].

Case 2: To investigate the proficiency of the recommended scheme, now consider a sixth-order MIMO system from the existing literature [34]

By employing the proposed algorithm, the poles of the second-order ROM are calculated as

$$a_{e1} = -1; \quad a_{e2} = -2.05.$$

Finally, the proposed ROM in the TF form is determined as

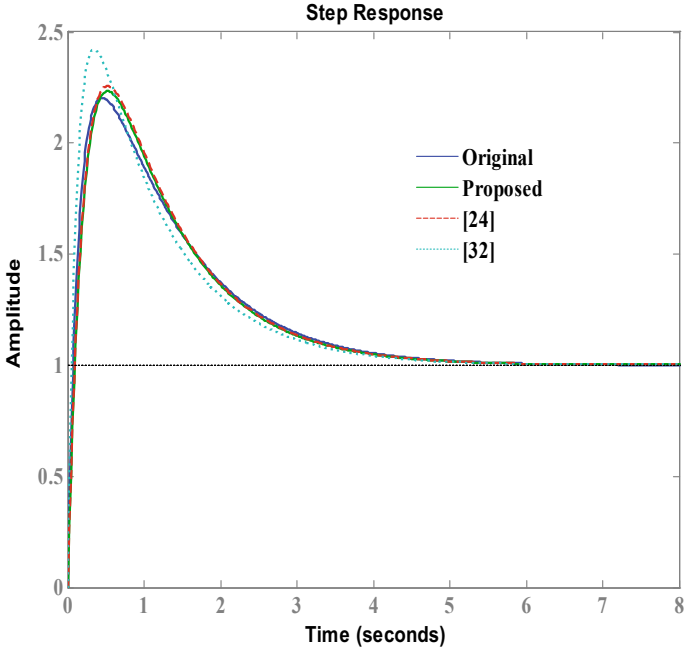


Fig. 1 Step responses of the HOS and ROM

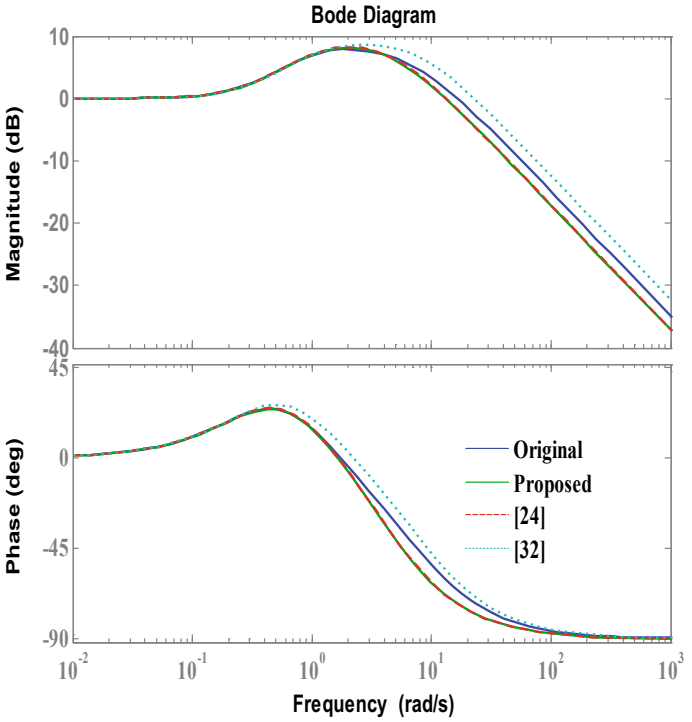


Fig. 2 Frequency responses of HOS and ROM

Table 1 Comparison of performance indices for Case 1

Reduction method	ISE	ITSE	IAE	ITAE
Proposed method	9.6016×10^{-3}	3.7661×10^{-3}	0.122	0.143
Gupta et al. [20]	9.628×10^{-3}	3.96×10^{-3}	0.1236	0.141
Singh et al. [24]	10.08×10^{-3}	4.77×10^{-3}	0.128	0.138
Parmar et al. [32]	48.1×10^{-3}	23.5×10^{-3}	0.3007	0.389
Mukherjee et al. [31]	56.9×10^{-3}	74.8×10^{-3}	0.4574	0.948
Mittal et al. [30]	268.9×10^{-3}	191×10^{-3}	0.8098	1.210

$$R_2(s) = \frac{1}{s^2 + 3.05s + 2.05} \begin{bmatrix} 1.224s + 2.05 & 0.8597s + 0.82 \\ 0.5550s + 1.025 & 1.5885s + 2.05 \end{bmatrix}$$

Figure 3(a–d) depicts the step responses of sixth-order MIMO system and second-order models computed by the proposed and prevailing methodologies [16, 18, 34]. It is detected that the responses of the proposed ROM replicate the response of the HOS better than [16, 18, 34]. Further, a considerable improvement in the ISE (refer

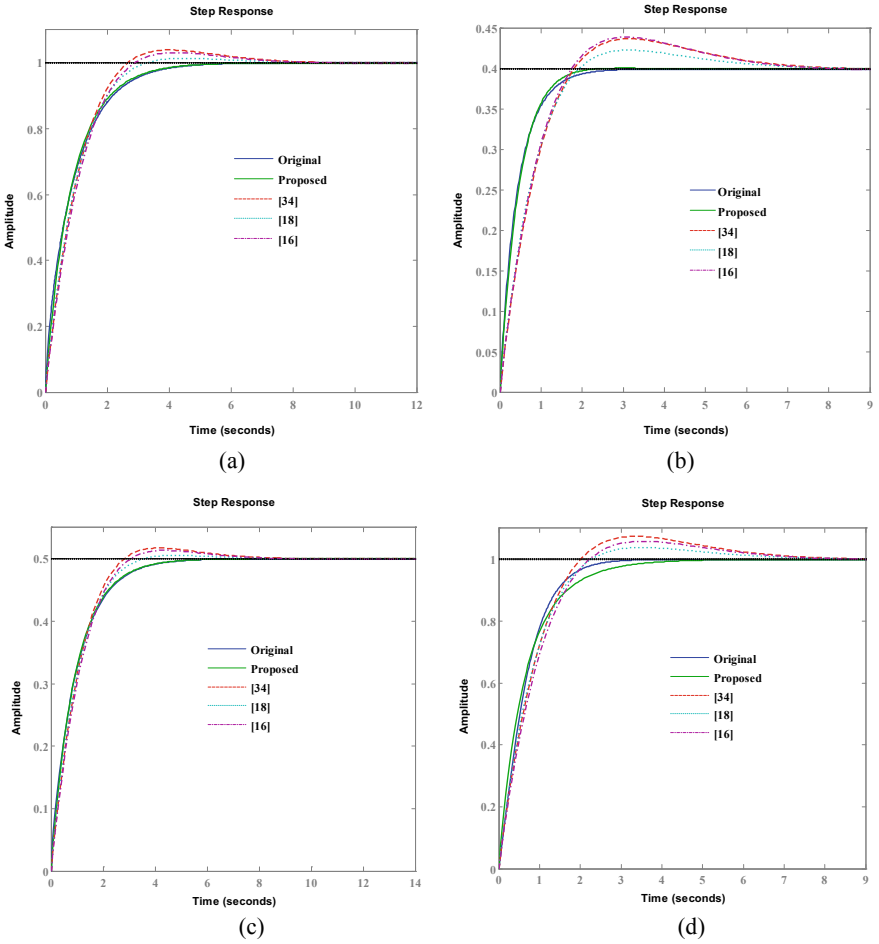


Fig. 3 (**a–d**) Comparison of step responses for the HOS and ROMs for Case 2

Table 2 Comparison of ISE for ROMs of Case 2

Reduction techniques	R_{11}	R_{12}	R_{21}	R_{22}
Proposed approach	7.7318×10^{-4}	7.6903×10^{-5}	7.1226×10^{-3}	3.6×10^{-3}
Gupta et al. [20]	7.732×10^{-4}	7.690×10^{-5}	7.123×10^{-5}	3.55×10^{-3}
Parmar et al. [16]	1.45×10^{-2}	8.75×10^{-3}	2.539×10^{-3}	1.574×10^{-2}
Desai and Prasad [18]	7.026×10^{-3}	6.031×10^{-3}	1.024×10^{-3}	6.367×10^{-3}
Narwal and Prasad [34]	1.62×10^{-2}	8.98×10^{-3}	2.97×10^{-3}	1.71×10^{-2}
Prasad and Pal [28]	1.365×10^{-1}	2.446×10^{-3}	4.029×10^{-2}	6.79×10^{-2}

Table 2) endorses the superiority of developed method over existing methods [16, 18, 28, 34].

5 Conclusion

A novel mixed method based on eigen permutation (EP) and differential evolution (DE) is presented for the approximation of higher-order continuous-time LTI systems. The denominator coefficients of ROM are evaluated by EP, whereas its numerator polynomial is obtained by DE using ISE minimization. The performance indices are also included to ascertain the superiority of the proposed method over prevailing approaches. The proposed method also promises the stability of approximants if the original HOS is stable. Further, the methodology suggested in this chapter can also be extended for the simplification of LTI interval systems.

References

1. Y. Shamash, Continued fraction methods for the reduction of constant-linear multivariable systems. Int. J. Syst. Sci. **7**(7), 743–758 (1976)
2. V. Zakian, Simplification of linear time-invariant systems by moment approximants. Int. J. Control. **18**(3), 455–460 (1973)
3. Y. Shamash, Stable reduced-order models using Padé-type approximations. IEEE Trans. Autom. Control. **19**(5), 615–616 (1974)
4. M.F. Hutton, B. Friedland, Routh approximations for reducing order of linear, time-invariant systems. IEEE Trans. Autom. Control. AC **20**(3), 329–337 (1975)
5. R.K. Appiah, Linear model reduction using Hurwitz polynomial approximation. Int. J. Control. **28**(3), 477–488 (1978)
6. T.C. Chen, C.Y. Chang, K.W. Han, Stable reduced-order Padè approximants using stability equation method. Electron. Lett. **16**(9), 345–346 (1980)
7. J. Pal, System reduction by a mixed method. IEEE Trans. Autom. Control. AC **25**(5), 973–976 (1980)
8. V. Singh, D. Chandra, H. Kar, Improved Routh-Padé approximants: a computer-aided approach. IEEE Trans. Autom. Control. **49**(2), 292–296 (2004)
9. V. Singh, Obtaining Routh-Padé approximants using the Luus-Jaakola algorithm. IEE Proc. Control Theo. Appl. **152**(2), 129–132 (2005)
10. S. Panda, S.K. Tomar, R. Prasad, C. Ardil, Model reduction of linear systems by conventional and evolutionary techniques. Int. J. Electr. Comput. Energ. Electron. Commun. Eng. **3**(11), 2150–2156 (2009)
11. D.E. Goldberg, *Genetic Algorithms* (Pearson Education, India, 2006)
12. Y. Zhang, A comprehensive survey on particle swarm optimization algorithm and its applications. Math. Probl. Eng. 931256 (2015)
13. O.K. Erol, E. Ibrahim, A new optimization method: big bang-big crunch. Advances in Eng. Soft. **37**(2), 106–111 (2006)
14. A.B. Mohamad, A.M. Zain, N.E.N. Bazin, Cuckoo search algorithm for optimization problems-a literature review and its applications. Appl. Artif. Intel. **28**, 419–448 (2014)
15. Satakshi, S. Mukherjee, R.C. Mittal, Order reduction of linear discrete systems using a genetic algorithm. Appl. Math. Model. **29**(6), 565–578 (2005)

16. G. Parmar, R. Prasad, S. Mukherjee, Order reduction of linear dynamic systems using stability equation method and GA. Int. J. Electr. Comput. Energ. Electron. Commun. Eng. **1**(2), 236–242 (2007)
17. Z. Gallehdari, M. Karrari, O.P. Malik, MOR using PSO algorithm and its application to power systems. IEEE Int. Conf. EPECS09, 1–5 (2009)
18. S.R. Desai, R. Prasad, A novel order diminution of LTI systems using big bang-big crunch optimization and Routh Approximation. Appl. Math. Model. **37**(16–17), 8016–8028 (2013)
19. S.R. Desai, R. Prasad, A new approach to order reduction using stability equation and big bang-big crunch optimization. Syst. Sci. Control Eng. **1**(1), 20–27 (2013)
20. A.K. Gupta, D. Kumar, P. Samuel, A meta-heuristic cuckoo search and Eigen permutation approach for model order reduction. Sadhana, 1–11(2018)
21. A. Narain, D. Chandra, R.K. Singh, Model order reduction using fuzzy C-means clustering. Trans. Inst. Meas. Control, 1–7 (2014)
22. R.K. Gautam, N. Singh, N.K. Choudhary, A. Narain, MOR using factor division algorithm and fuzzy C-means clustering technique. Trans. Inst. Meas. Control, 1–8 (2018)
23. C.N. Singh, D. Kumar, P. Samuel, Improved pole clustering-based LTI system reduction using a factor division algorithm. Int. J. Model. Simul. **39**(1), 1–13 (2019)
24. J. Singh, K. Chatterjee, C.B. Vishwakarma, System reduction by Eigen permutation algorithm and improved Padé approximations. Int. J. Math. Comput. Phys. Electr. Comput. Eng. **8**(1), 180–184 (2014)
25. R. Storn, On the usage of differential evolution for function optimization. IEEE Proc. North Am. Fuzzy Inf. Process, 519–523 (1996)
26. J. Pal, Stable reduced-order Padé approximants using the Routh-Hurwitz array. Electron. Lett. **15**(8), 225–226 (1979)
27. S. Mukherjee, R.N. Mishra, Order reduction of linear systems using an error minimization technique. J. Frankl. Inst. **323**(1), 23–32 (1987)
28. R. Prasad, J. Pal, Use of continued fraction expansion for stable reduction of linear multivariable systems. J. Inst. Eng. Electr. Eng. Div. **72**, 43–47 (1991)
29. R. Prasad, J. Pal, A.K. Pant, Multivariable system approximation using polynomial derivatives. J. Inst. Eng. Electr. Eng. Div. **76**, 186–188 (1995)
30. A.K. Mittal, R. Prasad, S.P. Sharma, Reduction of linear dynamic systems using an error minimization technique. J. Inst. Eng. Electr. Eng. Div. **84**(2), 201–206 (2004)
31. S. Mukherjee, Satakshi, R. C. Mittal.: Model order reduction using response matching technique. J. Frankl. Inst. **342**(5), 503–519 (2005)
32. G. Parmar, S. Mukherjee, R. Prasad, System reduction using factor division algorithm and Eigen spectrum analysis. Appl. Math. Model. **31**(11), 2542–2552 (2007)
33. B. Philip, J. Pal, An evolutionary computation based approach for reduced-order modelling of linear systems. IEEE ICCIC, 28–29 (2010)
34. A. Narwal, R. Prasad, A novel order reduction approach for LTI systems using cuckoo search optimization and stability equation. IETE J. Res. **62**(2), 1–10 (2015)
35. V.P. Singh, D. Chandra, Model reduction of discrete interval system using clustering of poles. Int. J. Model. Identification Control **17**(2), 116–123 (2012)
36. V.P. Singh, D. Chandra, Analysis and adjustment of steady state response of rational systems associated with interval systems, in *IEEE Students Conference on Engineering and Systems (SCES)* (Allahabad, India, 2012)
37. V.P. Singh, D. Chandra, Model reduction of discrete interval system using dominant poles retention and direct series expansion method, in *IEEE International power Engineering and Optimization Conference (PEOCO)* (Malaysia, 2011)
38. V.P. Singh, D.P.S. Chauhan, S.P. Singh, T. Prakash, On time moments and Markov parameters of continuous interval systems. J. Circ. Syst. Comput. **26**(3) (2017)

Simulation Studies of Inverted Pendulum Based on VSC Using Whale Optimization Algorithm

Satyendra Kumar and Moina Ajmeri

1 Introduction

The nonlinear and unstable systems are frequently used in scientific laboratories for implementing and validating new ideas emerging in control engineering. The inverted pendulum is an open-loop unstable having nonlinear dynamics. It is also a under-actuated mechanical system [1]. Hence, inverted pendulum is known as one of the most important problems in control engineering and it is frequently used [2–4]. Also it has been studied excessively in the control literature. It is used as a test bed for the control system laboratories, where it is generally taken to describe various issues in control communities [4, 5]. Thus, the inverted pendulums are used as classical tools in the control system laboratories to overcome the challenges of the control. In the control engineering, the control of inverted pendulum system can be classified into three parts. The first case is the swing up control of the cart-inverted pendulum, where it is used to swing the pendulum shaft till the position finds stabilization region [6–8]. The second case is stabilization of the cart-inverted pendulum[9, 10]. The third case is the pendulum tracking control, which is used to control the position of pendulum to follow a fixed path [11, 12]. The models of the x-inverted pendulum are analyzed in detail. Various controllers have been proposed in previously for balancing and tracking of cart-inverted pendulum.

In this work, we aim to design a VSC having robustness and good responses to parameters variations or external disturbances. Generally, it is often difficult to determine the optimal or near optimal parameters of the recent controller design. These parameters can be determined from mathematical methods (or analytical methods,

S. Kumar (✉) · M. Ajmeri
Department of Electrical Engineering, National Institute of Technology, Ashok Rajpath, Patna 800005, India

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_7

e.g., simson's 1/3 rule, etc.) and nature-inspired metaheuristic optimization techniques such as particle swarm optimization (PSO), ant colony optimization (ACO), genetic algorithm (GA), whale optimization algorithm (WOA), etc. The nature-inspired optimization techniques are more popular nowaday because they can be used in a variety of problems from different disciplines.

Therefore, a recent metaheuristic algorithm known as "Whale Optimization Algorithm (WOA)" [13] is used in the controller design to get m the optimal parameters. There is a need of performance index in order to get the desired results. There are various available performance criteria such as IE, IAE, ITAE, ISE, ITSE, ISTE, etc. In this algorithm, ITAE is used as objective function. To find the optimal VSC parameters, there is need to minimize the performance criterion IAE.

2 Model and Structure of the Cart-Inverted Pendulum

The cart-inverted pendulum is fitted on a movable cart which is controlled by horizontal force as shown in Fig. 1. The control action is based on the horizontal displacements of cart.

The state equation of the cart-inverted pendulum may be written as [14]

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \frac{-mg\cos x_3 \sin x_3 + ml \sin x_3 x_4^2 + F_x}{M + m\sin^2 x_3} + d_1 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = \frac{-ml\cos x_3 \sin x_3 x_4^2 - \cos x_3 F_x + (M+m)\sin x_3}{Ml + ml\sin^2 x_3} + d_2 \end{cases} \tag{1}$$

where $x_1 = x, x_2 = \dot{x}, x_3 = \theta, x_4 = \dot{\theta}$ and d_1, d_2 are outer disturbances,θ is the pendulum angle from vertical, x is the position of cart, Remaining parameters are enlisted in Table 1.

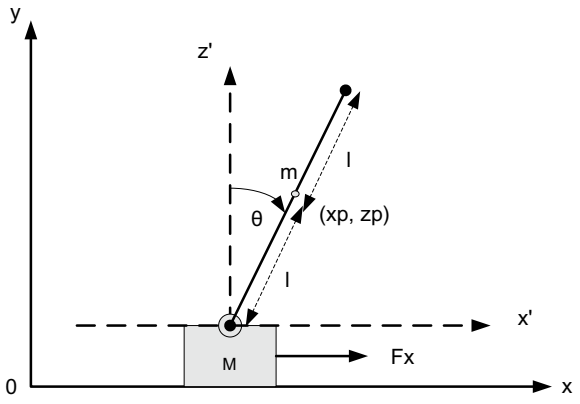


Fig. 1 Cart-inverted pendulum

Table 1 Various parameters of the pendulum

Mass of the cart M (kg)	Pendulum mass m (kg)	Pendulum length l (m)	Gravitational acceleration g (m/s^2)
1	0.1	0.3	9.8

3 Whale Optimization Algorithm (WAO)

Nature-inspired optimization algorithms becoming popular in the engineering applications day by day because their simple concepts are easy for implementation. Also, the algorithms may bypass local optima. Nature-inspired optimization algorithms solve optimization engineering problems by using biological behavior (e.g., swarm based) or physical phenomena. For example: Particle swarm optimization (PSO) [15], genetic algorithm (GA) [16], simulated annealing (SA) [17], etc. The whale optimization algorithm [13] is nature-inspired optimization technique. The algorithm has been tested for 23 benchmark functions and it is found that it performs better in comparison of the other meta-heuristic algorithms. This optimization technique is proposed by S. Mirjalily and Lewis in 2016.

The algorithm mimics humpback whale behavior [13]. The humpback whales hunt prey near the surface of sea water using spiral-shaped path and may trap the prey in a net of bubbles. It has two phases:

(1) *Exploitation phase*, which refers catching a prey using spiral bubble-net by the humpback whale.
(2) *Exploration phase*, random search for prey.

The mathematical explanation of the two phases are explained in the reference [13]

There are five unknown parameters in the proposed VSC controller namely, $k_i, k_p, \lambda_1, \lambda_2$ and k_e. These parameters can be determined (approximately) using analytical method, but in this way, one cannot find the optimal setting of the parameters. Hence, there needed modern optimization techniques which includes various conventional optimization techniques such as "Nelder-Mead simplex algorithm," Simpson's rule, "Genetic Algorithm (GA)," etc. In this work, a novel use of recent nature-inspired optimization algorithm called "whale optimization algorithm (WOA)" is carried out for determining the optimal controller parameters. In order to get the optimal parameter values, the *performance index* is need. As there are various performance index (i.e., IE, IAE, ITAE, ISE, etc.) are available. Here, ITAE is taken as the objective function (for minimization) in the MATLAB codes of WOA. The ITAE criterion is given by

$$\text{ITAE} = \int_0^\infty t|e(t)|\mathrm{d}t \tag{2}$$

4 Variable Structure Controller Design for the Inverted Pendulum

. In VSC, first of all, a sliding surface is defined which represents the desired behavior of the closed loop system. Further, a control law is designed such that it keeps the trajectories of the system on the predefined surface. Here, the sliding surface (swithching surface) is chosen as [10],

$$\sigma = \lambda_1 e + \lambda_2 \dot{e} \tag{3}$$

where $e = r - y$, is tracking error with $y(t)$ and $r(t)$ are outputand reference (or set-point), respectively, and $\lambda_1 (> 0)$, λ_2 are tunig parameter which determine system performance on the sliding surface. In this work, a new VSC is designed to bring the trajectories of the error on the surface and to maintain on it. The VSC is as follows [10]:

$$\begin{cases} u = -y + k_p |\sigma|^{1/2} \text{sat}(\sigma) + u_1 + k_e e \\ \text{where } \dot{u}_1 = k_i |\sigma|^{1/2} \text{sat}(\sigma) \end{cases} \tag{4}$$

where k_p, k_e, k_i are the gains, $|u_1| \leq c_3$ for some $c_3 > 0$, y (i.e. θ or x) is the system output, u(orF_x) is the control signal and σ is the sliding surface and sat (.) is a saturation function which is given by

$$\text{sat}(\sigma) = \begin{cases} \sigma/|\sigma| \text{ or } \text{sgn}(\sigma), & \text{if} |\sigma| \geq 1 \\ \sigma & \text{if} |\sigma| < 1 \end{cases} \tag{5}$$

4.1 The First Step with VSC Controller Only to Control the Angle of the Cart-Inverted Pendulum

In this case, we design a VSC controller as given in Eq. (4) to control the angle of the cart-inverted pendulum. Here, we do not consider the cart position control. The controller design of the cart-inverted pendulum with VSC structure is shown in Fig. 2. The initial angle is taken to be 0.5 rad and the pendulum parameters are listed in Table 1. The five parameters of the VSC controller are obtained by the "Whale optimization algorithm(WOA)" [13], as given blow.

$$k_p = -19.0946, k_i = -17.2315, k_e = -12.1937,$$
$$\lambda_1 = 2.5261 \text{ and } \lambda_2 = 0.5125$$

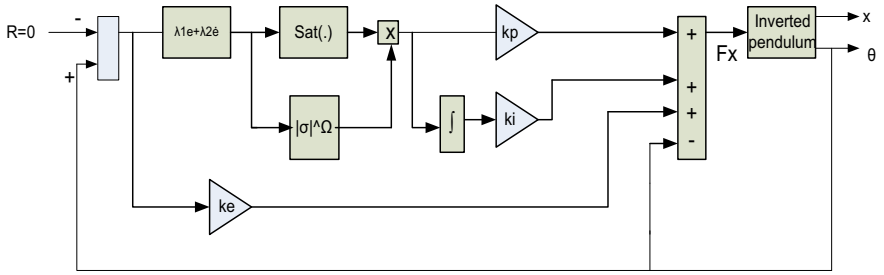


Fig. 2 VSC controller design for x inverted pendulum for angle control

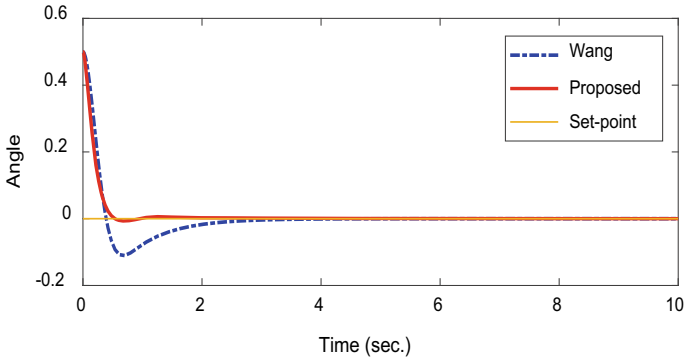


Fig. 3 Angle of the pendulum

The parameters of the PID controller (J. J. Wang) are given by

$$k_p = 25, k_i = 15 \text{ and } k_d = 3$$

A comparative response for the variation of angle of cart-inverted pendulum and control signal using proposed controller and that given by J. J. Wang [14] is shown in Figs. 3 and 4, respectively. From Fig. 3, it is clear that the proposed controller gives satisfactory response in terms of settling time, IAE, and undershoot.

4.2 The Second Step with VSC Design and a PID Controller to Control the Position of Cart-Inverted Pendulum

In this case, the VSC controller with a PID controller (as given by J. J. Wang, i.e., PID1) is designed as shown in Fig. 5. For controlling the position of pivot (cart), the parameters of PID controller are as follows

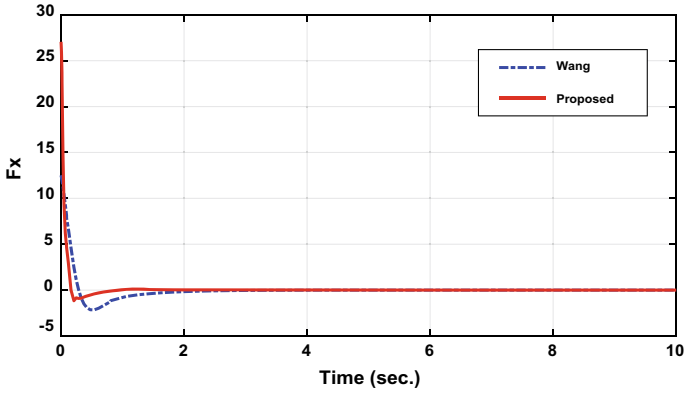


Fig. 4 Control signal the pendulum

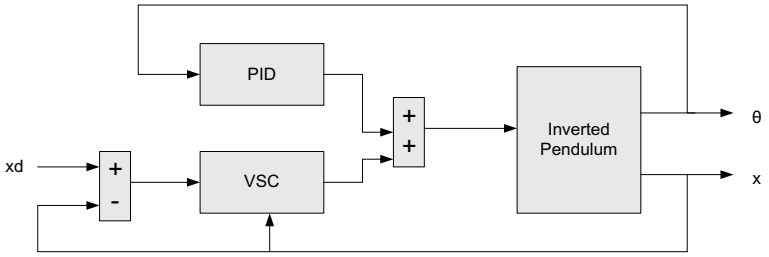


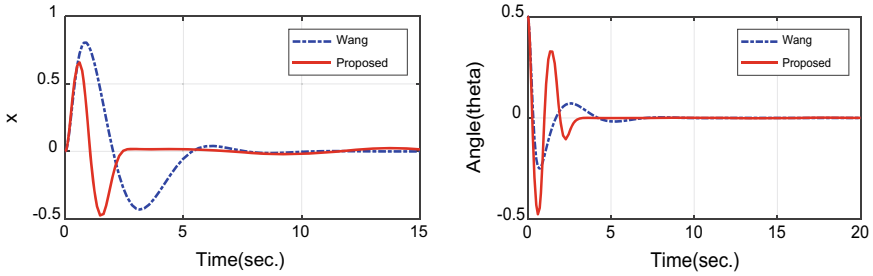
Fig. 5 VSC- and PID-based controller design for position and angle control

$$k_p = 25, k_i = 15 \text{ and } k_d = 3$$

Furthermore the parameters of the VSC controller are obtained by the "Whale optimization algorithm (WOA)," as given below.

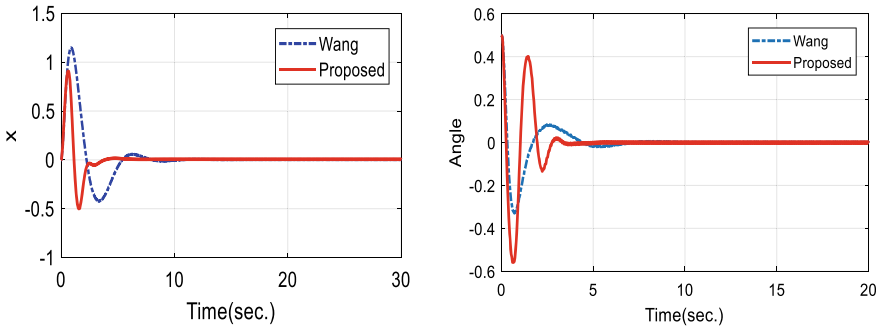
$$k_p = -3.269, k_i = -4.0625, k_e = -0.05, \lambda_1 = 1.3899 \text{ and } \lambda_2 = 1.0$$

Response for the position of pivot and angle of the art-inverted pendulum using proposed controller and that given by J. J. Wang [14] is shown in Fig. 6. From Fig. 6, it is clear that the proposed controller gives better response in term of the settling time, IAE, and undershoot. Further, to check the disturbance rejection capabilities, the outer disturbances d_1 and d_2 in Eq. (4) are considered and $d_1 = d_2 = 20 \sin(20\pi t)$ is taken. The simulation results with these disturbances are shown in Fig. 7. Figure 8 shows the control signals with and without disturbances. Simulation results show that the proposed controller with PID can stabilize the cart-inverted pendulum with good robustness compared to that proposed by the J. J. Wang.



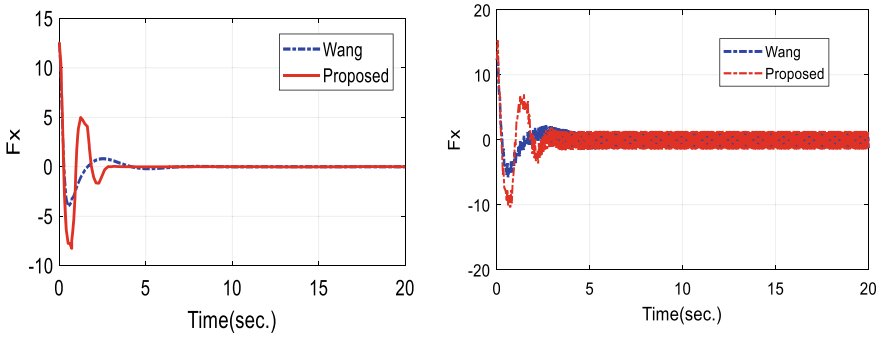
(a) X-position of inverted pendulum (b) Angleof cart-inverted pendulum

Fig. 6 Position and angle of the cart- inverted pendulum



(a) x position of the pendulum with outer disturbances (b) Angle of the pendulum with outer disturbances

Fig. 7 Position and angle of cart- inverted pendulum with outer disturbances



(a) Control signal without disturbance (b) Control signal with disturbance

Fig. 8 Control signals

5 Conclusion

In this chapter, a variable structure control (VSC) system is designed for the stabilization of the cart-inverted pendulum. All the parameters of the controller are obtained using a recent optimization technique known as "Whale optimization algorithm (WOA)." From the simulation results, we conclude that the proposed controller gives better performance to stabilize the cart-inverted pendulum in the upright direction. The proposed controller has good robustness toward outer disturbances. The proposed controller design may be further applied to various unstable and nonlinear processes such that a good robustness and disturbance rejection capabilities may be achieved.

References

1. M. Bugeja, *Non-Linear Swing-Up and Stabilizing Control of an Inverted Pendulum System* (IEEE, EUROCON, 2003), pp. 437–441
2. K.F.K.J. Astrom, Swinging up a pendulum by energy control. Autometica **36**, 287–295 (2000)
3. B. Hemalatha, A. Kanchana, Stability control of inverted pendulum based on fractional order PID controller. IJARET **4**, 170–175 (2017)
4. P. Chalupa, Modelling and predictive control of inverted pendulum. ECMS. https://doi.org/10.7148/2008-0531 (2014)
5. X. Ruan, Y. Wang, The modelling and control of flywheel inverted pendulum system. Int. Conf. Comput. Sci. Inf. Technol. **6**, 423–427. https://doi.org/10.1109/ICCSIT.2010.5563931
6. P. Mason, M.E. Broucke, B. Piccoli, Time Optimal Swing-Up of the Planar Pendulum, pp. 1–21 (2007)
7. J.S. Tao, C.W. Taur, T.W. Hsieh, C.L. Tsai, Design of a fuzzy controller with fuzzy swing-up and parallel distributed pole assignment schemes for an inverted pendulum and cart system. IEEE Trans. Control Syst. Technol. **16**, 1277–1288 (2008)
8. T. Glück, A. Eder, A. Kugi, Automatica Swing-up control of a triple pendulum on a cart with experimental validation. Automatica **49**, 801–808 (2013). https://doi.org/10.1016/j.automatica.2012.12.006
9. J. Yi, N. Yubazaki, Stabilization fuzzy control of inverted pendulum systems. Artif. Intel. Eng. **14**, 153–163 (2000)
10. G. Ablay, Variable structure controllers for unstable processes. J. Process Control **32**, 10–15 (2015). https://doi.org/10.1016/j.jprocont.2015.04.017
11. R. Wai, S. Member, L. Chang, Adaptive Stabilizing and tracking control for a nonlinear inverted-pendulum system via sliding mode technique. IEEE Trans. Ind. Electron. **53**, 674–692 (2006)
12. L. Chang, A. Lee, Design of nonlinear controller for bi-axial inverted pendulum system. IET Control Theory App. **1**, 979–986 (2007). https://doi.org/10.1049/iet-cta
13. S. Mirjalili, A. Lewis, The whale optimization algorithm. Adv. Eng. Softw. **95**, 51–67 (2016). https://doi.org/10.1016/j.advengsoft.2016.01.008
14. J.-J. Wang, Simulation studies of inverted pendulum based on PID controllers. Simul. Model. Pract. Theory **19**, 440–449 (2011). https://doi.org/10.1016/j.simpat.2010.08.003
15. J.H. Holland, Genetic algorithm. Sci. Am. Inc. **267**, 66–73 (1992)
16. S. Kirkpatrick, C.D. Gelatt, Optimization by simulated annealing. Sci. JS. **220** 671–680 (1983)
17. J. Kennedy, R. Eberhart, Particle swarm optimization. IEEE, 1942–1948 (1995)

Vision-Based Fault Detection and Parameter Identification for Flight Control Law Reconfiguration

Kishan S. Chowhan, Vijay V. Patel, Hemendra Arya, and Girish S. Deodhare

1 Introduction

The current research trend in flight control system is to improve the safety and to reduce the risk under critical failures. To mitigate the risk under sensor, actuator, and/or flight control system failure, redundancy is built-in and reliability has been improved a lot. The fact that reliable cum accurate detection and identification of fault [1] in any system is a crucial step even before trying to find a solution cannot be ignored. It becomes much more important when we are dealing with safety critical systems such as flight control which if left undetected can have hazardous consequences. In this chapter, we specifically cover the fault detection and identification due to battle damage of vertical tail of a high performance fighter aircraft. Compared to an industrial system, the flight control system of a fighter aircraft faces a number of challenges trying to mitigate failures and upsets. Modern aircrafts are characterized by highly reliable components and flight software. Even though individual sensors and control effectors are highly reliable by design, modern aircrafts are equipped with redundant control system and at-least double or triple redundant flight computers.

K. S. Chowhan (✉) · H. Arya
Indian Institute of Technology Bombay, Mumbai 400076, India
e-mail: kishan@ada.drdo.in

K. S. Chowhan · V. V. Patel · G. S. Deodhare
Aeronautical Development Agency, Bangalore 560093, India

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_8

2 Fault Type—Vertical Tail Damage

Most of the time, the fault in an aircraft is attributed to three types of failures, i.e., actuator fault, sensor fault, or damage to aircraft structure. Of the three, the structural damage is the least researched topic in today's scenario just because it is simply not possible to risk a huge infrastructure like aircraft for damage testing to collect data; however, it is all the most essential to do so. Due to failure-induced aerodynamics and fight dynamics change, the tolerant control for structural damage is found to be far more challenging then any other fault on aircraft. Structural damage for vertical tail loss is reported in Refs. [2–4].It may be noted that any damage to shape of aircraft geometric shape will also modify its aerodynamic characteristics which are represent by stability and control derivatives. The safety of the aircraft is also impaired due to abrupt deterioration of handling qualities and flight performance. In general, the static and dynamic stability get degraded due to vertical tail damage affecting the directional characteristics of aircraft. Vertical tail damage can also result in a reduction in control effectiveness of rudder, either directly from partial loss of rudder area, or indirectly from disrupted airflow over the surface caused by upstream damage on the vertical tail. The capability of aircraft to land in crosswind or ability to compensate for engine thrust asymmetries is badly affected due to loss of rudder control power [5].

3 Vision-Based Fault Detection—Methodology

The focus is toward reliable and accurate detection of damage so as to perform control reconfiguration as soon as possible with minimal overhead on computational requirements. A vision-based damage (fault) detection and identification for vertical tail of a fighter aircraft is proposed in this chapter and subsequent change in aircraft dynamics is discussed. A pin hole camera mounted on either side of the wing is planned to continuously capture the image of vertical tail of aircraft during flight. An edge detection algorithm discussed later in Sect. 3.1 is continuously monitoring the image received from camera. As long as there is no change in structure of vertical tail, no faults will be triggered. However, when there is a change in the structure of vertical tail due to damage, a fault is triggered and the image captured is further processed to recognize/identify it as fault, assess damage and trigger appropriate control laws for reconfiguration.

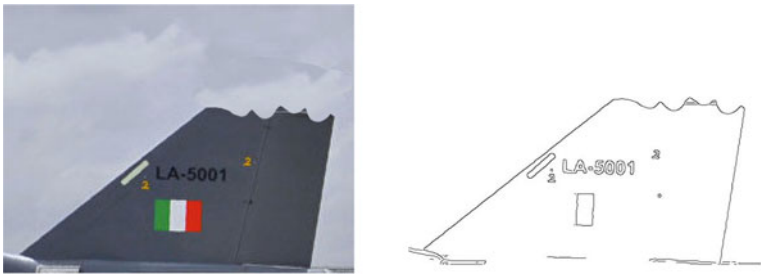
3.1 Edge Detection Algorithm

The edge detection algorithm works on the principle of reducing the amount of data in an image and performs the image processing while preserving the structural properties.

Though several algorithms exists, this chapter focuses on a particular one developed by John F. Canny (JFC) in 1986 [6]. From the edge detection methods developed till date, Canny's edge detection is the most strictly defined method that provides accurate and reliable detection. It uses an algorithm having multiple stages to detect a wide range of edges in images. The Canny's edge detection algorithm runs in five separate steps [7]:

1. **Smoothing**: Blurring of the image to remove noise and unwanted details and textures (using Gaussian filter).
2. **Finding gradients**: Gradients of the image with large magnitude should have their edges marked.
3. **Non-maximum suppression**: The spurious response to edge detection is eliminated using non-maximum suppression.
4. **Double thresholding**: Theresholding technique is used to determine potential edges.
5. **Edge tracking by hysteresis**: All the edges that are not connected to very certain edge are suppressed in order to arrive at final edge.

In this chapter, a Canny's edge detection program in MATLAB is used [8]. A vertical tail image of fighter aircraft with a span wise loss of 25% area is captured in Fig. 1a and image is passed through Canny's edge detection algorithm. The edge detected output is shown in Fig. 1b.



(a) Actual Image (b) Edge Detected Output

Fig. 1 Edge detection for 25% loss of vertical tail

4 Off-Line Fault Database Generation

An off-line fault data base is generated for comparison with real-time captured image and subsequent fault reconfiguration is performed with changed parameters of state matrix. The changed parameters of state matrix are obtained based on estimated stability and control derivatives. Before estimating the derivatives, it is required to identify the change in area of vertical tail in terms of standard parameters such as surface area (S_v), Span (b_v), location of aerodynamic center, i.e., horizontal distance L_v and vertical distance Z_v. For obtaining the same, we first gather the basic dimensions of vertical tail of aircraft which is a root chord C_R, tip chord C_T, sweep angle of leading edge, distance of leading edge from aircraft vertical datum, i.e., reference point in front of aircraft, and the Fuselage Reference Line (FRL).

4.1 Geometric Method

Geometric method is a novel and simple method thought by authors to arrive at the area of vertical tail after damage using traditional mathematical geometry. This method not only gives the closest approximation of the surface area but also accurately determines the location of change in aerodynamic center which was neglected in Hitachi's degree of damage method [9]. The geometric method can be further divided into two subgroups namely symmetric damage and asymmetric damage. In symmetric damage, the damage model is obtained by artificially generating damage by removing certain area along the tail height starting from the tip. It can be either 10, 25, or 50%, and so on. Here, we divide the shape approximately in to known geometries to arrive at an approximate area and corresponding aerodynamic center. In asymmetric damage case, the damage to the tail leads to an abrupt shape which will involve little more effort in terms on number of mathematical geometries to be approximated to estimate the remaining surface after damage and may also need additional methods such as parameter identification to verify the data.

In this chapter, the focus is on symmetric subgroup and an attempt is made to estimate the area of surface available and further estimate the changed stability and control derivative for the same. A simple geometric method is used to arrive at surface area and span for 25% damaged vertical tail as shown in Fig. 2, where C_R is root chord, C_T is tip chord, surface area is S and span of vertical tail is given by b_V.

Here, readers are requested to note that, the slant angle for trailing edge of vertical tail is assumed to be close to 90° for ease of implementing geometric method. The sweep angle of vertical tail as shown in Fig. 2 is 50°. Using geometric method:

$$\tan 40^\circ = \frac{3b_V/4}{y} = \frac{b_V/4}{x} \tag{1}$$

From Fig. 2, $y = 3x$, and root chord can be represented by

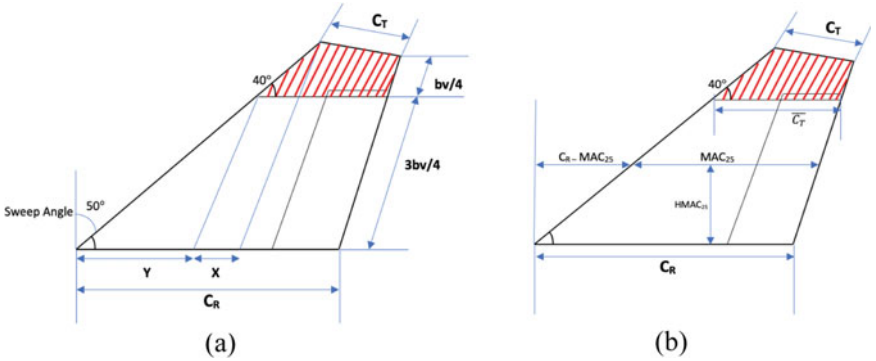


Fig. 2 **a** Surface area calculation for 25% loss of vertical tail, **b** MAC calculation for 25% loss of vertical tail

$$C_R = C_T + y + x. \tag{2}$$

Therefore, the new tip chord due to 25% loss of vertical tail from tip is given by $\overline{C}_T$.

where

$$\overline{C}_T = x + C_T. \tag{3}$$

Hence, the change in surface area of vertical tail for 25% loss is given by

$$\overline{S}_{V25} = 0.5 * y * \frac{3b_V}{4} + (x + C_T) * \frac{3b_V}{4}. \tag{4}$$

Since with change is surface area, there will be a change in aerodynamic center, the same can be calculated again with simple geometric method as shown in Figs. 2 and 3. The mean aerodynamic chord (MAC) for 25% loss of vertical is obtained by.

$$\text{MAC}_{25} = \frac{2}{3} * C_R * \left[\frac{1 + \lambda + \lambda^2}{1 + \lambda}\right]. \tag{5}$$

where λ is the taper ratio of vertical tail.

The height of MAC_{25} above C_R can be obtained by geometric measurement of vertical tail. Distance of leading edge of MAC from the leading edge of root chord C_R is obtained by subtracting MAC_{25} from root chord and can be denoted by LE_{25}.Since the aerodynamic center is located at quarter chord of MAC from leading edge; hence, the position of same can now be obtained as shown in Fig. 2b. In Fig. 3, a is the distance of leading edge of vertical tail from leading edge of mean aerodynamic chord (MAC_{25}), b is distance of aerodynamic center of vertical tail from leading edge of MAC_{25}, c is the vertical distance of aerodynamic center of vertical tail from

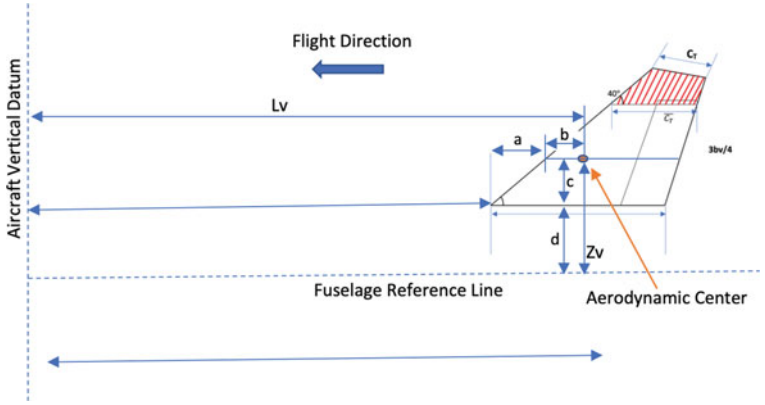


Fig. 3 Aerodynamic center calculation for 25% loss of vertical tail

Table 1 Changed vertical tail parameters for different loss percentage

Loss % of VT	S'_V	b'_V	L'_V	Z'_V
25	3.16	1.299	11.03	1.47
50	2.80	1.531	11.22	0.98
75	1.29	1.704	11.37	0.1875

root chord C_R and d is the vertical distance of C_R from Fuselage Reference Line (FRL) of aircraft.

Using the above geometric method, the change in surface area, span, and location of aerodynamic center for given percentage of loss of vertical tail is calculated and shown in Table 1.

5 Mathematical Modelling of Damaged Aircraft

The mathematical model of aircraft with partially damaged vertical tail can be divided in two sections, namely estimation of aerodynamic derivatives and parametrization of state matrix.

5.1 Estimation of Aerodynamic Derivatives

Wind tunnel experiments conducted by NASA confirm that changes of the stability derivatives are approximately proportional to the percentage of loss in structure [10]. In general, the vertical tail damage/loss will have a significant impact on lateral and

directional dynamic behaviors. Let Δ denote the deviation of derivatives caused by the damage. In correspondence, the set of lateral directional derivative deviations of interest is represented by:

Stability derivatives: $\Delta C_{y_\beta}, \Delta C_{n_\beta}, \Delta C_{l_\beta}, \Delta C_{y_p}, \Delta C_{n_p}, \Delta C_{l_p}, \Delta C_{y_r}, \Delta C_{n_r}, \Delta C_{l_r}$.

Control derivatives: $\Delta C_{y_{\delta r}}, \Delta C_{l_{\delta r}}, \Delta C_{n_{\delta r}}$.

The work of Hitachi [9] and Roskam [11] is adopted to give an estimated representation of the derivative deviations with several mild assumptions. However, unlike Hitachi [9], the change in aerodynamic center is not considered as negligible and is actually computed using geometric method as described in previous section. The only assumption made is that the mass loss due to the vertical tail damage is negligible compared to the stability derivative deviations which anyway has no role in analytical estimation of derivatives. Let the subscript 'V' in $C_{y_\beta V}$ denotes the portion of the vertical tail contribution to C_{y_β}.

$$C_{y_\beta V} = -kC_{L_\alpha V}\left(1 + \frac{\mathrm{d}\sigma}{\mathrm{d}\beta}\right)\eta_V \frac{S_V}{S} \tag{6}$$

Readers are requested to refer to Bandu [12] for detailed calculations. The ΔC_{y_β} caused by the vertical tail loss is, therefore, equal to $\Delta C_{y_\beta V}$ if any other parts of airframe are free from the effects of the damage. The derivatives $C_{l_\beta V}$ and $C_{n_\beta V}$ are estimated [11] by Eqs. (7) and (8) as shown below.

$$C_{l_\beta V} = C_{y_\beta V}\left(\frac{z_v}{b}\cos\alpha - \frac{l_v}{b}\sin\alpha\right). \tag{7}$$

$$C_{n_\beta V} = -C_{y_\beta V}\left(\frac{l_v}{b}\cos\alpha + \frac{z_v}{b}\sin\alpha\right). \tag{8}$$

where α is angle of attack, b is wing span, l_v and z_v are horizontal and vertical location of the aerodynamic center of the vertical tail. When the vertical tail is partially damaged, the deviation in $C_{l_\beta V}$ and $C_{n_\beta V}$ can be expressed in terms of Δ (defined earlier) as follows:

$$\Delta C_{l_\beta V} = \Delta C_{y_\beta V}\left\{\frac{(z_v + \Delta z_v)}{b}\cos\alpha - \frac{(l_v + \Delta l_v)}{b}\sin\alpha\right\}. \tag{9}$$

$$\Delta C_{n_\beta V} = \Delta C_{y_\beta V}\left(\frac{(l_v + \Delta l_v)}{b}\cos\alpha + \frac{(z_v + \Delta l_v)}{b}\sin\alpha\right) \tag{10}$$

Similarly, for $\Delta C_{y_p V}$, $\Delta C_{l_p V}$ and $\Delta C_{n_p V}$

$$\Delta C_{y_p V} = 2\Delta C_{y_\beta V}\left(\frac{(z_v + \Delta z_v)}{b}\cos\alpha - \frac{(l_v + \Delta l_v)}{b}\sin\alpha\right). \tag{11}$$

$$\Delta C_{l_p V} = 2\Delta C_{y_\beta V}\left(\frac{(z_v + \Delta z_v)^2}{b}\right). \tag{12}$$

$$\Delta C_{n_p V} = -\frac{2}{b^2}\Delta C_{y_\beta V}((l_v + \Delta l_v)\cos\alpha + (z_v + \Delta z_v)\sin\alpha)$$
$$((z_v + \Delta z_v)\cos\alpha - (l_v + \Delta l_v)\sin\alpha) \tag{13}$$

Now, for the rth derivative, $\Delta C_{y_r V}$, $\Delta C_{l_r V}$ and $\Delta C_{n_r V}$

$$\Delta C_{y_r V} = \frac{-2}{b}\Delta C_{y_\beta V}((l_v + \Delta l_v)\cos\alpha + (z_v + \Delta z_v)\sin\alpha). \tag{14}$$

$$\Delta C_{l_r V} = \frac{-2}{b^2}\Delta C_{y_\beta V}((l_v + \Delta l_v)\cos\alpha + (z_v + \Delta z_v)\sin\alpha)$$
$$((z_v + \Delta z_v)\cos\alpha - (l_v + \Delta l_v)b\sin\alpha) \tag{15}$$

$$\Delta C_{n_r V} = \frac{2}{b^2}\Delta C_{y_\beta V}((l_v + \Delta l_v)\cos\alpha + (z_v + \Delta z_v)\sin\alpha)^2. \tag{16}$$

From Eqs. (6–16) above, a set of post-failure stability derivatives C' is estimated as follows:

$$\begin{bmatrix} C'_{y_\beta} & C'_{y_p} & C^{y_p}_{y_r} \\ C^{y_r}_{l_\beta} & C'_{l_p} & C'_{l_r} \\ C'_{n_\beta} & C'_{n_p} & C'_{n_r} \end{bmatrix} = \begin{bmatrix} C_{y_\beta} - \Delta C_{y_\beta} & C_{y_p} - \Delta C_{y_p} & C_{y_r} - \Delta C_{y_r} \\ C_{l_\beta} - \Delta C_{l_\beta} & C_{l_p} - \Delta C_{l_p} & C_{l_r} - \Delta C_{l_r} \\ C_{n_\beta} - \Delta C_{n_\beta} & C_{n_p} - \Delta C_{n_p} & C_{n_r} - \Delta C_{n_r} \end{bmatrix} \tag{17}$$

The equations for control derivatives are also obtained from Roskam method [11] and are given below for ready reference:

$$C_{y_{\delta r} V} = C_{L_\alpha V}\left[\frac{(\alpha_\delta)_{C_L}}{(\alpha_s)_{C_L}}\right](\alpha_\delta)_{C_L} k' K_b \frac{S_V}{S}. \tag{18}$$

$$C_{l_{\delta r} V} = C_{y_{\delta r} V}\frac{(z_v \cos\alpha - l_v \sin\alpha)}{b}. \tag{19}$$

$$C_{n_{\delta r} V} = -C_{y_{\delta r} V}\frac{(l_v \cos\alpha - z_v \sin\alpha)}{b}. \tag{20}$$

5.2 *Parametrized Damaged Aircraft Model*

The linearised flight dynamics model can be described as follows:

$$\dot{x} = \left(A - \overline{A}\right)x(t) + \left(B - \overline{B}\right)u(t). \tag{21}$$

$x(t) = [\alpha, q, \theta, v_{\text{tot}}, \text{thrust}, p, r, \beta, \phi, \psi]'$ is the state, $u(t) = [\delta e, \delta a, \delta r]'$ is the control input. A and B denote the linear model of undamaged aircraft where as $\overline{A}$ and $\overline{B}$ denote the delta change due to damage.

5.2.1 Example of Linear Aircraft Model for Damaged Vertical Tail

A 6-DOF simulation is performed for a generic undamaged fighter aircraft at flight conditions of 0.39 mach, altitude of 3.16 km and at an angle of attack of 7.73° and the corresponding linear model can be obtained in terms of A and B state matrix. Next, using Roskam's estimation as described in Sect. 5, the change in stability and control derivatives for three different percentage loss of vertical tail is shown in Table 2.

Subsequently, a state matrix for $\overline{A}$ and $\overline{\text{B}}$ can be generated for each of the damage case of vertical tail area. Finally, the linear model of damaged aircraft can be obtained using Eq. (21) for different percentage of damages for different flight conditions. Thus, the generated linear models can be used for control reconfiguration of aircraft for safe recovery.

Table 2 Estimated derivatives for different percentage of vertical tail loss

Stability derivatives	25%	50%	75%
$C_{y_\beta V}$	−0.0057	−0.0057	−0.0057
$C_{l_\beta V}$	0.0059	0.0068	0.0074
$C_{n_\beta V}$	−0.0127	−0.0120	−0.0113
$C_{y_p V}$	0.0118	0.0135	0.0148
$C_{l_p V}$	−0.0131	−0.0182	−0.0225
$C_{n_p V}$	0.0263	0.0286	0.0292
$C_{y_r V}$	−0.0253	−0.0241	−0.0225
$C_{l_r V}$	0.0263	0.0286	0.0292
$C_{n_r V}$	−0.0563	−0.0508	−0.0445
Control derivatives	25%	50%	75%
$C_{y_{\delta r} V}$	−0.0013	−0.0013	−0.0013
$C_{l_{\delta r} V}$	0.0019	0.0022	0.0024
$C_{l_{\delta r} V}$	−0.0045	−0.0043	−0.0041

6 Conclusion

Reliable and accurate fault detection is key to identification and subsequent estimation of change in aircraft dynamics. In this chapter, it is demonstrated that vision-based damage detection gives reliable identification of damage and extent of damage can be accurately determined using simple geometric methods. Subsequently, analytical estimation techniques can be used to arrive at change in stability and control derivatives is due the damage. The state matrix generated using the non-dimensional derivatives can be used for control reconfiguration of aircraft as per desired linear or nonlinear control design methods. Further work can be carried out for visio- based detection at different camera view angle and in different lighting conditions including that in night. On-line parameter estimation techniques based on algorithms such as recursive least squares method can be employed to validate the parameters from geometric method.

References

1. C. Edwards, T. Lombaerts, H. Smaili, *Fault Tolerant Flight Control: A Benchmark Challenge* (Springer, 2010)
2. L. Crider, Control of commercial aircraft with vertical tail loss, pp. 1–11 (2012)
3. G. Bramesfeld, M.D. Maughmer, S.M. Willits, Piloting strategies for controlling transport aircraft after vertical-tail loss. J. Aircr. **43**, 216–225 (2008)
4. X. Li, H.H.T. Liu, A passive fault tolerant flight control for maximum allowable vertical tail damaged aircraft. J. Dyn. Syst. Meas. Control **134**, 031006 (2012)
5. G. Shah, Aerodynamic effects and modeling of damage to transport aircraft. AIAA Atmos. Flight Mech. Conf. Exhib. 1–13 (2008)
6. J. Canny, A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell. **8**, 679–714 (1986)
7. Canny Edge Detector, https://en.wikipedia.org/wiki/Canny edge detector, Last accessed Sep 2019
8. MATLABProgram, https://in.mathworks.com/matlabcentral/fileexchange/46859-canny-edge-detection
9. Y. Hitachi, Damage-tolerant flight control system design for propulsion-controlled aircraft (2009)
10. H.N. Nabi, T. Lombaerts, Y. Zhang, E. van Kampen, Q.P. Chu, C.C. de Visser, Effects of structural failure on the safe flight envelope of aircraft. J. Guid. Control. Dyn. **119** (2018)
11. J. Roskam, Methods for estimating stability and control derivatives for conventional subsonic airplanes, 1–3 (1973)
12. Bandu N. Pamidi, Performance, stability, dynamics and control. AIAA, 270–277 (1998)

Development of Hardware-In-The-Loop Test Bench for Automation of After-Treatment Control Systems Tests

Arundhatti Bezbaruah, Bhanu Pratap, and Sandeep B. Hake

1 Introduction

Engine in a vehicle is the vital part that powers and controls it. Cummins Pvt. limited, being an OEM manufactures engines ranging from 23 to 4400 HP used for light duty, medium duty, heavy duty, on-highway, and off-highway applications. There are various sources of exhaust emission like power plants, gensets, cars, trucks, ship, locomotives, motor cycle, etc., and the basic regulated constituents of exhaust gases are hydro carbon (HC), carbon monoxides (CO),oxides of sulphur (SOx), oxides of nitrogen (NOx), and particulate matter (PM). For clean emissions from these power sources, there is an after-treatment system which ensures that the emissions of exhaust gases are below the government required regulations. Government of India has instituted Bharat Stage Emissions Standards to control the pollutants from vehicles. BSVI emission norms are going to be implemented nationwide from April 2020. Cummins Emission Solutions designs and manufactures the after-treatment technology to meet the state-of-the-art emission standards worldwide.

The rest of the paper starts with the description of after-treatment system and different techniques in use to reduce the pollutants from diesel engines. Second, a brief overview of the types of different test that is performed to check the working

A. Bezbaruah (✉) · B. Pratap
Electrical Engineering Department, National Institute of Technology Kurukshetra, Kurukshetra, India
e-mail: bzbarn33@gmail.com

B. Pratap
e-mail: bhanumnnit@gmail.com

S. B. Hake
Cummins Emission Solutions, Cummins Technical Centre India, CES, Pune, India
e-mail: sandeephake@outlook.com

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_9

of the components and the integrated system is given. Third section describes the hardware in loop simulation and different hardware and software tools used for manual testing along with the problem or disadvantage associated with the process of manual tests. Fourth section demonstrates the automation of manual tests and tools required for it and are validated with two considered test cases and their results.

1.1 After-Treatment System

There are two types of engines, mechanically controlled and electronically controlled. Earlier, most of them are mechanically controlled. With the proliferation of electronics, it is possible to meet the emission norms as electronic control unit ensures integration of the engines with the after-treatment system. Different engine manufacturers provide various techniques to reduce the emitted pollutants. Exhaust gas recirculation (EGR) is one such technique of rerouting back the polluted gas to engine cylinder that decreases the temperature of combustion chamber by absorbing the heat of exhaust gases. Reduction in temperature reduces oxygen and thus the NOx. In diesel oxidation catalyst (DOC), oxidation reaction reduces carbon monoxide and hydrocarbon by 90%. Increase in temperature due to oxidation burns the particulate matter (soot and ashes) and reduces it by 25–50%. Diesel particulate filter (DPF) along with selective catalyst reduction (SCR) reduces more than 90% PM but increases the NOx level [1–3]. Nowadays, urea dosing system used with SCR helps to reduce the NOx and PM. Dosing is a technique of precise urea injection to the diesel exhaust flow at a spray angle. The two types of dosing system in use and they are air-assisted and liquid-assisted.

1.2 Tests

Tests are required to be performed regularly after intervals to check the functioning of the software that controls the hardware in real. Software controls logic are updated or modified as per the customer's requirements and need to be validated. The regression tests in controls laboratory carry a variety of tests to check the proper functioning of software before delivering it for practical applications on field.

- Functional failure mode effect test: The functional tests are done to check fault conditions generated by sending faulty messages, interrupting the power supply, etc.
- Electrical failure mode effect test: This test is to check the circuit continuity fault conditions when input or output signal lines to ECM are open circuit, short circuit to ground, and battery positive (5 V/24 V).
- Feature test: It is done to check the working of control algorithms to detect the functionality of different diagnostics related to the after-treatment system [4, 5].

Regression testing is required to be done during different phases of software development process to recheck its working after changes or up gradation. Optimization serves to provide the best result. The purpose of optimization technique for the test cases is to reduce the number of test cases and their execution time along with detection of faults in the testing process. Genetic algorithm, wind-driven optimization, particle swarm optimization, etc., are such techniques of optimization where proper assessment of different optimized test cases for modified components can be done [6].

2 Hardware-In-The-Loop Simulation

Hardware-in-the-loop (HIL) system is a real-time simulation technique for the development and testing of embedded systems. It depicts how the controller will give output to virtual but realistic inputs. It is basically used for the validation of the modeled system and thus enhances the safe operations of machines.

2.1 Manual Test Bench

In manual test bench set-up, the test engineer simulates the test cases by sending CAN messages and simulates electrical fault conditions manually using breakout box by connecting the jumper wires to ground or battery corresponding to a fault condition. The manual set-up contains ECM, wiring harness, manual breakout box, dosing system, CAN adapter, and all the sensors and other actuators as depicted in Fig. 1.

- Electronic Control Module: It acts as the control unit of the whole system. The electronic processor takes the control actions between sensors and actuators and

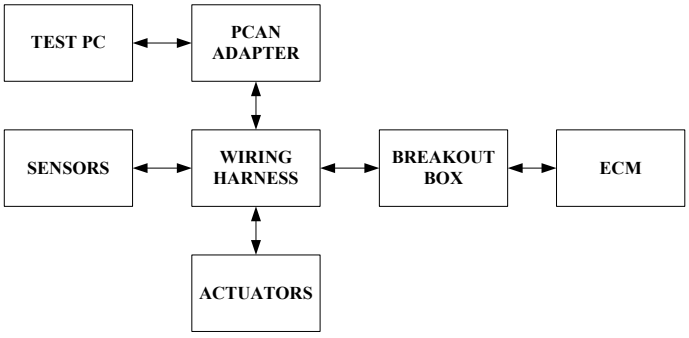


Fig. 1 Block diagram of manual test bench

makes on-board decisions. It detects the presence of fault conditions and fault codes corresponding to specific fault condition shown in the calibration tool in test pc/host.

- Wiring harness: It makes the connection between power supply, ECM, test pc via PCAN adapter and all the sensors and actuators required to be connected according to the need of test.
- CAN adapter: An adapter is used to connect the test pc with the wiring harness. It works on CAN protocol of SAE J 1939 for a serial control and communications of vehicle network.
- Testing/validation tool: It is Cummins-specific software that is used to verify all the changes while performing a test.
- Manual breakout box: It is a device inserted between two electrical devices to diagnose the communications and continuity and the user can connect or disconnect different signals using jumper wires.
- Dosing system: It is the vital part of after treatment system used for injection of diesel exhaust fluid (DEF) to the exhaust gases flow. DEF consists of 32.5% urea and 67.5% water. The amount of DEF to be injected by pump is controlled by the ECM in proportion to the amount of NOx present in the exhaust. DEF breaks into ammonia in presence of heat. Ammonia in presence of SCR catalyst converts NOx into harmless nitrogen and water [7].

The dosing system has sensors like temperature, level, quality; pressure sensors and actuators like pump motor, dosing system, injector valve, supply module heater, air shut off valve and metering valve.

2.2 Problem Statement

In a regression test, there are more than 200 fault conditions are to be tested and annually there are four software release cycles. So, total numbers of tests to be done is $4 \times 200 = 800$. Therefore, the time required in manual test bench is much more [1]. The chances of manual errors, e.g. parallax and variation in judgments are more in manual testing and does not lead to correct results. Therefore, automation of test procedures is useful in conducting the tests to get faster results.

3 Automation of Hardware-in-the-Loop Tests

In the development of automated test benches, Load User Interface Box (LUIS), and Digital FMET are used along with the ECM, CAN adapter, Dosing system and harness. Digital FMET box is used in place of manual breakout box to simulate the fault conditions.

- Load User Interface System (LUIS): It is an engine simulator used to facilitate testing of engine control system hardware and software. Different values of engine and after treatment system's parameters measured through sensors can be provided using LUIS without using the actual sensors. LUIS hardware has modules like wave-maker, resistive load, analogue, switch [8] (Figs. 2 and 3).

- Digital FMET box: It is the replacement of manual break out box. It has relays to simulate the fault conditions like open circuit and short to battery positive and ground. The relays are controlled electronically. It is a gateway between ECM and LUIS.

In the automation of tests, test sequences are scripted in NI TestStand and it sends commands to the ECM accordingly.

To simulate different test cases as referred in Fig. 4, first, the configuration and calibration files are flashed into the respective ECM of Cummins whose software logic is to be tested and the automation of test cases is done. After the flash being

Fig. 2 LUIS box

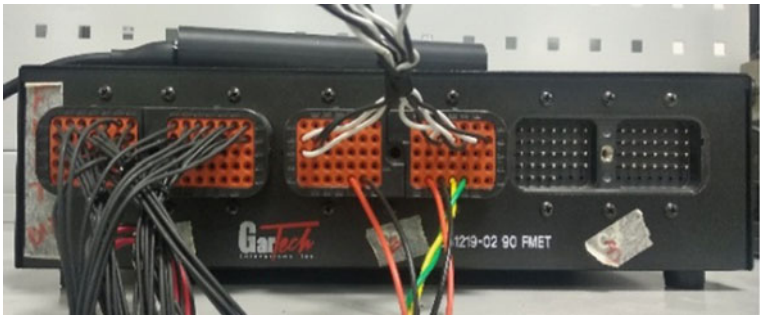


Fig. 3 Digital FMET box

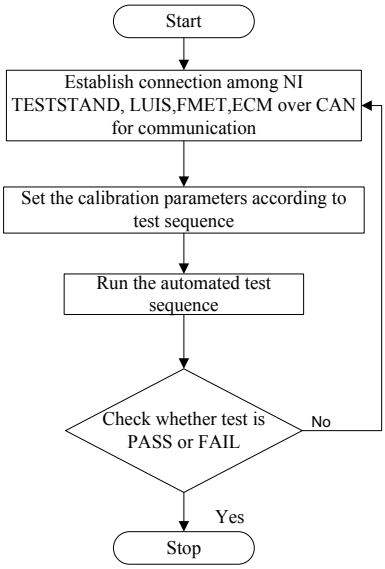


Fig. 4 Steps for automation of Tests in NI TestStand

successful, the automated test procedure for a test is run in NI TestStand. A test report is generated automatically while the test is under execution regarding the result of the test that whether the steps in the scripted automated test procedure are satisfied or not and, respectively, PASS or FAIL status is created.[9, 10].

4 Results and Implementations

As a case study, functionality of dosing system is checked by performing both manual and automation testings. With reference to Fig. 5, the working of dosing system is such that initially when the key is on and engine is at running state, the control unit reads the SCR temperature, ambient air temperature, and the DEF tank temperature and decides for thawing. The controller of the pump commands according to the dosing request and pressure starts to increase. As dosing started, increase in intake NOx to the after treatment system increases the dosing command rate and thus the urea injection to exhaust. Urea reduces the NOx into harmless Nitrogen gas which is emitted into the atmosphere while vehicle is on road. The increase or decrease in dosing command rate as function of intake NOx or exhaust gases can be monitored in the testing/validation tool in the PC.

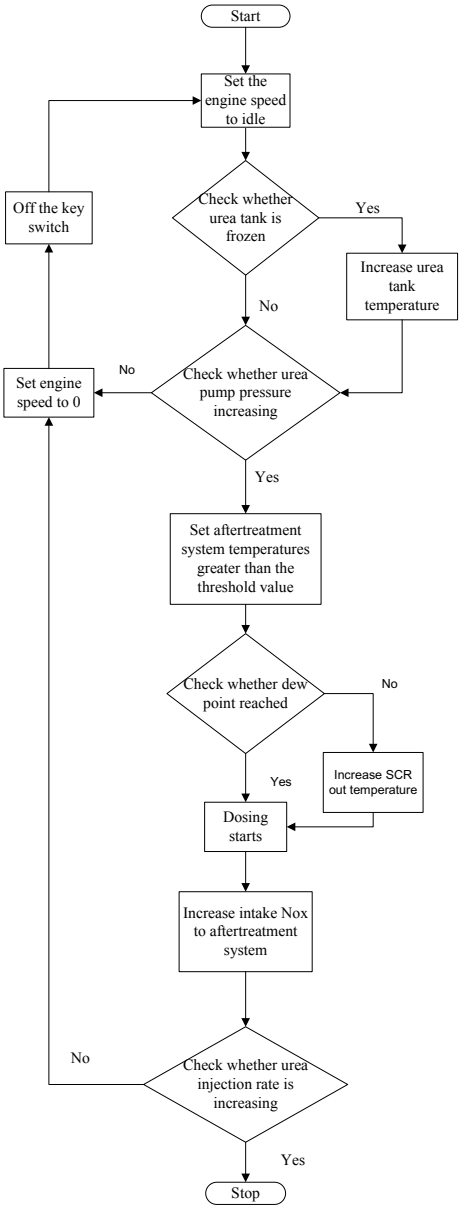


Fig. 5 Algorithm to simulate dosing

4.1 Manual Test Results

After executing the dosing test, a log sheet is generated. It contains the information regarding different inputs given through CAN like speed of vehicle, after-treatment temperatures, flow rate of exhaust gases, amount of NOx present, etc. and output of

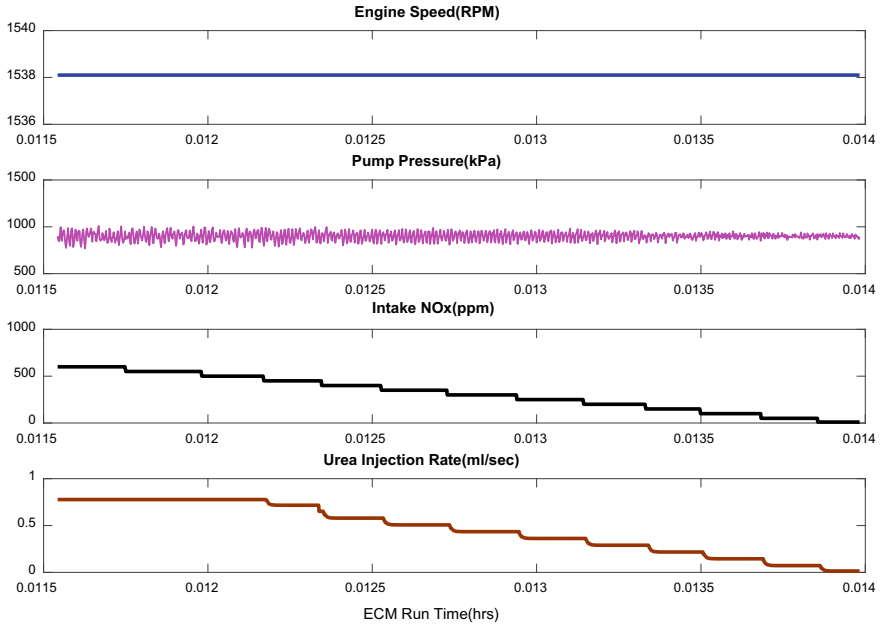


Fig. 6 Graph of automated dosing

all the simulated inputs is the optimized urea injection rate as the output. Figure 6 shows the change in injection rate of urea as the amount of NOx in exhaust gases changes [11].

4.2 Automation Test Results

In the automation of dosing test, total bench configuration is done in NI TestStand to use the automated bench set-up and session with the scripting host is established. There are different phases for which software logics differ and so the test sequences. Prioritization of test cases using optimization technique is done for the best fit test case depending on the metadata of which phase is flashed in the ECM. The code checks conditions of modified software and executes the appropriate test case present in the subsequences. As the test execution is started, according to the sequences in the automated test procedure values of different parameters used are fetched into the ECM from LUIS box and dosing command is actuated after receiving defined inputs with proper set levels. A report for this feature test is generated as in Fig. 7 which dictates the accuracy of each test step with optimization in the execution of test case as shown in Table 1 [12, 13].

Action Description	Actual Value	Target Value	Units	Tolerance	Pass/ Fail
Verify pump pressure starts increasing and greater than low threshold	831.6	>=750	kPa	0	Pass
Verify pump pressure starts increasing and less than high threshold	836.9	< 950	kPa	N/A	Pass
Sending CAN message for input Nox as 100ppm					
Verify Nox_in value	100	100	ppm	0	Pass
Wait until urea injection rate increments and stabilizes					
Verify urea injection rate is greater than 0	0.1265	>0	ml/s	N/A	Pass

Fig. 7 Part of feature test report of dosing test

Table 1 Total time saved by using automated test bench

Tasks done	Time taken		
	Manual test time	Automation test time	Percentage time saved (%)
Test bench set-up	20	28	−40
Test time duration	120	50	+58.33
Total time	140	78	+44.28

5 Conclusion

The developed hardware-in-the-loop test bench allows performing the test in less time with overall time saving of 44.28% w.r.t. manual test while performing the dosing test. The automation work reduces the manual efforts and observational errors by performing the test on its own without any dependency on test engineer after starting the test. Scripting of tests done according to the standard procedures can be used for different regression test programs with different ECMs and compatible wiring harness. The optimization of test cases allows scripting of single test case comprising of sub-test cases which can be used according to different conditions or requirements in the software.

References

1. S.B. Hake, M. Mukhedkar, Design and development of universal test bench for engine aftertreatment controls system. Int. J. Adv. Res. Electron. Commun. Eng. **6**(4), 309–312 (2017)
2. A. Mouzakitis, A. Nayak, A. S. Puthiyapurayil, Automated fault diagnostics testing for automotive electronic control units deploying hardware-in-the-Loop, in *UKACC International Conference on Control*. Coventry, UK, 7–10 Sep 2010
3. D. Vashist, N. Kumar, M. Bindra, Technical challenges in shifting from BS IV to BS-VI automotive emissions norms by 2020 in India: a review. Arch. Curr. Res. Int. **8**(1), 1–8 (2017)
4. N.M. Raul, N.M. Wagdarikar, C.R. Bhukya, Development of hardware-in-loop automated test bench for liquid-assisted after-treatment controls system's regression tests, in *International Conference on Computing Communication Control and Automation*. Pune, India, 16–18 Aug 2018

5. P. Malani, C.R. Bhukya Rao, J. Vishwasrao, J. Sonkusare, Automation of FMET and regression tests for doser after-treatment controls using load-box user-interface-system hardware-in-loop bench. Int. J. Innov. Res. Electr. Electron. Instrument. Control Eng. **5**(5), 170–176 (2017)
6. K. Tyagi, K. Tyagi, A comparative analysis of optimization techniques. Int. J. Comput. Appl. **131**(10), 6–12 (2015)
7. https://www.cummins.com/components/aftertreatment
8. https://gartechenterprises.com/luisnextgen_v2.0.pdf
9. R.M. Kulkarni, R.P. Patil, C.R. Bhukya, Hardware–in–loop test bench based failure mode effects test automation. Int. J. Innov. Res. Elect. Electron. Instrument. Control Eng. **4**(10), 27–30 (2016)
10. M. Mane, S.J. Kulkarni, V. Gupta, Automation of FMET testing, in *International Conference on Computing Communication Control and Automation.* Pune, India, 16–18 Aug 2018
11. J. Ning, F. Yan, Detection of injected urea quantity and correction for SCR urea dosing control, in *SAE Technical Paper*, 2015-01-1038
12. P. Padmnav, G. Pahwa, D. Singh, S. Bansal, Test case prioritization based on historical failure patterns using ABC and GA, in *International Conference on Cloud Computing, Data Science and Engineering*. Noida, India, 10–11 Jan 2019
13. R. Dhiman, V. Chopra, Novel approach for test case prioritization using ACO algorithm, in *International Conference on Information and Computer Technologies.* Kahului, HI, USA, 14–17 Mar 2019

Intelligent Control System for Water Pollutant Monitoring Using ANN and Fuzzy Logic

Divya Pandey, Satanand Mishra, and Shivani Pandey

1 Introduction

The water bodies are polluted in different major areas of biosphere [1]. The advanced techniques should be used to identify the water pollutants in water distribution system. In this paper, the intelligent control system is used having advanced techniques artificial neural network and fuzzy logic for the detection of water pollutants in system. Artificial neural network approach has been applied to different areas of science. ANN technique is used to identify hidden relationships, thus facilitating the prediction and forecasting. ANN is used to approximate input–output relationships in nonlinear manner [2]. This technique is used to solve the problems involving decision making to observe the pollutants in different water resources for identifying water quality parameters [3]: continuous monitoring, supply of water, detection of pollutants and controlling the water supply in significant manner in water distribution system. The water distribution system infrastructure consists of pipes, valves, reservoirs, storage tanks and other hydraulic appliances. It is designed to deliver sufficient amount of water quality and quantity to the different water resources. Water pollutants are produced by different sources such as domestic wastes, insecticides and herbicides, food processing waste, pollutants from livestock, chemical waste, heavy metal, volatile organic compounds. The main reason for water pollutants and water quality is affected by exploitation of natural resources [4]. Water quality problems are generated due to point and non-point sources of pollution [4]. The quality of water is

D. Pandey
Madan Mohan Malaviya University of Technology, Gorakhpur, U.P. 273006, India

S. Mishra (✉) · S. Pandey
AcSIR & CSIR-Advanced Materials and Processes Research Institute (AMPRI), Bhopal 4620206, India
e-mail: snmishra07@gmail.com

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_10

very important, either it is source of life or source of diseases [5]. Regulation of water distribution system is based on "if–then" fuzzy logic-based rule which is related with logical implications and reasoning [3]. Fuzzy logic concepts are representing uncertainty due to vagueness. In this chapter, the fuzzy sets are used for the detection of water pollutants in system to identify the problems involving and to overcome the uncertainty in the water distribution system. A fuzzy inference system is related to fuzzy set theory, which interconnects input values to output [10]. The membership function used in fuzzy logic concepts is related with the properties such as normality, symmetry and monotonicity [3]. Water quality index gives detail on a scale from 0 to 100. More values of WQI show good quality of water and vice versa [9].

This chapter deals with the fact that water pollutants in water distribution system have led to monitored and controlled in specific manner and using the advanced techniques as ANN, fuzzy logic and wireless sensor. The intelligent control system is designed using these advanced methods for monitoring the water quality parameters like pH, conductivity, turbidity, dissolved oxygen, total dissolved solids.

2 Study Area

For the demonstration purpose and reality check in the ground level for intelligent control system, the study area is selected Gorakhpur. By making the comparison of water pollutants in three different regions of Gorakhpur, observations of main factors responsible for water quality parameters are obtained. The three different regions are given as: The first region is Rapti River which flows in Gorakhpur, U.P., India and it is located 28°28′32.52″N latitude and 82°52′43.68″E longitude. The second region is Ramgarh Taal which flows in Gorakhpur, U.P., India, and it is located 26.7321°N latitude and 83.4056°E longitude. The third region is supply tap water which flows in municipal area, Gorakhpur, and it is located 26°45′57.0384″N and 83°21′53.7984″E longitude.

3 Methodology

This chapter covers the parameters affecting the water quality of different regions. These parameters help to maintain the water quality level as well as provide different ranges of different parameters to maintain their quality. Water quality estimation needs not only an inspection of water pollutants and the recognition of main water pollutants factors but also the evaluation of contaminated risky regions. To obtain this goal, the water samples are collected from three different regions, and by comparison of the water samples of these regions, water pollutants are detected according to the water parameters.

3.1 Standard Parameters of Water

Water is major abiotic factor in the environment. Water is essential commodity to all life, and without water, there can be no life. Three fourth of the earth surface is covered by water. Smart water quality techniques and the system having advanced methods to identify the water pollutants in distribution system. The wireless sensor networks are used to monitor the water parameters over a water distribution system and send the monitored parameters to intelligent control system using wireless communication [4].

Based on extensive evaluation by World Health Organization (WHO), the standard parameters of water quality are given by Table 1.

In this chapter, the quality of water is determined by testing and taking the samples of three different regions with the help of different standard parameters of water quality, for detection of water pollutants that may help to solve future problems.

In Fig. 1, variation in electrical conductivity (μs/cm) is measured for three different areas. It's average value obtained for residential area is 445 μs/cm, for commercial area is 766 μs/cm, and for vegetated area is 348 μs/cm, respectively.

Table 1 Standard parameters of water quality (drinking water)

Water parameters	Range	Remarks
pH	6.5–8.5	Surface water Identify acidity or basic in nature
Turbidity	1NTU to 5NTU	Total suspended solids
Conductivity	200–800	Micro-sec/compresence of ions
Temperature	0–60 °C	Presence of organic compounds
BOD	Upto 5 ppm	–
Chlorine concentration	Upto 4 ppm	–

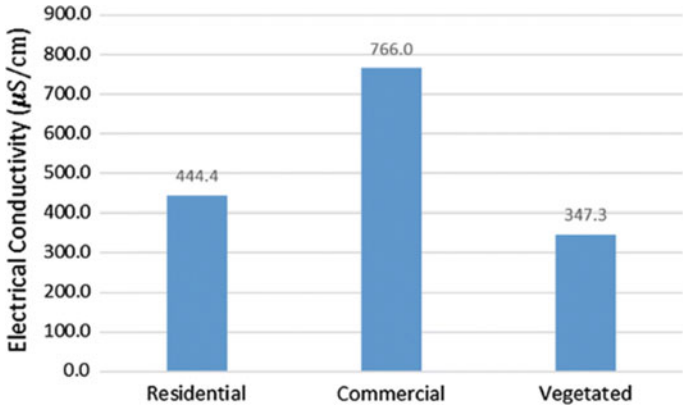


Fig. 1 Variation in electrical conductivity

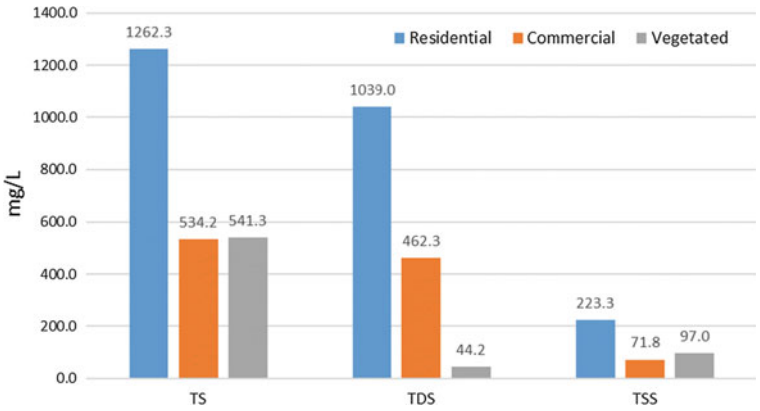


Fig. 2 Variation in TS, TDS, TSS

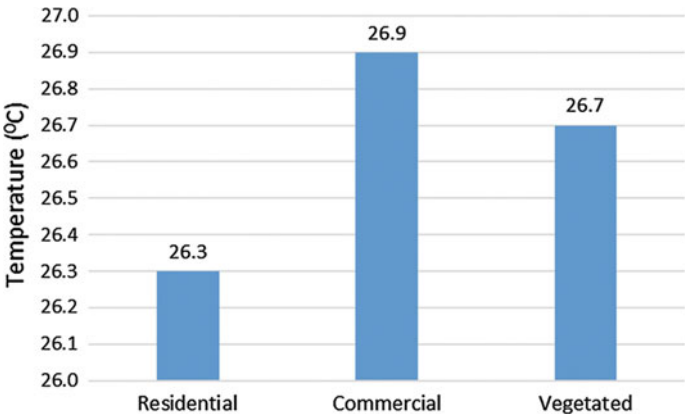


Fig. 3 Variation in temperature

In Fig. 2, variation in TS, TDS and TSS for three different regions is concerned.

In Fig. 3, variation in temperature is measured for three different regions. For residential area, average value is 26.3 °C, for commercial area, average value is 26.9 °C, and for vegetated area, average value is 26.7 °C.

3.2 Water Parameters

In this study, the information related to contamination of water is important to the current monitoring techniques and identification approaches for safeguarding the habitat [12]. The ultimate aim of water quality monitoring is to protect different

water bodies. In fact, land usage and enactment are probably the most dominant constituents in determining water quality in different regions.

3.2.1 Temperature

The temperature of different sources of water is variable with time. The variables are atmospheric temperature of solution in degree celcius with time duration in hours.

In Fig. 4, 5 and 6, temperature trend for different water sources is determined. For tap water, as time increases, temperature decreases for few hours then increases.

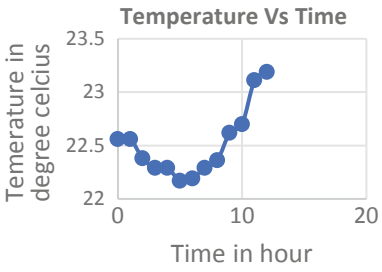


Fig. 4 Temperature trend for tap water

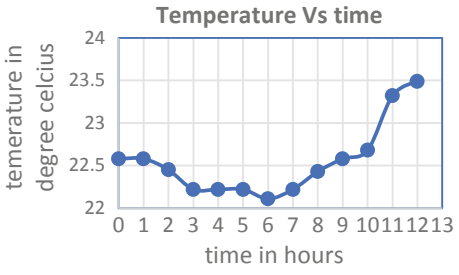


Fig. 5 Temperature trend for surface water

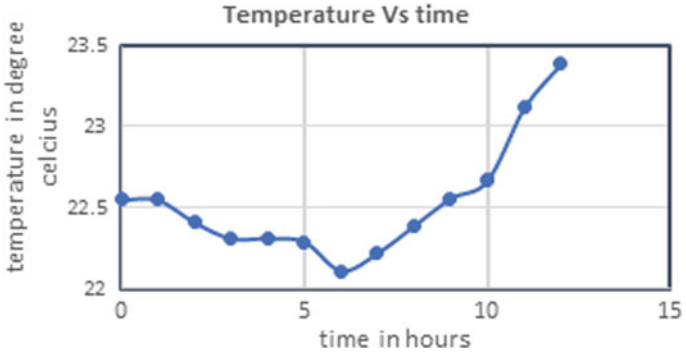


Fig. 6 Temperature trend for water pollutants

For surface water, as time increases, temperature decreases for the first few hours; then, after that it increases as time increases. For water pollutants, temperature first decreases for the first few hours, and then, it increases as time increases. Temperature range usually exists between 22 and 24 °C, in general at room temperature. Observation is done for the first fifteen hours.

3.2.2 pH

pH shows the logarithmic scale which ranges from 0 to 14. The aquox solution pH usually ranges from 6.5 to 9. More pH is toxic and dangerous and increases the solubility of elements, and compound temperature varies with inverse ratio with pH.

In Figs. 7, 8 and 9, for tap water, the first two hours pH are constant; then, after that, pH increases for increasing hours; after that, it follows decreasing and constant trend for increasing hours. For surface water, for the first few hours pH increases for the first increasing hours, and then, there is a drop in pH for certain specific hour approximately at 4 h; then, there is sudden increase in the next hour; after that, it decreases for increasing hours. For water pollutants, there is an increasing pattern for the first six hours, and then, it decreases for few continuous hours; after that, there is a sudden increase in pattern for 8–10 range of hours, and there is a decrement in pattern. All the sources pH usually range from 7 to 9, basic in nature.

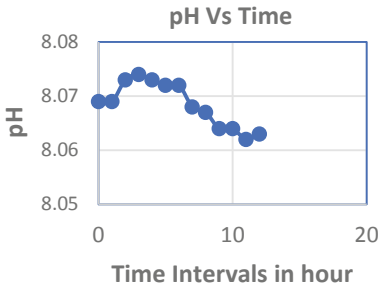


Fig. 7 pH versus time trend for tap water

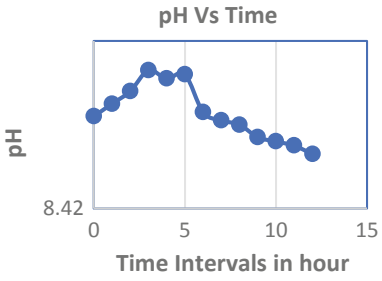


Fig. 8 pH versus time trend for surface water

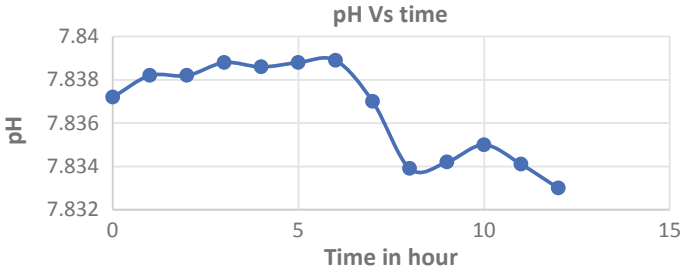


Fig. 9 pH versus time trend for water pollutants

3.2.3 Conductivity

It shows the ionization potential of any solution. I is the measurement of electrical conductivity. Its unit is µs/cm. If the concentration of ion increases, conductivity increases. Conductivity of tap water is usually lower than seawater due to low rate of ionization. Conductivity increases due to the presence of pollutants in water. Conductivity also increases with salinity. Conductivity increases with direct proportion with temperature. When deviation is within 2% range, then only it follows linearity, otherwise not followed.

In Figs. 10, 11 and 12, for tap water, conductivity is constant for the first two hours, then decreases for next few hours, and then, there is a sudden increase in conductivity, then falls for the next continuous hour, again increases for the next hours, then sudden falls at seventh hour, and then, there is a continuous increase in conductivity. Conductivity usually measured in µs/cm and ranges within 57–58. For surface water, it is constant for the first few hours, and then, there is fall in

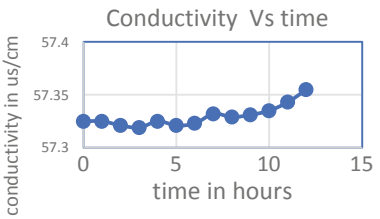


Fig. 10 Conductivity versus time trend for tap

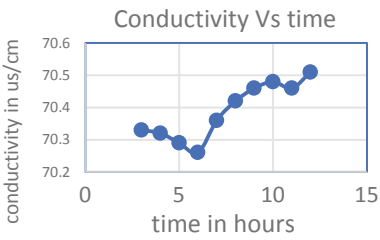


Fig. 11 Conductivity versus time trend for **a** water, **b** surface water

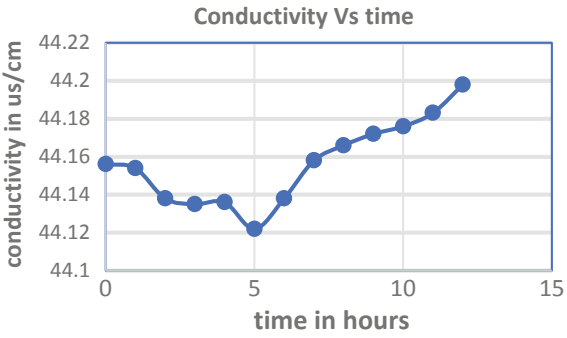


Fig. 12 Conductivity versus time trend for water pollutants

conductivity; after that, it increases upto tenth hour; then, there is a fall for the next continuous hour; after that, it increases ranges from 70 to 71 µs/cm. For water pollutants, it remains constant for the first hour, then falls in conductivity for the next continuous hour, then again constant for the next two hours, then falls for the next hour; then, there is a continuous increase after fifth hour which ranges from 44 to 45.

3.2.4 ORP

ORP stands for oxidation–reduction potential which is the measure of the oxidizing capacity solution. As the ORP increases, oxidation capacity increases. ORP range for tap water is usually high.

In Figs. 13, 14 and 15, for tap water for the first two hours, it decreases, and then, there is a sudden increase in next hours; then, it remains constant for third to ninth hour, then sudden decrement for next hour, then remains constant for 10–12 h, ranges from 0.2 to 0.6 V. For surface water, it remains constant for entire period, ranged from 0.2 to 0.3 V. For water pollutants, it remains constant for entire period, ranged from 0 to 0.004 V.

Fig. 13 ORP trend for tap water

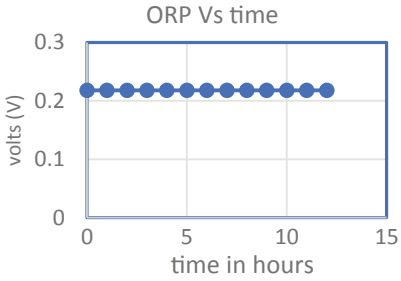


Fig. 14 ORP trend for surface water

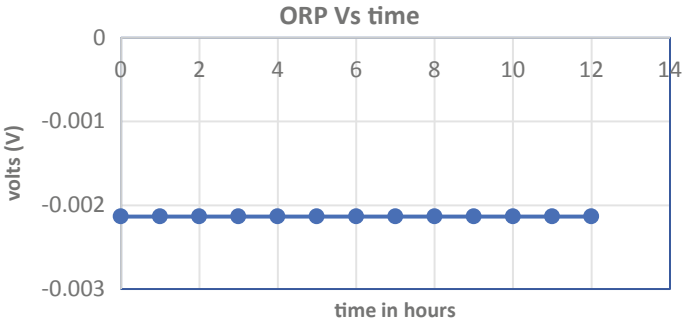


Fig. 15 ORP trend for water pollutants

3.3 Sensors Used for Determining Water Pollutants

The water quality assessment is important for delivering in different distribution networks. The standard parameters are given by World Health Organization (WHO) to fulfill the requirement in systems. Availability, compatibility, reliability and cost effective are the main parameters for the selection of high-performance sensors for monitoring the water pollutants [6]. Sensors are connected to the controller using Zigbee protocol. The main reason for the excessive sources of water pollutants is ill treatment of natural resources. The sensors monitored the distribution network for the observations of raw readings, and then, the readings are examined and give the resultant to determine the status of pollutants (Table 2).

3.4 Controller

In water pollutant monitoring system, the wireless sensor network is used to observe the water parameters and then signal given to controller. It is the main component of distribution system by which the decision making is done and observed and readings are monitored in efficient manner.

Table 2 Different sensors for monitoring water pollutants

Sensor	Description
TOC sensor	Total organic carbon
Turbidity sensor	Total suspended solids
Chlorine residual sensor	Amount of chlorine
Conductivity sensor	Conductivity of water
ORP sensor	Oxidation reduction potential
pH sensor	Acidic or basic in nature

3.5 Intelligent Controller

Artificial neural network (ANN)-based methodology is developed to observed the hidden relationship between the available identified values and the related sources measurements, resulting in observation of unknown water pollutants. These advanced techniques are used for decision making in identifying the problems related to water pollutants' monitoring system. The artificial neural networks are just like human brain with neuron nodes interlinks like a web. Human neurons are responsible for signal processing and sending and receiving signals in network. ANN can be implemented for water treatment system using standard water parameters (water pollutants sources and waste features) as inputs and the water pollutants as system outputs [8].

The fuzzy sets and fuzzy logic developed the base rules for nonlinear networks. The Mamdani-type fuzzy inference model has been used; fuzzification and defuzzification are used in distribution network. The fuzzy set represents the uncertainty in different types of systems. The system has been structured and implemented with the main reason for identifying the sources of water pollutants and store the information in database or the analysis in real time.

The ANN-fuzzy-based intelligent controller is designed for smart monitoring of water pollutants in water distribution system. Different types of pollutants present in systems due to various reasons such as industrialization, agriculture growth and misuse of natural resources. The conventional PID controllers' techniques are unable to solve the problems related to system complexity, uncertainty and nonlinearity. Intelligent control system is designed to analyze the problems in real-time operation [7]. The intelligent control system is just like human decision making than conventional logical systems.

Block Diagram

This problem is relevant in simulation-based control in water distribution system. The simulation models are required to reflect the behavior of real-time system in effective manner and introduce a methodology for controlling the distribution network by using ANN. This approach has advantages compared to other approaches. Different types of topologies of artificial neural network are used in different types of models. The architecture of neural network consists of individual scalar inputs, each weighted

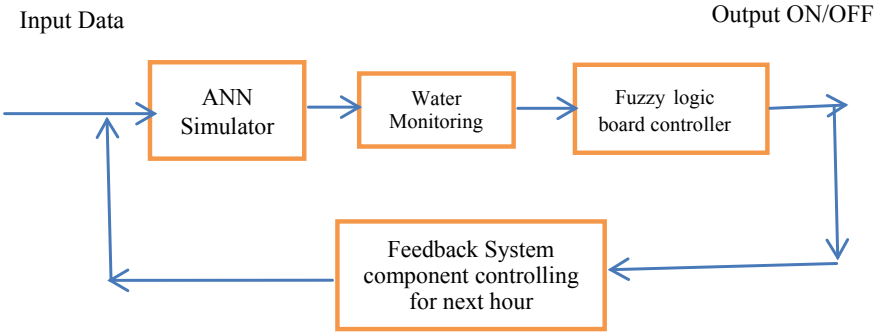


Fig. 16 Block diagram of intelligent control system

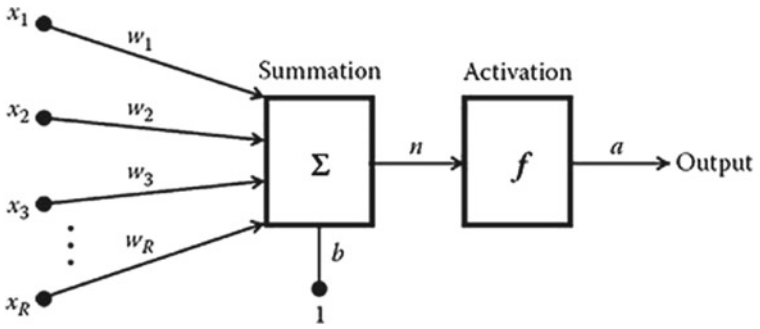


Fig. 17 Structure of artificial neuron

with appropriate values. The summation of the weighted inputs and bias form of net input gives transfer function (Figs. 16 and 17).

Fuzzy logic controller is designed by fuzzy inference system. Fuzzy inference system is based on four steps that is fuzzification, base rule, inference engine (decision-making process) and defuzzification methods. Membership functions are required to convert crisp values to fuzzy linguistic variables [11]. Linguistic variables are required in fuzzy logic controller to represent the input variables and output variables. The water parameters temperature, pH, conductivity and ORP curves variation show the presence of water pollutants in this distribution network. These advanced techniques include the intelligence to the controller in water pollutants monitoring. The desired value for the water quality parameter is entered as set point and then compared with this set point of controller by feedback element. The desired water pollutants monitoring is done by these steps followed by intelligent controller for the best results obtained.

Table 3 Summarized result

Source	Temperature (°C)	pH	ORP (mV)	Conductivity (μs/cm)
Raptiriver	18–28	7.5–8.5	185–230	85–95
Supply tap water	18–28	7.5–8.3	320–590	50–68
Water pollutants of RamgarhTaal	18–28	7.5–8.0	0–3	38–47

4 Results

A comparison can be made with the collected data between tap water, river water, polluted water. The pH levels for all were fairly similar with the only change being in relation with temperature. Conductivity for the water samples differed significantly because of different salinity concentrations for different water types. The highest conductivity is for seawater and the lowest being that of tap water. ORP for tap water and river water was similar with results being in the low 100–400 mV range. ORP for tap water was observed to be 350 mV which is fine considering that the acceptable range is from 300 to 590 mV.

The data obtained for polluted water has some interesting values for ORP and conductivity. A very low ORP value was observed, averaging at −2 mV which is an indication of overpowering reductants (Table 3).

5 Conclusion

This chapter deals with the advanced techniques used for the monitoring the water pollutants in water distribution system. The ANN and fuzzy logic are used for the improved mathematical modeling than conventional methodology. The water quality is very important for all the living beings for the survival. The wireless sensor network is used for smart water monitoring in distribution system. The experimental model can be improved by using simulations and algorithm by these advanced techniques for monitoring the water pollutants.

References

1. A. Hore, S. Dutta, S. Datta, C. Bhattacharjee, Application of artificial neural network in wastewater quality monitoring: prediction of water quality desalination, researchgate. Int. J. Nuclear Desalination **3**(2), 160–174 (2008)
2. A.M. Najah, A. El-Shafie, O. Karim, A. El-Shafie, Application of artificial neural networks for water quality prediction, researchgate. Neural Comput. Appl. **22**(Suppl 1), S187–S201 (2013)
3. P. Moorthi, A. Singh, P. Agnivesh, Regulation of water resources systems using fuzzy logic: a case study of Amaravathi dam, Springer, applied water science (2018), pp. 1–11

4. G.S. Geetha, Internet of things enabled real time water quality monitoring system, in *Smart Water*. Springer (2017), pp. 1–19
5. A. Demetillo, M. Japitana, E. Taboada, A system for monitoring water quality in a large aquatic area using wireless sensor network technology. Biomed. Central (BMC) J. Sustain. Environ. Res. 1–9 (2019)
6. A. Prasad, K. Mamun, F. Islam, H. Haqva, *Smart Water Quality Monitoring System, Researchgate* (2016), pp. 1–7
7. PUBS (Public Utilities Board Singapore), *SmartWater: Managing the Water Distribution Network with a Smart Water Grid* (Springer, 2016), pp. 1–13
8. H. Avdihodzic, Z. Redzovic, S. Halilovic, Application of artificial neural network in modelling of photo-degradation suspension of manganese doped zinc oxide nanoparticles under visible-light irradiation, in *IFMBE Proceedings*, vol. 62 (Springer, 2017), pp. 137–141
9. M. Sakizadeh, Artificial intelligence for the prediction of water quality index in groundwater systems. Springer Model. Earth Syst. Environ. 1–9 (2015)
10. A. Mourhir, T. Rachidi, M. Karim, River water quality index for Morocco using a fuzzy inference system. Environ. Syst. Res. 1–13 (2014)
11. T. Dereli, C. Çetinkaya, N. Çelik, Designing a fuzzy logic controller for a single intersection: a case study in Gaziantep. Sigma J. Eng. Nat. Sci. 767–781 (2018)
12. R. Altenburger, W. Brack, R. Burgess, W. Burgess, B. Escher, A. Focks, L. Hewitt, B. Jacobsen, M. Alda, S. Ait-Aissa, T. Backhaus, A. Ginebreda, K. Hilscherova, J. Hollender, H. Hollert, P. Neale, T. Schulze, E. Schymanski, I. Teodorovic, A. Tindall, G. Umbuzeiro, B. Vrana, B. Zonja, M. Krauss, Future water quality monitoring : improving the balance between exposure and toxicity assessments of real-world pollutant mixtures, environmental sciences Europe. Springer Altenburger et al. Environ. Sci. Eur. 1–17 (2019)

Design of Fractional-Order PID/PID Controller for Speed Control of DC Motor Using Harris Hawks Optimization

Vijaya Kumar Munagala and Ravi Kumar Jatoth

1 Introduction

PID controllers are being used in industries as reliable components because of their robust nature [1]. These controllers use integer-order controlling mechanisms. Even though they have shown good performance, they also suffer from few drawbacks like sluggish response in rapidly changing environment, poor performance in delay dominant processes, and difficulty in controlling unstable time-delay processes [2]. On the other hand, there are other controllers called fractional-order controllers that minimize the above problems and these work on the principles of fractional mathematics [3]. FOPID controllers are difficult to tune and requires the use of fractional integro-differential equations. One method to simplify the process of tuning is the use of optimization algorithms. Different types of optimizations algorithms were used in the literature to tune the fractional controllers like GWO [4], PSO [5], and constrained PSO (CPSO) [6]. Some of the algorithms discussed in literature to tune general PID controllers were GWO [4, 7, 8], genetic algorithm(GA) [9], invasive weed optimization algorithm (IWO) [10], salp swarm algorithm (SSA) [11], stochastic fractal search algorithm (SFS) [12], and Jaya algorithm (JOA) [13]

In this chapter, a novel approach for tuning FOPID/PID controllers to control the speed of DC motor was discussed. The proposed method was compared with different heuristic optimization algorithm-based tuning methods. Section 1 describes the working principle of HHO optimization and its related mathematical formulation. In Sect. 2, the procedure to apply HHO to tune the parameters of the FOPID/PID

V. K. Munagala (✉) · R. K. Jatoth
National Institute of Technology, Warangal, Hanamkonda, India
e-mail: vijayakumar.nitw@gmail.com

R. K. Jatoth
e-mail: ravikumar@nitw.ac.in

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_11

controller was presented. In Sect. 3, the effectiveness of the proposed algorithm was compared concerning steady-state error, rise time, overshoot, and settling time.

1.1 Overview of Harris Hawks Optimization (HHO) Algorithm

HHO [14] was a meta-heuristic, population-based, gradient-free optimization algorithm developed based on the hunting behavior of hawks. Initially, the hawks perch the search area for prey. Once they identify the prey, the hawks confuse the prey and then make it exhausted. The perching behavior of hawks is like exploration and hunting the exhausted prey is like exploitation.

Let p be a random number, if $p > 0.5$, then the perching is based on family members else it is based on the position of the prey. This is mathematically modeled as

$$Z(t+1) = \begin{cases} Z_{rnd}(t) \cdot p_1 |Z_{rnd}(t) - 2p_2 Z(t)| \text{ if } p \geq 0.5 \\ (Z_{rbt}(t) - Z_m(t)) \cdot p_3(L + p_4(U - L)) \text{ if } p < 0.5 \end{cases} \quad (1)$$

In (1), $Z(t+1)$ represents location of hawks in $(t+1)$ iteration, $Z_{rnd}(t)$ indicates random hawk location, Z_{rbt} (t) is the rabbit location, $Z(t)$ is the present position vector of hawks, $p1$, $p2$, $p3$, $p4$ represents random variables $\in(0, 1)$, the upper and lower bounds are indicated by U and L, and $Z_m(t)$ indicates average position for current population and described in (2).

$$Z_m(t) = \frac{1}{K} \sum_{i=1}^{K} Z_i(t) \quad (2)$$

where $Z_i(t)$ represents location of ith hawk during tth iteration and K indicate the total population.

1.1.1 Condition to Interchange Between Exploitation and Exploration

The algorithm changes between exploration and exploitation phases based on prey energy. The prey gradually loses energy during the hunting process and this is represented in Eq. (3).

$$N = 2N_0\left(1 - \frac{t}{T}\right) \quad (3)$$

Here, N represents the energy of prey, T is the value of final iteration, and $N_0 \in (-1, 1)$ indicates initial energy of prey. If prey energy$N \geq 1$, then hawks enter into the exploration stage, and when $N < 1$, the hawks enter into the exploitation stage.

1.1.2 Exploitation Phase

In this process, surprise attacks are carried by Harris hawks by targeting the planned prey. Let p represents the probability of escaping, then $p < 0.5$ indicates fruitful escape and $p \geq 0.5$ indicates failed to escape before a surprise attack.

Soft Besiege

If $p \geq 0.5$ and $|N| \geq 0.5$, the prey ensures enough energy and it tries to fade out by following tricky movements. These moves are modeled with Eqs. (4) and (5)

$$Z(t+1) = \Delta Z(t) - N|Q * Z_{rbt}(t) - Z(t)| \tag{4}$$

$$\Delta Z(t) = Z_{rbt}(t) - Z(t) \tag{5}$$

where $Q = 2 * (1 - p_5)$ represents the length of rabbit movement.

Hard Besiege

If $p \geq 0.5$ and $|N| < 0.5$, the rabbit is tired and it has low N value. In this case, the positions of hawks are updated with Eq. (6)

$$Z(t+1) = Z_{rbt}(t) - N|\Delta Z(t)| \tag{6}$$

1.1.3 Soft Besiege Using Advanced Moves

If $|N| \geq 0.5$ but $p < 0.5$, indicates prey has sufficient energy to escape. Therefore the hawks follow soft besiege. The positions of the hawks are updated with the equation shown in (7)

$$A = Z_{rbt}(t) - N|Q * Z_{rbt}(t) - Z(t)| \tag{7}$$

The hawks will move according to L-based patterns using the rule given in (8)

$$B = A + D \times L(C) \tag{8}$$

where C indicates problem dimension and D represents a random vector of size $1 \times C$and the leavy function L can be obtained from Eq. (9).

$$L(z) = 0.01 \times \frac{\mu \times \rho}{|u|^{\frac{1}{\gamma}}}, \rho = \left(\frac{\Gamma(1+\gamma) \times \sin(\frac{\pi\gamma}{2})}{\Gamma(\frac{1+\gamma}{2}) \times \gamma \times 2^{(\frac{\gamma-1}{2})}} \right)^{\frac{1}{\gamma}} \tag{9}$$

The random values u, $\mu \in (0, 1)$ and $\gamma = 1.5$. The final equation to update the positions of hawk during soft besiege was given in (10).

$$Z(t+1) = \begin{cases} A \; if \; F(A) < Fn(Z(t)) \\ B \; if \; F(B) < Fn(Z(t)) \end{cases} \tag{10}$$

1.1.4 Hard Besiege Using Advanced Moves

Hard besiege will be performed if $|N| < 0.5$ and $p < 0.5$. In this case, the prey may not contain sufficient energy to escape. This is modeled by using Eq. (11)

Where

$$A = Z_{rbt}(t) - N|Q * Z_{rbt}(t) - Z_m(t)| \text{ and } B = A + D \times L(C) \tag{11}$$

2 Modeling of DC Motor

An externally excited DC motor was considered as plant. The approximate diagrammatic representation of the plant was shown in Fig. 1 and its specifications are indicated in Table 1.

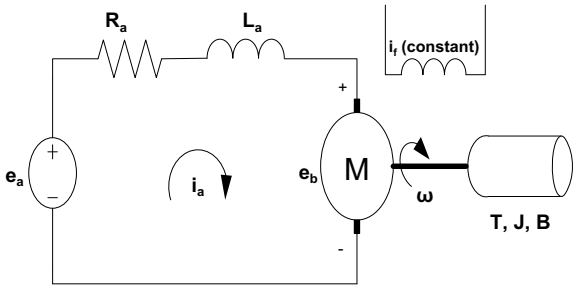


Fig. 1 Approximate model of DC motor

Table 1 DC motor specifications

S. No.	Parameter	Value
1	R_a	0.4 Ω
2	L_a	2.7 H
3	J	0.0004 kg m^2
4	B	0.0022 N m s/rad
5	K	0.015 N m/A
6	K_b	0.05 V S

Using the parameters mentioned in Table 1, the mathematical representation of DC motor was developed. The detailed procedure for modeling the DC motor was given in [10].

3 Fractional-Order PID Controller for DC Motor

Fractional-order PID controllers (FOPID) contain two additional tuning parameters (λ, μ) than integer-order or normal PID controllers (IOPID).The general time-domain equation governing the operation of FOPID controller was given in (12).

$$r(t) = K_p \cdot er(t) + K_i \cdot D_t^{-\lambda} \cdot er(t) + K_d \cdot D_t^{\mu} \cdot er(t) \tag{12}$$

where r(t) is the desired signal, er(t) indicates error signal, and $\lambda, \mu \in (0, 2)$. The terms $D_t^{-\lambda}$ and D_t^{μ} are related to fractional calculus [3] which are defined as

$${}_aD_t^{\alpha} = \begin{cases} \frac{\mathrm{d}^{\alpha}}{\mathrm{d}t^{\alpha}}, & R(\alpha) > 0, \\ 1, & R(\alpha) = 0, \\ \int_a^t (\mathrm{d}\tau)^{-\alpha} & R(\alpha) < 0 \quad \text{Where } a, t \text{ are limits and } \alpha \text{ is order} \end{cases} \tag{13}$$

Another commonly used definition for fractional calculus operator *Grünwald-Letnikov* definition [3] given by

$${}_aD_t^{\alpha} f(t) = \lim_{h \to 0} \frac{1}{h^{\alpha}} \sum_{k=0}^{\left[\frac{t-a}{h}\right]} (-1)^k \binom{\alpha}{k} f(t - kh) \tag{14}$$

where h is the computation step size. The Laplace transform of $r(t)$ and the controller transfer function are obtained as

$$R(s) = (K_p + \frac{K_i}{s_{\lambda}} + K_d s^{\mu}) * E(s) \tag{15}$$

$$G_C(s) = K_p + \frac{K_i}{s^{\lambda}} + K_d s^{\mu} \tag{16}$$

4 HHO-Based Fractional PID Controller Design

To find the best tuning parameters for the controller, a good objective function need to be used. There are different objective functions were proposed for tuning of PID controller parameters like IAE, ISE, ITSE, and ITAE. Out of these functions, ITSE produces low values of settling and rise times with reduced overshoot compared to other functions [1]. To further investigate the effectiveness of the proposed algorithm, other objective functions called ITAE and ZLG [1] (ZweLee Gaing) were used. For these objective functions, HHO optimization was applied to find the optimum values of K_p, K_i, K_d, λ and μ. For each iteration of the algorithm, the error value for the output of DC motor was calculated and substituted back into the corresponding objective function. The parameter error values were adjusted according to the described procedure of the optimization algorithm. The above process is repeated until the terminating criterion was met. Then, finally, the tuned values are used to find the FOPID controller. The block diagram of the complete system was shown in Fig. 2.

4.1 Objective Function and Constraints

The equations for ITAE, ITSE, and ZLG functions were shown in (17) and (18). The proportional, integral, and derivative parameters (K_p, K_i, and K_d) were limited to the range [0, 20] and fractional powers (λ and μ) were limited to the range [0, 2].

$$ITAE = \int_0^a t \cdot |er(t)|\mathrm{d}t, ITSE = \int_0^a t \cdot er^2(t)\mathrm{d}t \tag{17}$$

$$ZLG = \left(1 - er^{-\gamma}\right) \cdot \left(O_p + ess\right) + er^{-\gamma} \cdot (t_s - t_r) \tag{18}$$

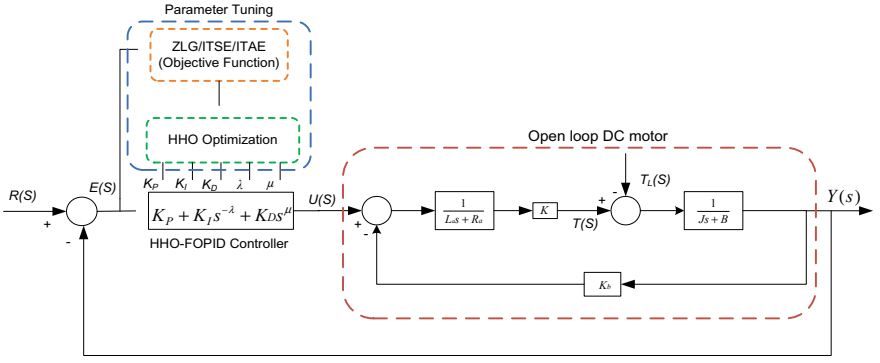


Fig. 2 Proposed system block diagram

Table 2 Minimum values of objective functions obtained

Algorithm—controller	ITSE	ZLG	ITAE
HHO—PID (Proposed)	*4.82E−05*	*0.016704*	*0.0626*
GWO—PID [4]	4.82E−05	0.016707	0.0654
SFS—PID [12]	4.84E−05	0.017032	0.0634
IWO—PID [10]	4.82E−05	0.017155	0.0985
HHO—FOPID (Proposed)	*3.18E−05*	*0.00218*	*0.0819*
GWO—FOPID [4]	*3.18E−05*	0.00256	*0.0819*
SFS—FOPID [12]	3.25E−05	0.00491	0.0843
IWO—FOPID [10]	4.26E−05	0.00361	0.0956

where ess, t_r, O_p, ess, and t_s indicates steady-state error, rise time, maximum overshoot, and settling time, respectively and the value of γ is equal to 1.0 [1] (Table 2).

5 Results and Discussion

All the simulations were performed using MATLAB/Simulink (with FOMCON toolbox) version 8.1a on the computer with Intel i5 processor @ 3.00 GHz and 8 GB RAM. For optimization algorithms, the population size is 50 and the number of iterations performed was 30 (Figs. 3, 4 and 5).

To verify the efficacy of HHO-PID and HHO-FOPID controllers, different controllers are compared by considering the same plant (DC motor). The step reactions of proposed controllers were indicated in Figs. 3 and 6. The step response comparison graph was shown in Fig. 5. The tuned parameter (K_p, K_i, K_d, λ, μ) values of various controllers were shown in Table 3. In Table 4, the proposed algorithm was compared for steady-state error, rise time, overshoot, and settling time.

The results clearly show that HHO-PID and HHO-FOPID controllers are performing better than GWO-FOPID/PID [4], SFS-PID [12], and IWO-PID [10] controllers. In the case of PID controllers, all the controllers are producing nearly the same values with very slight deviation. For FOPID controllers, the settling time was improved from 0.05406 to 0.0373, the steady-state error reduced from 3.446E−03 to 3.17E−04 whereas the rise time was slightly increased from 1.79E−02 to 1.92E−02. To further investigate, ITAE, ITSE, and ZLG values are calculated (shown in Table 2) for all the controllers and it is clear from the values that the proposed controllers have a little edge over the others. It is observed that GWO [4]-based controllers performance is very nearer to the HHO-based controllers and SWS [12] and IWO [10]-based fractional controllers are producing little overshoot. Finally, the bode plot (Fig. 4) was drawn for the HHO-FOPID and HHO-PID controllers and it shows both controllers have infinite gain margin and phase margin value of 178.9201@0.928 Hz.

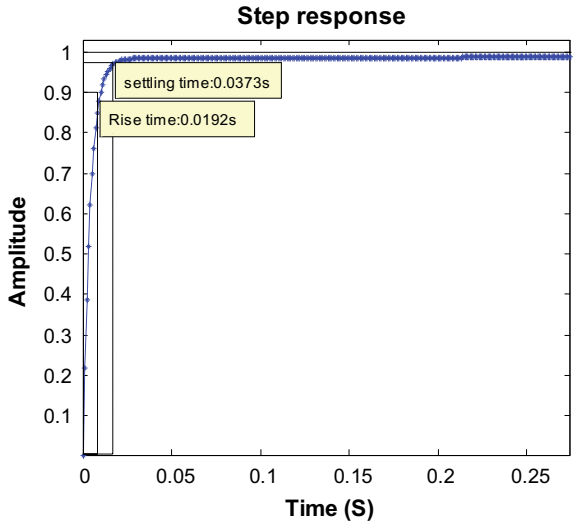


Fig. 3 HHO-FOPID controller step response

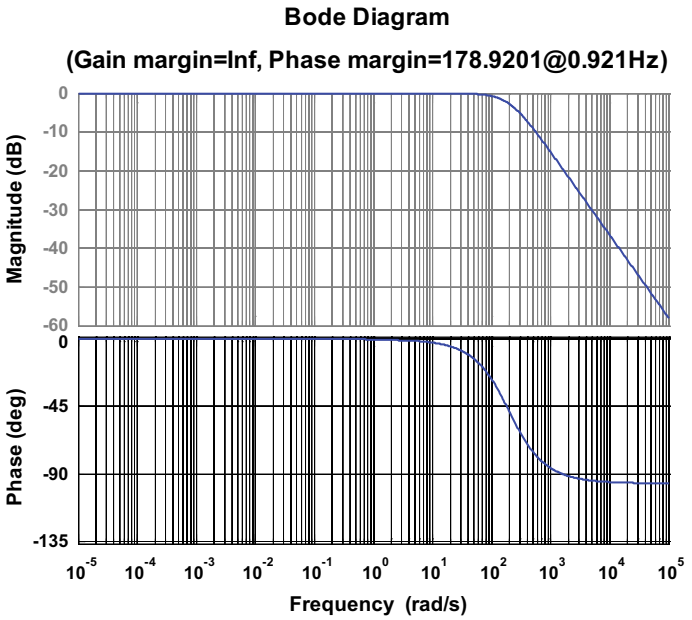


Fig. 4 Bode plot of HHO-FOPID controller

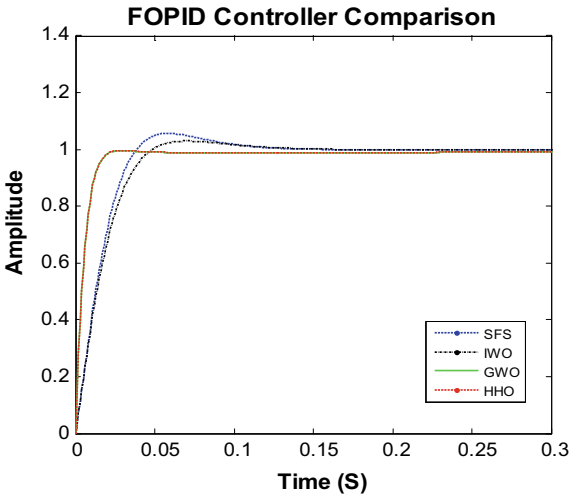


Fig. 5 Response of different FOPID controllers

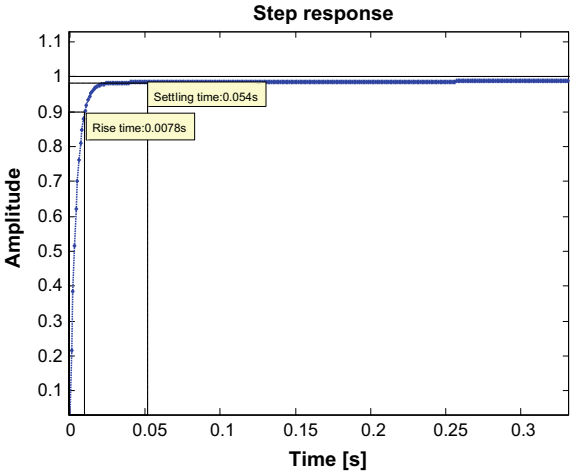


Fig. 6 HHO-PID controller step response

Table 3 Obtained values of K_p, K_i, K_d, λ and μ

Algorithm—Controller	K_p	K_i	K_d	λ	μ
HHO—PID (Proposed)	20	19.71	20	1	1
GWO—PID [4]	20	19.716	20	1	1
SFS—PID [12]	20	19.026	19.988	1	1
IWO—PID [10]	20	19.709	20	1	1
HHO—FOPID (Proposed)	20	20	20	0.508	0.931
GWO—FOPID [4]	20	20	20	0.503	0.933
SFS—FOPID [12]	20	19.896	16.193	0.402	0.980
IWO—FOPID [10]	19.842	15.020	12.802	0.431	0.989

Table 4 Comparison of parameters for various controllers

Algorithm—controller	Rise time	Settling time	Overshoot	Steady state error
HHO—PID (Proposed)	*1.79E−02*	*0.05406*	0	*3.446E−03*
GWO—PID [4]	*1.79E−02*	0.05408	0	3.448E−03
SFS—PID [12]	1.79E−02	0.0561	0	3.448E−03
IWO—PID [10]	1.79E−02	0.0550	0	*3.446E−03*
HHO—FOPID (Proposed)	*1.92E−02*	*0.0373*	0	*3.17E−04*
GWO—FOPID [4]	1.95E−02	*0.0373*	0	3.44E−04
SFS—FOPID [12]	2.08E−02	0.0456	0.0021	6.24E−04
IWO—FOPID [10]	2.43E−02	0.0586	0.0057	8.06E−04

6 Conclusion

A novel design approach, based on HHO was presented for identifying the optimum parameters of FOPID/PID controllers. The proposed method was applied to control the speed of the DC motor and results show that the algorithm was able to properly optimize the parameters of the controllers. The performance HHO-FOPID/PID controllers were evaluated using different objective functions (ITAE, ITSE, and ZLG) and compared them with existing GWO-FOPID/PID, SFS-FOPID/PID, and IWO-FOPID/PID controllers. In the future, the proposed method can be implemented on hardware platform by approximating the tuned fractional-order controller to integer order using suitable methods.

Acknowledgements This work is supported by the Department of Science and Technology, DST-ICPS division, Govt. of India under the grant number DST/ICPS/CPS-INDIVIDUAL/2018/433(G).

References

1. Z.L. Gaing, A particle swarm optimization approach for optimum design of PID controller in AVR system. IEEE Trans. Energy Convers. **19**(2), 384–391 (2004)
2. A. Khurram, H. Rehman, S. Mukhopadhyay, D. Ali, Comparative analysis of integer-order and fractional-order proportional integral speed controllers for induction motor drive systems. J. Power Electron **18**(3), 723–735 (2018)
3. C.A. Monje, Y. Chen, B.M. Vinagre, D. Xue, V. Feliu-Batlle, *Fractional-Order Systems and Controls: Fundamentals and Applications* (Springer, 2010)
4. J. Agarwal, G. Parmar, R. Gupta, A. Sikander, Analysis of grey wolf optimizer based fractional order PID controller in speed control of DC motor. icrosyst Technol. **24**(12), 4997–5006 (2018)
5. R.V. Jain, M.V. Aware, A.S. Junghare, Tuning of fractional order PID controller using particle swarm optimization technique for DC motorspeed control, in *Proceedings of 1st ICPEICES* (Delhi, India, 2016), pp. 1–4
6. A. Roy, S. Srivastava, Design of optimal *PID* controller for speed control of DC motor using constrained particle swarm optimization, in *Proceedings IEEE ICCPCT* (Nagercoil, India, 2016), pp. 1–6

7. A. Madadi, M.M. Motlagh, Optimal control of DC motor using grey wolf optimizer algorithm. Tech. J. Eng. Appl. Sci. **4**(4), 373–379 (2014)
8. U. Bhatnagar, A. Gupta, Application of grey wolf optimization in optimal control of DC motor and robustness analysis. Skit Res. J. **8**(1), 19–25 (2018)
9. A.T. El-Deen, A.A.H. Mahmoud, A.R. El-Sawi, Optimal PID tuning for DC motor speed controller based on genetic algorithm. Int. Rev. Autom. Control **8**(1), 80–85 (2015)
10. M. Khalilpuor, N. Razmjooy, H. Hosseini, P. Moallem, Optimal control of DC motor using invasive weed optimization (IWO) algorithm, in *Proceedings Maclesi Conference in Electrical Engineering* (Isfahan, Iran, 2011), pp. 1–7
11. B. Hekimoğlu, S. Ekinci, V. Demiray, R. Doguruci, A. Yildirim, Speed control of DC motor using PID controller tuned by salp swarm algorithm, in *Proceedings IENSC* (Diyarbakr, Turkey, 2018), pp. 1878–1889
12. I. Khanam, G. Parmar, Application of SFS algorithm in control of DC motor and comparative analysis, in *Proceedings of 4th UPCON* (Mathura, India, 2017), pp. 256–261
13. R.K. Achanta, V.K. Pamula, DC motor speed control using PID controller tuned by Jaya optimization algorithm, in *Proceedings ICPCSI* (Chennai, India, 2017), pp. 983–987
14. A.A. Heidari, S. Mirjalili, H. Faris, et al., Harris hawks optimization: algorithm and applications, in *Future Generation Computer Systems* (Elsevier, 2019), pp. 849–872

Artificial Intelligence-Based Robust Sliding Mode Control Technique for Plant with Input Uncertainty

Nikhil K. Yadav

1 Introduction

Practical systems in the real world are engrossed with nonlinearities and theses nonlinearities with great challenges to generate the stabilization of performance index. Nonlinear systems are facing the problems of instability as they are consisting of nonlinear characteristics such as saturation, hysteresis, friction, backlash, exponentiality, etc. in the system.

Zadeh has developed the fuzzy approach to solve the unmodelled system problems [1]. Fuzzy has been used extensively wide variety of industrial applications. Model-free approach described [2, 3]. The digital signal work has been reported in [4–8] to obtain the stability of the systems with human information systematically. Discrete-time robust sliding mode control has been described for stability in [9, 10]. Fuzzy logic is constructed from a collection of fuzzy IF–THEN rules.

Most work in fuzzy control field used the error and the change-of-error as fuzzy input variables regardless of the complexity of controlled plants in [11–14]. However, unstructured systems are unmodelled, and fuzzy formulation will be more effective for resolving problems to obtain the desired performance [15, 16]. The development of discrete-time technique by application of digital signal processing where the sampling process will be incorporated for the continuous-time systems, which provides the faster responses, fast calculations, reducing parallax error which leads to saving of time, cost factor, and facilitates for better performance. Discrete fuzzy logic controllers (DFLC) are so far the most commercially successful implementations of fuzzy logic circuits for achieving the hybrid results.

N. K. Yadav (✉)
Faculty of Electrical and Computer Engineering, Arba Minch Institute of Technology, Arba Minch University, Arba Minch, Ethiopia
e-mail: drnyadav8@gmail.com

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_12

In this work, robust intelligent sliding mode control technique has been constructed for unstructured systems. Moreover, unstructured systems realize the unpredicted input uncertainties. Theses input uncertainties create unstabilization in the system. Proposed algorithm will facilitate to obtain the stabilization of complex unstructured and engrossed input uncertainties in the systems. The paper has been formulated as below.

Problem statement described in Sect. 2, Sect. 3 demonstrates discretization of the system, Sect. 4 illustrates design of discrete sliding manifold, Sect. 5 describes the fuzzy logic controller, and Sect. 6 proposes fuzzy logic controller. Section 7 describes the simulation results, and finally, Sect. 8 leads to describe the conclusion.

2 Problem Statement and System Description

Consider the continuous-time system below

$$\dot{x} = f(x(t), u(t), t, \phi(t)) \tag{1}$$

$$\dot{x} = Ax(t) + Bu(t) + \phi(t)) \tag{2}$$

where $x(t) \in \Re^n$ and $u(t) \in \Re^m$ are continuous-time system's state and an input vector, respectively. $A \in \Re^{n\times n}$ is system matrix of the system, and $B \in \Re^{m\times 1}$ is control vector. $\phi(t)$ is input engrossed disturbance in the system. The pair $(A,\ B)$ is controllable (A, B).

3 Discretization of System

Continuous-time system is converted into discrete-time system as follows:

$$x(k+1) = f(x(k), u(k), \phi(k), k) \tag{3}$$

$$x(k+1) = Gx(k) + Fu(k) \tag{4}$$

$$\begin{aligned} x_1(k+1) &= G_{11}x_1(k) + G_{12}x_2(k) \\ x_2(k+1) &= G_{21}x_1(k) + G_{22}x_2(k) + Fu(k) \end{aligned} \tag{5}$$

where $x(k) \in \Re^n$ and $u(k) \in \Re^m$ are discrete-time system's state and input vector, respectively. $G \in \Re^{n\times n}$ discrete-time system matrix and $F \in \Re^{m\times 1}$ control vector, k is the sampling period for discretization of continuous-time system.

If the desired systems trajectory is $x_d(k)$, then error dynamics can be designed as follows:

$$e(k) = x(k) - x_d(k) \tag{6}$$

The error dynamics can be obtained as follows:

$$e_1(k+1) = G_{11}e_1(k) + G_{12}e_2(k) \tag{7}$$

$$e_2(k+1) = G_{21}e_1(k) + G_{22}e_2(k) + Fu(k) \tag{8}$$

$$e(k+1) = Ge(k) + Fu(k) \tag{9}$$

4 Designing of Discrete Sliding Manifold

Sliding motion condition for sliding manifold as follows

$$\sigma(k) \leq 0 \tag{10}$$

where $\sigma(k)$ is the sliding manifold

$$\sigma(k+1) \leq 0$$

$$\sigma(k+1) < \sigma(k) \leq 0 \tag{11}$$

$$\sigma(k) = \psi^T e(k) = 0 \tag{12}$$

where ψ is sliding manifold design parameter

$$\psi^T e(k) = \begin{bmatrix} \psi_1 & I \end{bmatrix} e(k) \tag{13}$$

$$e(k) = \begin{bmatrix} e_1(k) \\ e_2(k) \end{bmatrix} = \begin{bmatrix} e_1(k) & e_2(k) \end{bmatrix}^T \tag{14}$$

$$\sigma(k) = \psi^T e(k) = [\,\psi_1 \;\; I\,]\, e(k) \;\; = 0 \tag{15}$$

5 Fuzzy Logic Controller

Fuzzy logic technique helps to formulate the IF–THEN rules to incorporate the human-based knowledge in plant activities solution.

Plant Rule I:

$$\text{IF } \theta_1(k) \text{ is } W_{i1}(k) \text{ and } \theta_g(k) \text{ is } W_{ig}(k) \tag{16}$$

Then

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) + \xi(k) \\ x(k) = \phi(k),\ k \in \mathrm{T} \cong \{-\tau, \ldots, \infty, -1, 0\} \end{cases} \tag{17}$$

where $i \in \Phi = \{1,\ 2, \ldots,\ \tau\}$, τ is the number of IF–THEN fuzzy rules, W_{ig} is fuzzy set. $\theta_1(k)$ to $\theta_j(k)$ are premise variables.

$$\begin{aligned} x(k+1) &= \frac{\sum_{i=1}^{m} \mu_i(\theta(k))\{A_i x(k) + B_i u(k)\}}{\sum_{i=1}^{m} \mu_i(\theta(k))} \\ &= \sum_{i=1}^{m} h_i(\theta(k))\{A_i x(k) + B_i u(k)\} \end{aligned} \tag{18}$$

where $\theta(k) = (\theta_1(k)\,, \theta_2(k) \cdots \theta_j(k))$

$$\mu_i(\theta(k)) = \prod_{j}^{g} W_{ij}(\theta(k)), \tag{19}$$

$$h_i(\theta(k)) = \frac{\mu_i(\theta(k))}{\sum_{i=1}^{m} \mu_i(\theta(k))} \tag{20}$$

where $W_{ij}(\theta_j(k))$ is the grade of membership of $\theta_j(k)$ in $W_{ij}(k)$. The membership value consists greater than unity.

$$\mu_i(\theta(k)) \geq 0$$

if $i \in \Phi \sum_{i=1}^{m} \mu_i(\theta(k)) > 0$ for all k. Then, following condition holds for all k.

$h_i(\theta(k)) \geq 0$, $i \in \Phi$ and $\sum_{i=1}^{m} h_i(\theta(k)) = 1$. For convenience, it has been denoted as follows

$$\bar{A}(k) = \sum_{i=1}^{m} h_i(\theta(k))\, A_i \tag{21}$$

$$\bar{B}(k) = \sum_{i=1}^{m} h_i(\theta(k))\, B_i \tag{22}$$

$$x(k+1) = \bar{A}(k)\, x(k) + \bar{B}(k)\, u(k) \tag{23}$$

For proving the stability of the system, following Lemma has been incorporated.

Lemma 1 [10] Let $Q = Q^T$ and N, H real matrices of compatible dimension with E satisfying $\mathrm{E}^T\mathrm{E} \leq I$, then $Q + N\mathrm{E}H + H^T\mathrm{E}^T N^T \leq 0$ if and only if there exists a scalar $\varepsilon > 0$ such that $Q + \varepsilon N N^T + \varepsilon^{-1} H^T H \leq 0$.

$$\begin{bmatrix} Q & \varepsilon N & H^T \\ * & -\varepsilon I & 0 \\ * & * & \varepsilon I \end{bmatrix} \leq 0 \tag{24}$$

Lemma 2 [17] Let $a \in \Re^{n_a}$, $b \in \Re^{n_b}$ and $\aleph \in \Re^{n_a \times n_b}$ the for matrices $X \in \Re^{n_a \times n_b}$, $Y \in \Re^{n_a \times n_b}$ and $Z \in \Re^{n_a \times n_b}$ satisfying the following holds, i.e.,

$$\begin{bmatrix} X & Y \\ Y^T & Z \end{bmatrix} \geq 0$$

$$-2a^T \aleph b \leq \begin{bmatrix} a \\ b \end{bmatrix}^T \begin{bmatrix} X & Y - \aleph \\ Y^T - \aleph^T & Z \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} \tag{25}$$

A fuzzy controller with c rules is to be designed for the error $e(k)$ and the next state discrete $e(k+1)$. The jth rule can be represented of fuzzy controller in following format.

$$\begin{aligned} &\text{Rule: If } e_1(k) \text{ is } N_1^j \text{ and } e_n(k) \text{ is } N_n^j \\ &\text{THEN } y_j \text{ is } \Lambda_j^i, \;\; j = 1, 2, \ldots, p \end{aligned} \tag{26}$$

6 Design of Proposed Fuzzy Logic Controller

The input and output variables are constructed on error dynamics. The input variables for fuzzy rules are error $e(k)$ and derivative of error $e(k+1)$. Output variable is represented by $u(k)$. The input and output variables are partitioned into fuzzy sets big negative (BN), small negative (SN), small positive (SP), and big positive (BP). In the fuzzy rule base, table represents linguistic values for error $e(k)$ and derivative

of error $e(k+1)$ row and column, respectively. If $e(k)$ is big negative and $e(k+1)$ is big negative, the output $u(k)$ is big negative. After igniting, the result shows that it shows that when $e(k)$ and $e(k+1)$ are very large, then large control force required to minimize the tracking error. Hence, the tracking error must be minimum. Once tracking error reduces, the system will leads to stabilization and satisfying the performance index. All fuzzy rules will be ignited on the basis of IF–THEN conditions (Table 1).

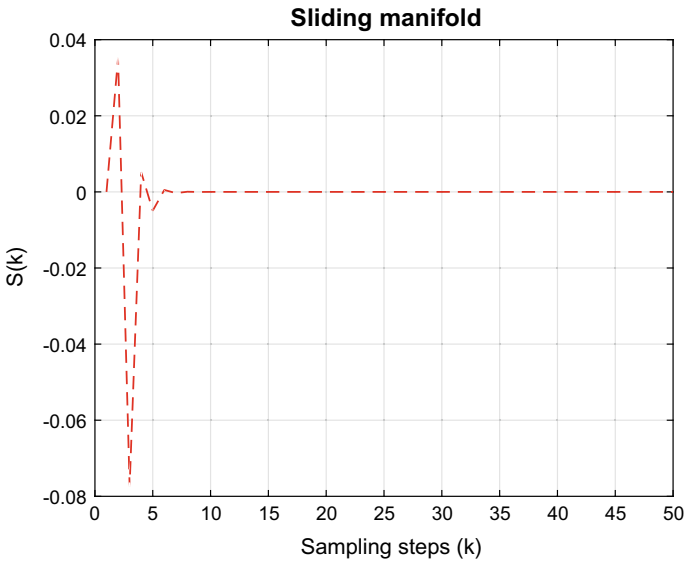


Fig. 1 Stable sliding manifold

Table 1 Rule base for input and output of discrete-time system

e(k)/e($k+1$)	BN	SN	SP	BP
BN	BN	BN	SN	SN
SN	BN	BN	SN	SP
BP	SN	SP	SP	BP
BP	SN	SP	BP	BP

7 Simulation Results

7.1 Stable Sliding Manifold

In Fig. 1, the sliding motion of sliding manifold has a deflection force of 0.0330 units magnitude in forward direction and −0.0785 units in reverse direction. Sliding manifold travelling span occurred 0.1115 units magnitudes. It indicates that the initially manifold realizes the deflection of a large value of control efforts or its rated value and subsequently it reducing the oscillation and stabilized after seven steps, i.e., 0.35 s. Sliding manifold is stabilized and remains consistently at designed characteristics. It will force the system to bring all the variables on asymptotic stable sliding manifold for all future infinite time.

7.2 Discrete-Time Error Signal

In Fig. 2, the error signal is drawn. Tracking error leads to minimize by incorporating the designed algorithm. When the control surface has a deflection of 5% from its ideal position, the maximum error signal appears to be maximum 0.180 unit magnitude and −0.172 units in opposite direction total span of error 0.352 units. Therefore, tracking error span is 0.352 unit. Reduction of tracking error needs the less amount of control efforts by control surfaces. Simulation results show that the tracking error

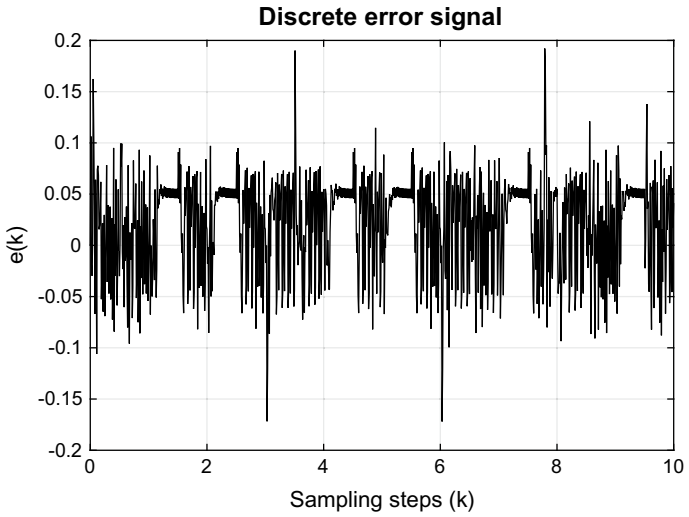


Fig. 2 Discrete-time error signal

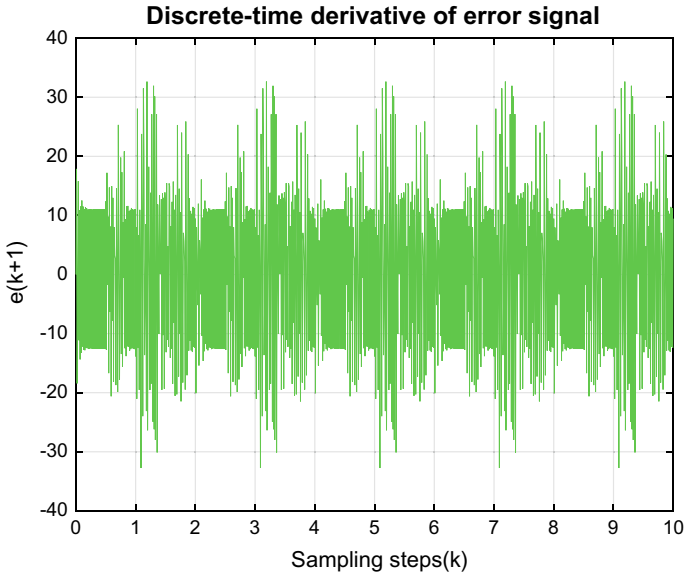


Fig. 3 Discrete-time derivative error signal

is minimized and consistent of 0.352 unit magnitude. For better stability of the system performance, the tracking should be minimized.

7.3 Discrete-Time Derivative of Error Signal

In Fig. 3, the derivative of error signal is showing the bandwidth of 66 units magnitude. When the control surface has a deflection of 5% from its ideal position, the maximum error signal appears to be 15% with a 17% change of error. When derivative error is more, the control surface required more force to reduce the error. Reducing derivative error will generate the stability of the system performance. The parametric indexes will be within the permissible limit. The derivative error is within the stabilized bandwidth of 60 units for all future time.

7.4 Output Signal

In Fig. 4, the control surface is varied by 20.0%. The tracking error is plotted, and the derivative error is realized by 17.0% from its nominal values. The control effort decreases when parameter value increases from its nominal value. The control effect is decreasing and going away from the set point. Therefore, the amount of required

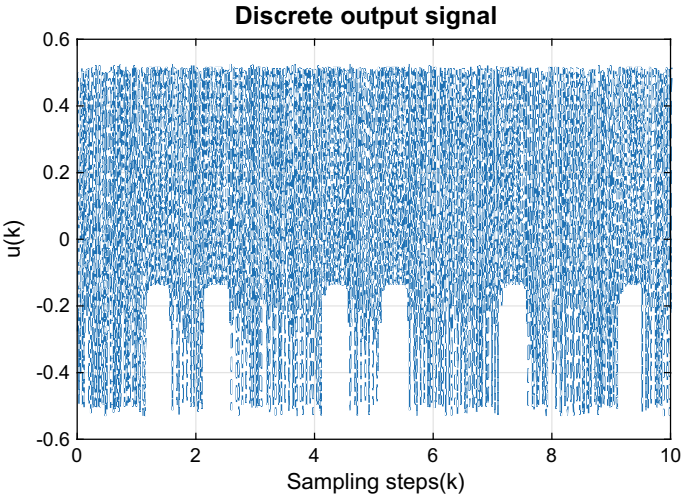


Fig. 4 Stable output response of the system

control effort is greater. The change in error signal is reduced to 12.0% from its nominal value. The tracking error is greater in the plotted simulation result. Therefore, the required control effort is increased to its value of 0.53 units of magnitude forward and backward directions. Simulation results are presenting to maintain the steady-state value consistently for all future time.

8 Conclusion

Practical systems are nonlinear in nature and for their very fast speed computation for processing by using the microcontrollers and microprocessors. Performance of system can be enhanced by discretization of continuous-time system which will profusely affirmative for achieving the better performance. Nonlinear systems are complex in nature, which are highly unstable in nature and their stability problem has become the major issues. Designed sliding manifold is stable at seven sampling steps, i.e., 0.035 s. Tracking error span is maintained by 0.353 units throughout of system operation and change of tracking errors span in the bandwidth of 60 units magnitude for all future time. Designed algorithm is showing the consistent stabilized output value of 0.53 unit magnitude in forward and backward direction by varying the system parameters 20%. Designed control technique controller will enhance the stability of the system, and this will be more useful where the system modelling is quite complex in nature whose mathematical modelling is difficult, nonavailable. Heuristic knowledge-based designed controller will be more feasible for all types of practical systems. The proposed controller is more compatible and effective to all highly nonlinear practical systems such as aviation machineries in which much

threats are engrossed, uncertain at any eventuality. Simulation results are showing stable output response of the system.

References

1. L. Zadeh, Fuzzy sets, information control **8**(1), 338–353 (1965)
2. T. Takagi, M. Sugeno, "Fuzzy identification of systems and its applications to modelling and control. IEEE Trans. Syst. Man Cybern. **SMC-15**(1), 116–132 (1985)
3. K. Tanaka, H.O. Wang, *Fuzzy Control Systems Design and Analysis: A Linear Matrix Inequality Approach* (Wiley, New York, 2001)
4. H.N. Wu, Reliable LQ fuzzy control for nonlinear discrete-time systems via LMIs. IEEE Trans. Syst. Man Cybern. B Cybern. **34**(2), 1270–1275 (2004)
5. G. Feng, Stability analysis of discrete-time fuzzy dynamic systems based on piecewise Lyapunov functions. IEEE Trans. Fuzzy Syst. **12**(1), 22–28 (2004)
6. W.J. Wang, C.H. Sun, A relaxed stability criterion for T–S fuzzy discrete systems. IEEE Trans. Syst. Man Cybern. B Cybern. **34**(5), 2155–2158 (2004)
7. S. Zhou, T. Li, Robust stabilization for delayed discrete-time fuzzy systems via basis-dependent Lyapunov-Krasovskii function. Fuzzy Sets Syst. **151**(1), 139–153 (2005)
8. M. Haifeng, J. Wu, A novel exponential reaching law of discrete-time sliding mode control. IEEE Trans. Ind. Electron. **64**(5), 3840–3850 (2017)
9. Y.S. Moon, P. Park, W.H. Kwon, Y.S. Lee, Delay-dependent robust stabilization of uncertain state-delayed systems. Int. J. Control **74**(14), 1447–1455 (2001)
10. M. Yan, Y. Shi, Robust discrete-time sliding mode control for uncertain systems with time varying state delay. IET Control Theory Appl. **2**(8), 662–674 (2008)
11. K. Tanaka, T. Hori, H.O. Wang, A multiple Lyapunov function approach to stabilization of fuzzy control systems. IEEE Trans. Fuzzy Syst. **11**(4), 582–589 (2003)
12. T.M. Guerra, L. Vermeiren, LMI-based relaxed non-quadratic stabilization conditions for nonlinear systems in the Takagi–Sugeno's form. Automatica **40**(5), 823–829 (2004)
13. P. Park, S.S. Lee, D.J. Choi, State-feedback stabilization for nonlinear time-delay systems: a new fuzzy weighting-dependent Lyapunov–Krasovskii functional approach, in *Proceedings on IEEE Conference Decision and Control* (Maui, HI, 2003), pp. 5233–5238
14. Y.Y. Cao, P.M. Frank, Stability analysis and synthesis of nonlinear time-delay systems via linear Takagi-Sugeno fuzzy models. Fuzzy Sets Syst. **124**(2), 213–229 (2001)
15. R.J. Wang, W.W. Lin, W.J. Wang, Stability of linear quadratic state feedback for uncertain fuzzy time-delay system. IEEE Trans. Syst. Man Cybern. B Cybern. **34**(2), 1288–1292 (2004)
16. Z. Yi, P.A. Heng, Stability of fuzzy control systems with bounded uncertain delays. IEEE Trans. Fuzzy Syst. **10**(1), 92–97 (2002)
17. Y.Y. Cao, P.M. Frank, Analysis and synthesis of nonlinear time-delay systems via fuzzy control approach. IEEE Trans. Fuzzy Syst. **8**(2), 200–211 (2000)

Designing of Fractional-Order PI/PID Controller by Meta-heuristic Algorithm Using PSO for PEM Fuel Cell

Govind Anil, M. Siva, S. Srinath, and G. Uday Bhaskar Babu

1 Introduction

Consumption of fossil fuels is increasing with the increase in demand of energy. An alternate source of energy is required to fulfill the energy demands. Hydrogen technology can be an option for reducing the carbon emission in the energy generation and can be easily produced from various sources. Proton exchange membrane (PEM) fuel cell [1] devices use hydrogen and oxygen as fuel with no harmful by-product being generated and can control the emission of greenhouse gases. The products of the reaction are water, heat, and electricity. At anode end, hydrogen disintegrates into hydrogen ions and electrons through catalyst. Membrane allows only ions to pass, and electrons moves through external circuit to cathode end. Cathode recombines hydrogen ions with oxygen to form water. PEM fuel cells are focused more toward transport or vehicle applications, stationary, and portable application as the startup time of the system is very less and the response is quick with sudden change in operating conditions. Other factors are operating temperature of PEM fuel cells are low, i.e., below 100 °C. Creating high efficiency PEM fuel cells requires proper fuel supply, heat management, water generated from the reaction and partial pressure of the reactants to ensure that the fuel cell system runs consistently.

The control of the stack voltage is required to operate it at a set point as numerous factors affect output voltage such as a sudden increase in current demand, oxygen starvation, cathode flooding, humidification of reactant gases, hotspots, and dryness of the membrane; an optimization-based controller is designed to maintain the stack voltage to a set point with changes in the operating parameters. Optimal controller

G. Anil · M. Siva · S. Srinath · G. Uday Bhaskar Babu (✉)
National Institute of Technology, Warangal, Telangana 506004, India
e-mail: udaybhaskar@nitw.ac.in

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_13

parameters are obtained by optimizing the tuning parameters to get optimal control for the stack system.

2 Model Description

2.1 Plant Model

A voltage control of PEMFC stack model is considered using MATLAB with the subspace system identification, and different models are identified by varying the stack current for the stack. Due to the nonlinear behavior of the fuel cell system, different stack currents are considered, and system shows different behavior as other parameters are influenced such as temperature inside stack, water generation due to the reaction and active area, humidification of the reactant gases, and many others. So from the different models, the best model is taken by considering the gap metric developed for the system at various current values [3]. The air flow rate is manipulated by giving chirp signal as an input, and the output is taken as stack voltage. The basic block diagram of the model is shown in Fig. 1.

The plant model is defined as

$$G_1(s) = (-0.102s - 8.314)/s^2 + 13.09s + 18.58 \tag{1}$$

$$G_2(s) = (0.263s - 2.05)/s2 + 13.32s + 74.77 \tag{2}$$

Above transfer function has output as the stack voltage and input as the air flow rate (oxygen) and hydrogen flow rate, respectively. By manipulating the supply in the stack, the output voltage of the stack is maintained. Flow rate affects the overall reaction of the cell which in turns affect the voltage generated, and for application in real-time system, a constant power requirement is preferred [7].

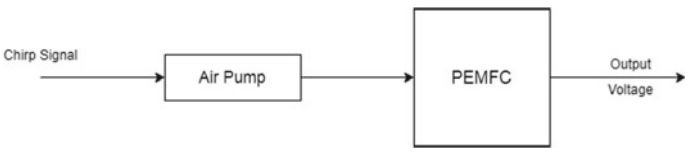


Fig. 1 Block of PEMFC system

2.2 PSO Algorithm

An algorithm is an iterative methodology, and the results are generated through repeated pattern by evaluation. Particle swarm optimization [4] is based on bird flocking behavior and swarm intelligence, where a swarm of particles is generated, and each particle represents a potential solution. Each particle gains knowledge from other members in the swarm. PSO is random search optimization technique, as the algorithm progresses the particles taht are moved toward an optimized value for the objective function. PSO has both social and computational behavior as they can interact with the nearby particles to define their course of action. The algorithm is continued until all constrains are satisfied or maximum iterations are reached. The initial population is generated at random, and further speed of and position of the particles are updated as

$$V_i = W \cdot V_{i-1} + C_1 \cdot \text{rand}_1 \cdot (P_i - X_i) + C_2 \cdot \text{rand}_2 \cdot \left(P_g - X_i\right) \tag{3}$$

$$X_{i+1} = X_i + V_i \tag{4}$$

where V_i is the velocity of the ith particle, W is the weighing factor, C_1 and C_2 are cognitive and social learning rates of particle, P_i is the best individual position, P_g is the best position of all particles, and rand_1 and rand_2 are random numbers between (0, 1).

The time-varying inertia weight technique [2] is used to reduce the search space and velocity as the number of iteration are increased. The inertia weight is updated as

$$W = [(I_{\max} - I)/I_{\max}] \cdot (W_{\max} - W_{\min}) + W_{\min} \tag{5}$$

where $I_{\max}$ is maximum iterations allowed, I is current iteration, $W_{\max}$ and $W_{\min}$ are initial and final values, respectively. The approach can also be adaptive inertia weight or random inertia weight. The weighing factor is decreased as the iterations are increased so as to reduce the velocity of the subsequent particles. It helps to reduce the search space as the function is more toward its optimal value (Fig. 2).

2.3 Fractional-Order Controller

In calculus, we learn to calculate derivatives and anti-derivatives (which can be related to the integrals), and as real valued function can be iterated, we can calculate multiple orders for those function to get a solution. From equation involving fractional derivatives, real-world system can be modeled, and control of those system are

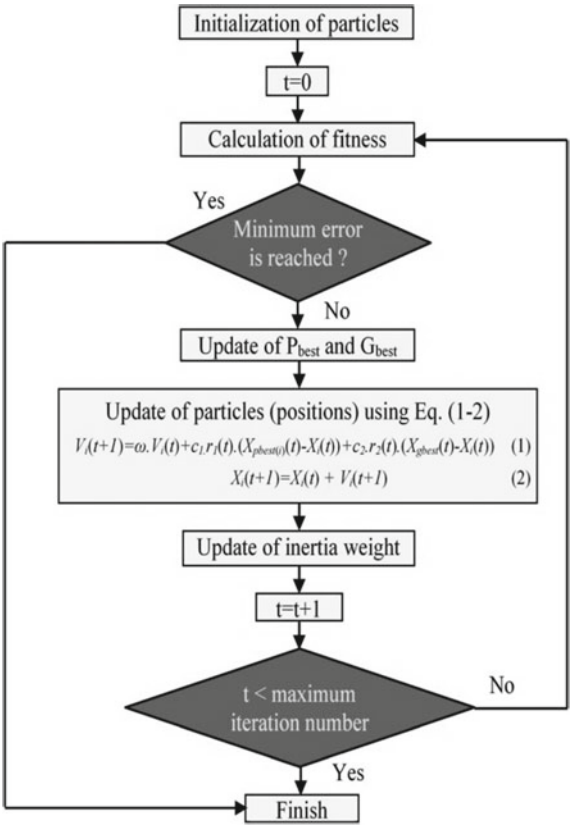


Fig. 2 Flowchart of PSO algorithm

required. In fraction control, we can control an integer-order plant with a fractional-order controller and a fractional-order plant with an integer-order controller. A basic fractional-order controller [5] is designed as

$$G_c(s) = K_p + K_i s^{-\lambda} + K_d s^{\mu} \tag{6}$$

In the fractional-order controller, K_p, K_i and K_d are the proportional integral and derivative gain of the controller, respectively. The controller designed using PSO-based optimization technique with five parameters is to be optimized for a fractional-order controller. A $PI^{\lambda}D^{\mu}$ controller is designed where powers of integral and derivative action are non-integer real numbers. So, we can say that PID controller is a special case of fractional-order controller where integral and derivative action has the power of unity.

2.4 Methodology

The voltage control is achieved by manipulating the oxygen inlet rate. An integer- and fractional-order PID controllers [6] are designed by optimization-based controller by taking minimization of IAE as objective function. For designing a fractional-order controller, five parameters need to be optimized where the integral and differential action power are non-integer. As the sudden change in the demand can affect the efficient working of the system, control is required for the PEM fuel cell system. The voltage is taken as the controlled variable for the controller design. As PSO is an iterative technique, the steps involved in designing the controller are

I. Initialize the tuning parameters of integer- and fractional-order PI/PID controller, i.e., proportional, integral, and derivative gain.
II. Evaluate the IAE value for each particle generated from initialization taken as the objective function.
III. Set the global best position from evaluation of objective function.
IV. Initialize the inertia weight for updating the algorithm.
V. For every particle, evaluate velocity using global best and individual best position of particle.
VI. Evaluate the objective function and update the global best position.
VII. Repeat steps IV–VI until termination criteria is satisfied.
VIII. Stop and state the best parameters.

The controller tuning parameters are obtained by executing above steps involved in PSO.

3 Simulation Results

The model defined above with air flow rate as the manipulated variable different simulations are done by using optimal values of simple PID and fractional-order PID controller. Performance for the system is measured using the integral of absolute value of error (IAE) and total variation (TV) for each controller. The controller parameters for a FOPID controller are set in the range of (−60, −80, −60, 0, 0) to (60, 80, 60, 2, 2). As from the table, comparing the IAE and TV values for both the model, we can conclude that in FOPID controller, the system response is improved, and the rise time for the system is reduced. A disturbance to the system is given at a time instant of 5 s for each controller, and the response of the system is analyzed (Table 1).

By comparing the IAE values of the system, we can see that FOPID have smaller value compared to other controllers, and the TV values are also reduced for the system (Fig. 3).

The comparison of PI, PID, FOPI, and FOPID based on particle swarm optimization for the control of voltage is obtained by the fuel cell. While considering different controller parameters, other parameters which are not involved in the designing the optimal controller parameters are not taken into consideration.

Table 1 IAE and TV values of various controllers for the PEMFC system

Controllers	IAE	TV
PI	0.7544	556.5
PID	0.4755	506.28
PSO-PID	0.5201	496.66
PSO-FOPID	0.2586	330.82

Table 2 IAE values of PID and FOPID for hydrogen flow rate as manipulated variable

Controllers	IAE
PID	2.29
FOPID	1.98

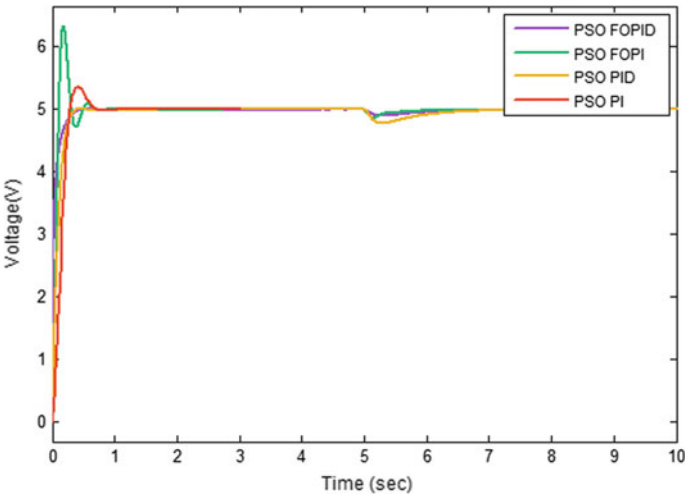


Fig. 3 Response of the system of air flow rate as manipulated variable with disturbance rejection

System described with the manipulating variable as hydrogen flow rate is controlled at a set point. Disturbance to the system is given at a time instant of 5 s, and the response of the system is measured.

The IAE values are compared as shown in table, and the response for the system is described as shown in Fig. 4. The range of controller parameters are lying in the range of (−100, −100, −100, 0, 0) to (100, 100, 100, 2, 2).

4 Conclusion

This chapter designed the optimal controller for a PEMFC system by using the PSO algorithm defined above. A comparison between fractional- and integer-order

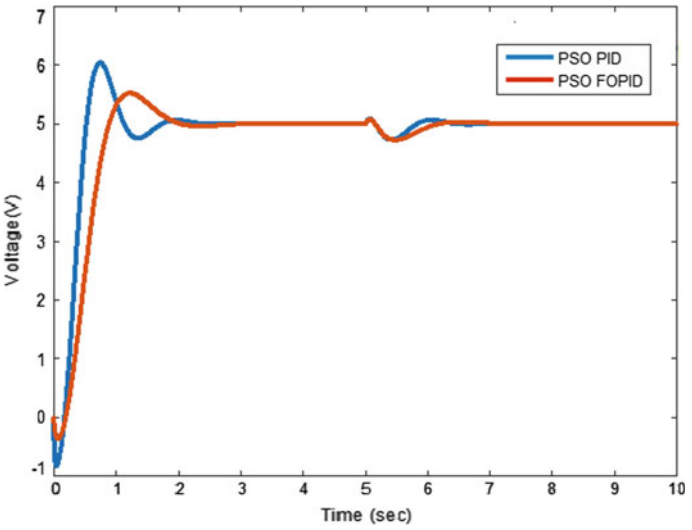


Fig. 4 Response of the system of hydrogen flow rate as manipulated variable with disturbance rejection

controller is commenced for a second-order model system by implementing a time-varying inertia weight technique. Simulation results using designed controllers displays the minimization of error for the closed-loop system and maintaining the stack voltage to desired level swiftly.

Reference

1. F. Barbir, *PEM Fuel Cells: Theory and Practice* (Academic Press, 2012)
2. A. Nickabadi, M.M. Ebadzadeh, R. Safabakhsh, A novel particle swarm optimization algorithm with adaptive inertia weight. Appl. Soft Comput. **11**(4), 3658–3670 (2011)
3. F.-C. Wang, H.-T. Chen, Y.-P. Yang, H.-P. Chang, J.-Y. Yen, Multivariable System identification and robust control of a proton exchange membrane fuel cell system, in *46th IEEE Conference on Decision and Control* (New Orleans, LA, USA, 2007)
4. R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in *Proceedings 6th International Symposium Micro Machine and Human Science (MHS)* (1995), pp. 39–43
5. R. El-Khazali, Fractional order $PI^{\lambda}D^{\mu}$ controller design. Comput. Math. Appl. **66**(5), 639–646 (2013)
6. Y. Qi, M. Thern, M. Espinoza-Andaluz, M. Andersson, Modeling and control strategies of proton exchange membrane fuel cells. Energy Proc. **159**, 54–59 (2019)
7. R. Salim, M. Nabag, H. Noura, A. Fardoun, The parameter identification of the Nexa 1.2 kW PEMFC's model using particle swarm optimization. Renew. Energy **82**, 26–34, 20

Fuzzy Logic Theory-Based PI Controller Tuning for Improved Control of Liquid Level System

K. Sandeep Rao, V. Bharath Kumar, V. N. Siva Praneeth, and Y. V. Pavan Kumar

1 Introduction

Controlling of liquid level is important in many industries such as chemical, water treatment, and pharmaceutical industries. The control of liquid level system (LLS) is done by using different types of classical controllers [1]. Of all the available classical controllers, PID controller is most commonly used. The main demerit associated with these types of controllers is their offline tuning. So, these cannot address the practical problems that occur in the system. Further, selection of the methods for tuning PID controller is a tedious job as the methods' applicability changes from system to system. These aforesaid problems can be addressed by using the artificial intelligence methods such as artificial neural networks (ANN), fuzzy logic theory, etc., [2]. For controlling the LLS, ANN controller was designed, and this controller gives the better system response when compared to the classical PID controller [3]. However, use of ANN controller leads to more peak overshoot and rise time in the system response [4]. There may be many reasons for these issues, such as insufficient data points and choosing of suitable algorithm for training the data. By taking all these points into consideration, this paper proposes an intelligent online controller

K. Sandeep Rao · V. Bharath Kumar · V. N. Siva Praneeth · Y. V. Pavan Kumar (✉)
Vellore Institute of Technology - Andhra Pradesh (VIT-AP) University, Amaravati 522237, Andhra Pradesh, India
e-mail: pavankumar.yv@vitap.ac.in

K. Sandeep Rao
e-mail: sandeep.18bec7094@vitap.ac.in

V. Bharath Kumar
e-mail: bharath.18bec7093@vitap.ac.in; pavankumar.yv@vitap.ac.in

V. N. Siva Praneeth
e-mail: praneeth.18bev7022@vitap.ac.in

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_14

that continuously updates the gain parameters of the PI controller according to the disturbance. This proposed controller uses the fuzzy logic theory in obtaining the gain parameters.

2 Description of the LLS with Conventional Controllers

As mentioned in [4], the transfer function of the LLS is considered as (1).

$$G(s) = \frac{0.315}{12.826s + 1} e^{-8.415s} \tag{1}$$

Among the different classical methods that are available as shown in Fig. 1, OLTR and EPI methods are very basic and simple methods that are available for the tuning of the PID gains [5, 6]. However, the major limitation of these methods is that the consideration of open-loop response for computing the gain parameters, which may not cope up when kept in closed-loop operation. To overcome this, UC, DT, and OU methods have been evolved, where closed-loop response is considered for the computation of PID gain parameters, thereby provides better closed-loop response when compared to OLTR methods. However, the limitation of UC methods is obtaining the sustained oscillations for calculating the gain parameters, as all systems cannot produce the sustained oscillations every time. Further, DT and OU methods are simple to implement, but they cannot cater all the needs of classical PID controller.

Another way of controller parameters' design is through ANN theory. This method is online design method and can intelligently operate the system according to the

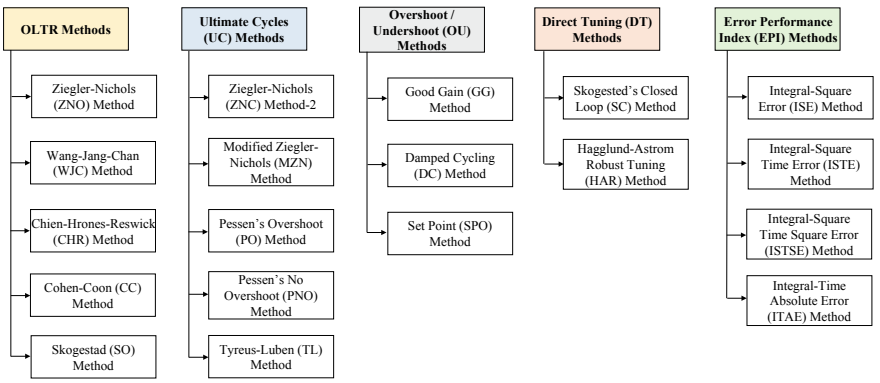


Fig. 1 Different PID tuning methods and their classifications

response and the disturbances. So, this can address all varieties of the practical disturbances that occur in the system. This controller gives far better output than the classical PID controller. However, the limitations of this controller are picking of appropriate algorithm for training the data as the efficiency of the algorithm changes from system to system and selecting the number of hidden layers and training validation percentages according to the system.

3 Proposed Fuzzy Logic-Based PI Controller in Presence of the External Disturbance

Fuzzy logic theory has been proved as useful to deal with systems that are nonlinear and time-variant [7]. The considered case study in this paper, i.e., LLS is a nonlinear system, whose liquid level has to be regulated smoothly under all external disturbances or inputs. So, the fuzzy logic theory-based PI controller is implemented in this paper to achieve this specific regulation. According to the system transfer function and disturbance, the fuzzy logic estimates the PI gain parameter values and updates in the system accordingly. This fuzzy logic-based controller interprets entire input/output values that are fed to it as in between 0 and 1. The development of fuzzy interface system (FIS), which maps the input and output is given as follows [8, 9].

The fuzzy logic controller converts the crisp values (0 or 1) into the fuzzy values (varies from 0 to 1). For the fuzzification process, normally triangular, signum, trapezoidal, etc., membership functions (MFs) are used, and for the fuzzification process, two traditional methods, namely mamdani and sugeno, can be used [10–12]. Once, the processing is done through FIS, defuzzification takes place at the output stage. Defuzzification is the process of converting the fuzzy set values back to the crisp values. It is just the opposite action of the fuzzification and can be done using various methods such as, first of maxima method, mean of maxima method, center of sums method, etc. The input and output membership functions are mapped using various rules. Here comes the intelligence of the fuzzy logic controller, which replicates the human brain. These rules are same as the ideas given by the brain [13].

This chapter proposes the fuzzy-PI controller (FPI), which retains the advantages of both the fuzzy logic theory and the classical PI controller [14, 15]. Proposed FPI is developed with triangular input/output membership functions with mamdani method. The range of each membership functions is given in Table 1. These ranges are identified based on the nature of system response under a real-time disturbance shown in Fig. 2. Similarly, the input/output mapping rules that are developed in the proposed case are given in Table 2. The complete model of LLS with FPI is shown in Fig. 3.

Table 1 Range of each membership function for constructing the FIS

Membership functions		Ranges	
Input variable	Deviation	PPV	[0.03 0.04 0.05]
		MPV	[0.01 0.02 0.04]
		MinPV	[0 0.01 0.02]
		MinNV	[−0.02 −0.01 0]
		MNV	[−0.04 −0.02 −0.01]
		PNV	[−0.05 −0.04 0.03]
Output variable	Kp	KpMF1	[0 0.025 0.05]
		KpMF2	[0.07 0.12 0.155]
		KpMF3	[0.115 0.170 0.23]
		KpMF4	[0.21 0.22 0.23]
	Ki	KIMF1	[0.25 0.5 0.75]
		KIMF2	[0.75 0.85 1.7]
		KIMF3	[0.85 1.5 2.2]
		K_IMF4	[1.74 2 2.6]
		K_IMF5	[2.55 2.614 2.615]

where *PPV* peak positive value, *MPV* moderate positive value, *MinPV* minimum positive value, *MinNV* minimum negative value, *MNV* moderate negative value, *PNV* peak negative value

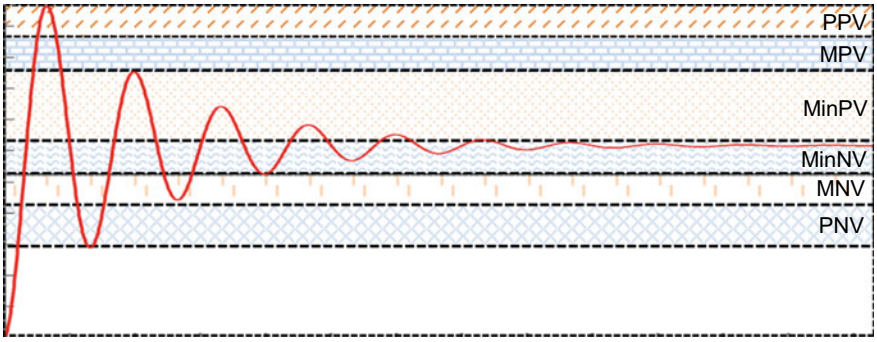


Fig. 2 Response of the system when the disturbance is injected

4 Results and Analysis

The aim of the proposed FPI controller is to obtain the better response when compared to classical PID and ANN-PID controller. For evaluating the performance of the LLS with the proposed FPI controller, its response is compared with the responses obtained with all these existing methods. The comparative analysis was performed by

Table 2 Rules for constructing the FIS

	K_P	K_I
PPV	K_pMF4	K_IMF4
MPV	K_pMF1	K_IMF1
MinPV	K_pMF3	K_IMF2
MinNV	K_pMF3	K_IMF3
MNV	K_pMF4	K_IMF4
PNV	K_pMF4	K_IMF5

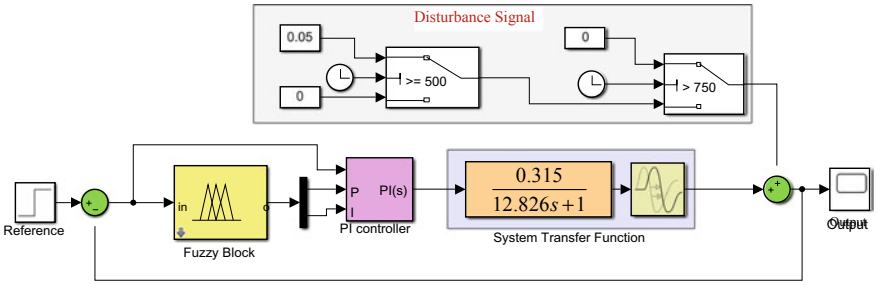


Fig. 3 Simulation model of the LLS with proposed FPI controller

injecting a disturbance during steady state. The respective responses are shown from Figs. 4, 5, 6, 7, 8, and 9, and cumulative quantitative comparison is given in Table 3. The dotted circles in each figure represents deviation in responses due to the external disturbance with a magnitude of 5% of the reference input that is applied during 500–750 s. Figures 4 and 5 show the system response with OU and DT methods, respectively. Figures 6, 7, and 8 show the system response with OLTR, UC, and EPI methods, respectively, where (a) of each figure gives the overall response, (b) of each figure gives response of the system at disturbance initiation (ON), and (c) of each

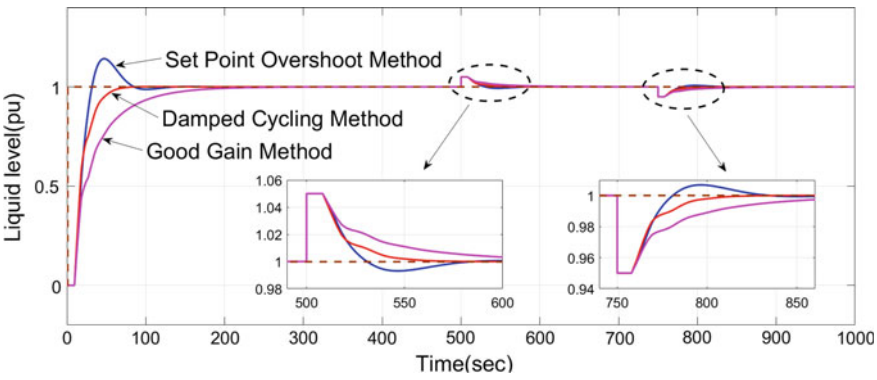


Fig. 4 Time domain response of the system with OU methods

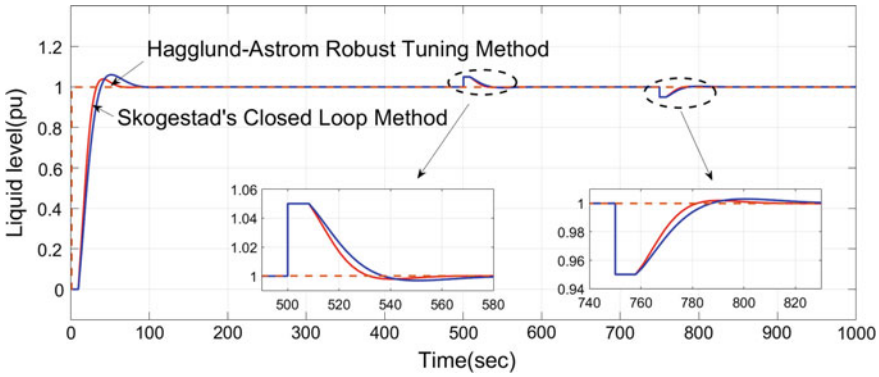


Fig. 5 Time domain response of the system with DT methods

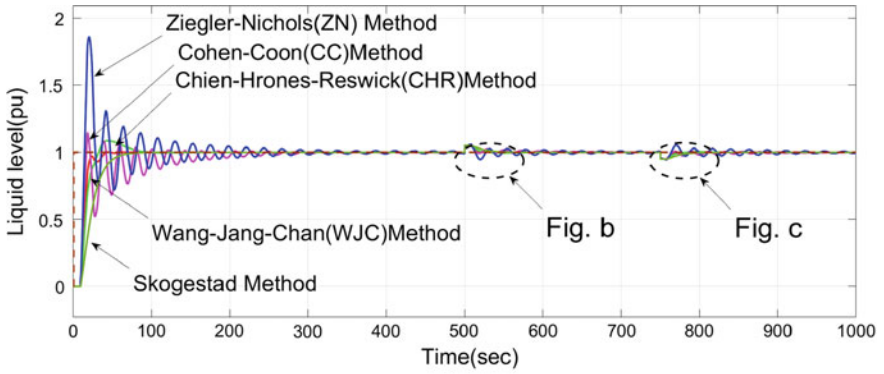
figure gives response of the system at disturbance closure (OFF). Figure 9 shows the system response with proposed FPI controller and conventional ANN-PID controller. From these results, a better method is identified, which produces a response with less peak overshoot, settling time, delay time, and rise time as computed in Table 3.

5 Conclusion

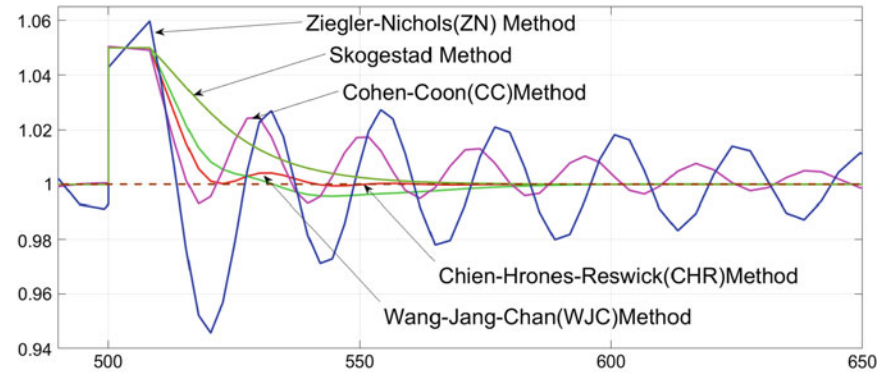
The drawbacks with the classical PID and ANN-PID controller for controlling the liquid level, such as the more transient time, no appropriate reaction to disturbance, and steady-state error are addressed in this chapter with the proposed FPI controller. From the cumulative comparison given in Table 3, the achievements of the proposed FPI are observed as follows.

- It reduces the peak overshoot, transient time, and oscillatory response of the system.
- Besides, the conventional cascaded ANN-PID controller uses training of PID controller, which needs to train all the three controller gains, K_p, K_i, K_d. But, the proposed method uses only PI controller for the same system. So, the implementation cost is reduced as well as processing speed can be improved.

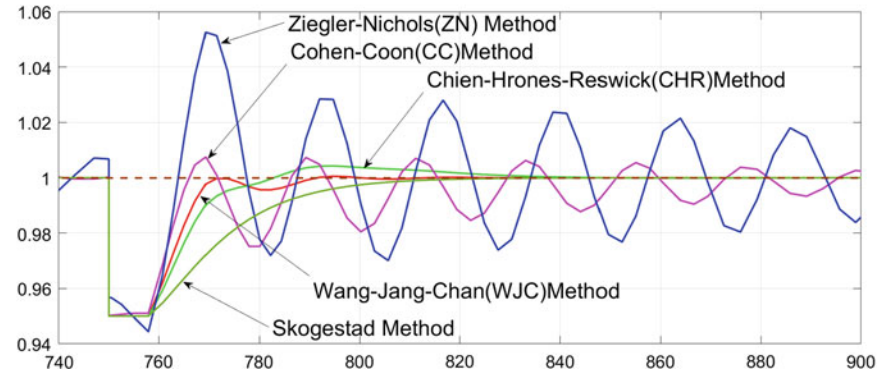
Hence, by considering all the above points, it can be concluded that the proposed FPI controller is a better choice for the control of practical LLSs.



(a) Full response of the system

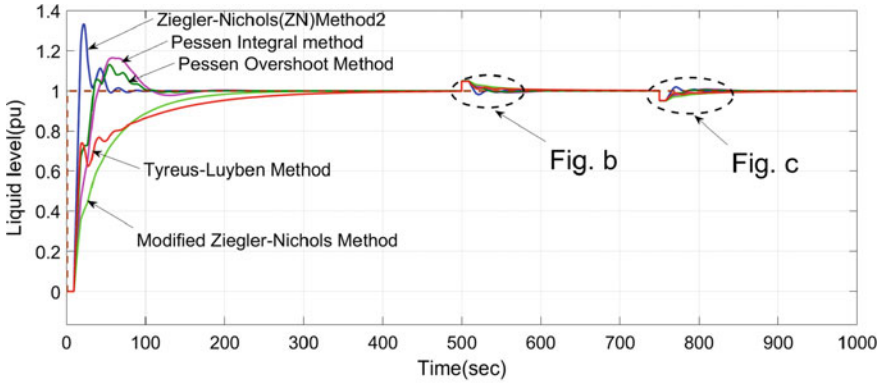


(b) Response of the system at disturbance initiation

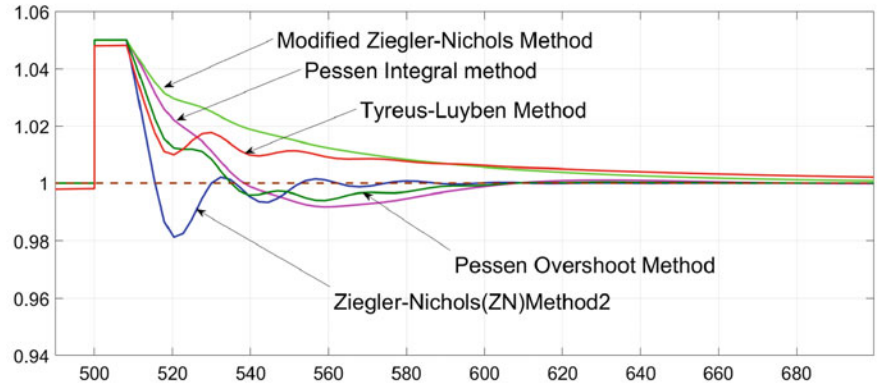


(c) Response of the system at disturbance closure

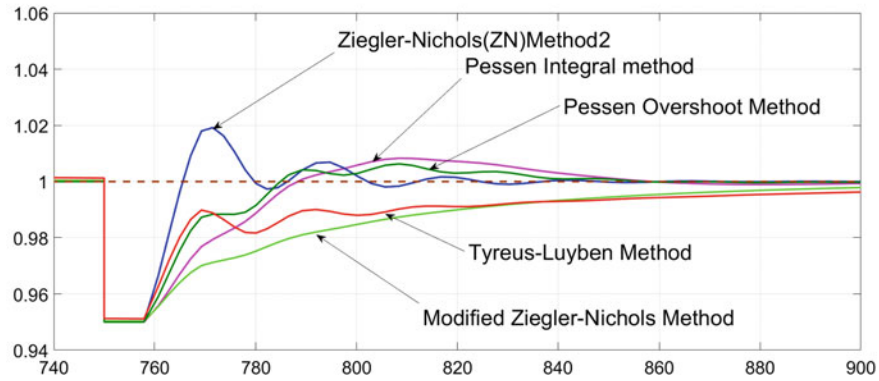
Fig. 6 Time domain response of the system with OLTR methods



(a) Full response of the system

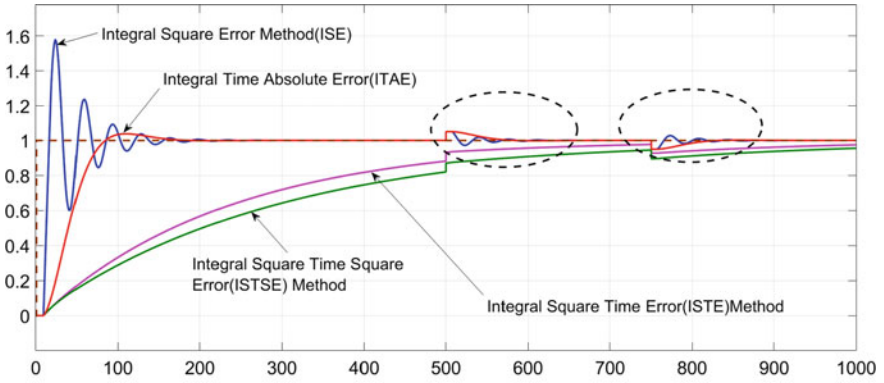


(b) Response of the system at disturbance initiation

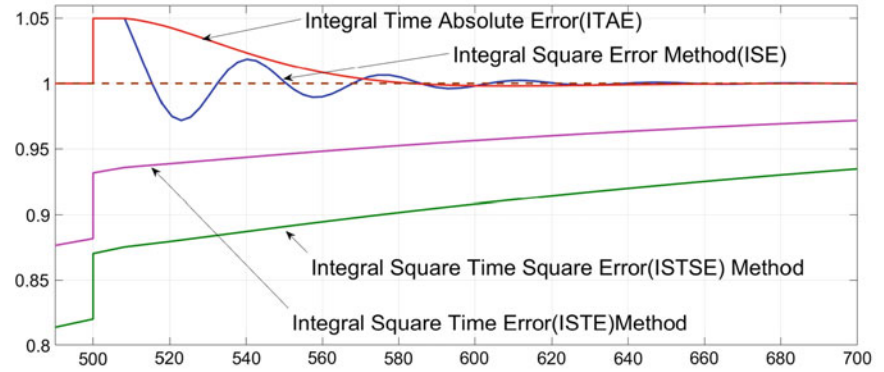


(c) Response of the system at disturbance closure

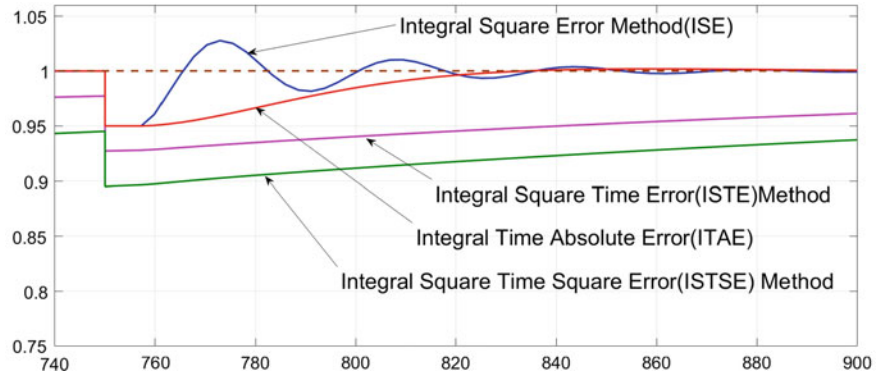
Fig. 7 Time domain response of the system with UC methods



(a) Full response of the system



(b) Response of the system at disturbance initiation



(c) Response of the system at disturbance closure

Fig. 8 Time domain response of the system with EPI methods

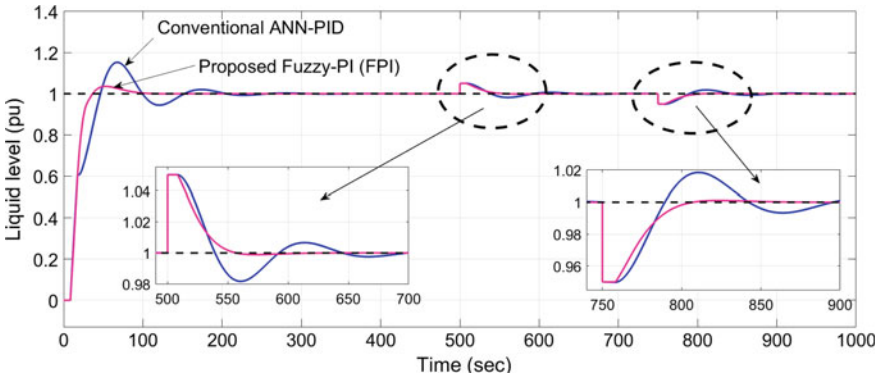


Fig. 9 System response with proposed FPI controller and conventional ANN-PID controller

Table 3 Conventional versus proposed method comparison using various performance metrics

Controller design method			Delay time	Rise time	Settling time	Peak overshoot (%)
Conventional OLTR methods		WJC	14.5	9.2	39.3	0
		CHR	15.6	15.1	72	8.5
		CC	12.7	5.8	216	14
		ZNO	12	3.8	222.5	86.7
		SO	21.8	31.8	57.6	0
Conventional UC methods		ZNC	12.93	5.7	50.1	33
		MZN	30.5	95.3	170.3	0
		PO	15.88	21.02	92	13
		PNO	19.8	22.9	98.07	16
		TL	14.89	119.62	243	0
Conventional OU methods		DC	16.8	27.55	53.75	0
		GG	22	71.4	135	0
		SPO	17.62	16.46	74.5	14
Conventional DT methods		SC	17.85	16.2	47.4	3.9
		HAR	20	20.27	68.9	6
Conventional EPI methods		ISE	12.84	5.68	45.4	57.8
		ITAE	39.6	49.45	51.2	37.6
		ISTSE	166.8	507.4	284.85	0
		ISTE	202.6	637.45	351.00	0
AI methods	Conventional	ANN	16.91	15.51	64.3	3.56
	Proposed	FPI	16.47	13.85	60.24	0

References

1. D.K. Sambariya, R. Prasad, D. Birla, Design and performance analysis of PID based controller for SMIB power system using firefly algorithm, in *International Conference on Recent Advances in Engineering & Computational Sciences* (Chandigarh, India, 2015)
2. A.V.K. Shankar, U. Subramaniam, S.V. Padmanaban, Investigation for performances comparison PI, adaptive PI, fuzzy speed control induction motor for centrifugal pumping application, in *IEEE 13th International Conference on Compatibility, Power Electronics and Power Engineering* (Sonderborg, Denmark, 2019)
3. S.B. Prusty, S. Padhee, U.C. Pati, K.M. Kamala, Comparative performance analysis of various tuning methods in the design of pid controller, in *Michael faraday IET International Summit: MFIIS-2015* (Kolkata, India, 2015)
4. B.V. Murthy, Y.V. Kumar, U.V.R. Kumari, Application of neural networks in process control: automatic/online tuning of PID controller gains for $\pm 10\%$ disturbance rejection, in *IEEE International Conference on Advanced Communication Control and Computing Technologies (ICACCCT)* (Ramanathapuram, India, 2012)
5. P.M. Meshram, G.K. Rohit, Tuning of PID controller using ziegler-nichols method for speed control of dc motor, in *IEEE International Conference on Advances in Engineering, Science and Management* (Nagapattinam, Tamil Nadu, India, 2012)
6. J.M. Lin, P.F. Kao, K.T. Cho, *Ziegler-Nichols Based Intelligent Controller Design of a SPM System* (Xiamen, China, 2012)
7. H.J. Live, H.B. Wang, X.M. Zhu, Z.H. Shen, J.Y. Chen, Simulation research of fuzzy autotuning PID controller based on matlab, in *International Conference on Computer Technology, Electronics and Communication* (Dalian, China, 2017)
8. R. Manikandan, A. Arulprakash, R. Arulmozhiyal, Design of equivalent fuzzy PID controller from the conventional PID controller, in *International Conference on Control, Instrumentation, Communication and Computational Technologies* (Kumaracoil, India, 2015)
9. A.K.A. Mohd, N.A. Ab Hadi, I.M. Saadon, M.H. Harun, Design and construction of liquid level measurement system. J. Adv. Res. Appl. Mech. **12**(1), 8–15 (2015)
10. I.M. Boiko, Application of second-order sliding-mode control algorithms in continuous cycling tests for PID tuning, in *IEEE European Control Conference* (Strasbourg, France, 2014)
11. Y.V.P. Kumar, K.M.N.S. Kiran, S. Yugandhar, K. Padma Raju, Online attitude controlling of longitudinal autopilot for general aviation aircraft using artificial neural networks, in *Nirma University International Conference on Engineering* (Ahmedabad, India, 2013)
12. H. Yu, C. Yang, Q. Zhao, Division of working variable domain fuzzy PID control for valley electric phase change heat storage, in *IEEE Innovative Smart Grid Technologies—Asia* (Chengdu, China, 2019)
13. D.E.P. Lebukan, A.N.I. Wardana, N. Effendy, *Implementation of Plant-Wide PI-Fuzzy Controller in Tennessee Eastman Process* (International Seminar on Application for Technology of Information and Communication, Semarang, Indonesia, 2019)
14. X. Xu, Q. Wang, Speed control of hydraulic elevator by using PID controller and self-tuning fuzzy PID controller, in *32nd Youth Academic Annual Conference of Chinese Association of Automation* (Hefei, China, 2017)
15. R. Saptarshi, M. Jayabrata, Fuzzy logic controlled cuk converter, in *International Conference on Communication and Signal Processing* (Chennai, India, 2018)

Modeling and Optimization of Surface Roughness in Machining of Brass Using Multi-linear Regression in Conjunction With Genetic Algorithm

Suhail Manroo and Suhail Ganiny

1 Introduction

Surface texture is often considered as a composite property as it is characterized by a multitude of factors, that include roughness, lay, and waviness of the surface, with surface roughness being the most prominent. Several investigations have shown that surface roughness influences the wear behavior, frictional properties, fatigue and fracture characteristics, and resistance that a component offers to corrosion [1, 2]. Besides these, other attributes like geometrical tolerances, lubrication effectiveness, electrical/thermal conduction, and optical properties of a manufactured component are also closely related to its surface roughness. It is widely acknowledged, that a rough surface wears off easily, displays high friction when in relative motion against other surfaces, and has a reduced fatigue life than a smooth one. High surface roughness, however, is not an evil always, and in certain practical situations, rougher surfaces may be even desirable. For instance, high surface roughness leads to higher oxygen-catalytic recombination and heat transfer rates in thermal protection systems (TPS) employed in hypersonic vehicles [3]. Furthermore, rough surfaces possess better adhesion properties and galling resistance [4, 5].

Ideally, the surface roughness upto which a component can be machined or fabricated is determined by the manufacturing technique employed for its production. While the techniques like grinding, honing, and superfinishing produce surfaces with 0.01–5 μm surface finish, however, others like sand casting, rolling, and forging yield surfaces with relatively poor surface finish (5–50 μm) [6]. A trade-off is known to exist between the manufacturing cost involved and the surface roughness upto which a component can be produced during a particular manufacturing process. This trade-off along with the dependence of a multitude of factors on the surface roughness makes it imperative to precisely control the surface roughness during a manufac-

S. Manroo (✉) · S. Ganiny
Mechanical Engineering Department, NIT Srinagar, Srinagar, Jammu and Kashmir, India
e-mail: suhail_06phd17@nitsri.ac.in

S. Ganiny
e-mail: suhail_15phd16@nitsri.ac.in

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_15

turing process. Moreover, from the knowledge of the input parameters alone, the prediction of surface roughness upto which a particular manufacturing process is likely to produce the component is of immense significance [7, 8].

In this work, we develop models for the surface roughness prediction of brass during a turning operation, while using the turning parameters of feed-rate, depth of cut (hereafter, DOC), and cutting speed as process variables. In a typical turning operation, a linearly moving cutting tool removes material from a rotating workpiece (here brass) in order to give it a cylindrical shape. The surface roughness upto which a component can be machined during turning is governed by many variables; that include feed-rate, DOC, cutting speed, shape and size of cutting tool, type and amount of lubrication, ambient conditions (temperature, pressure, humidity, etc.), operator skill, etc. [6]. It is thus very difficult, if not impossible, to determine a quantitative relationship between various parameters and the surface roughness upto which a component can be turned. Nevertheless, as already pointed out, the prediction of surface roughness from the knowledge of the input parameters alone is inevitably significant as it helps in assessing the capabilities of a turning operation [8]. While confronted with such a problem, the only solution is to derive the models empirically based on the experimentally gathered data.

Following this philosophy, many researchers have developed data-driven models for modeling and prediction of surface roughness. Vaxevanidis et al. [9], for example, have used a full quadratic regression model for prediction of the surface roughness during the turning of $CuZnPb_3$ brass alloy. Wang and Feng [10] modeled the surface roughness in turning using a nonlinear regression model, while Sharma et al. [11] utilized an artificial neural network model for the purpose. Cakir et al. [8] also made use of regression analysis in order to study the influence of feed-rate, DOC, and cutting speed on the roughness of AISI P20 tool steel using two geometrically similar cutting tools but; one with a coating consisting of a TiCN underlayer, an intermediate layer of Al_2O_3 and a TiN outlayer, while the other one coated with a thin TiAlN layer. It was found that high cutting speeds produce smooth surfaces, whereas, feed-rate has the opposite effect. In [7], artificial neural network and linear regression were used for modeling the surface roughness of AISI 1040 tool steel. Their investigation revealed that the artificial neural network based model estimates the roughness with better accuracy than the regression model. Furthermore, feed-rate was found to be the most dominating factor and cutting speed the least.

In the present investigation, as already mentioned, we model the surface roughness of brass during a typical turning process, while considering feed-rate, DOC, and cutting speed as the influencing variables. We restricted the input parameters to only these three because prior research has pointed out that these are the most dominating factors [8]. We develop the predictive models by using the multi-linear regression technique. While we follow the standard practice of using least square method for computation of the regression coefficients, we make a distinction in that we do not utilize the entire experimental dataset for the purpose. Instead, we arbitrarily choose a subset from the experimental data for computing the involved regression coefficients and accordingly for developing the models. The developed models are then

subsequently validated using the remaining dataset and their efficacy in predicting the surface roughness is ascertained.

The regression models developed serve as objective functions in the genetic algorithm based optimization. The aim of the optimization is to further check the validity of the models by comparing the minimum surface roughness obtained using genetic algorithm with that of the experimentally obtained. Genetic algorithm has been recently used by Bongale et al. [12] for wear rate optimization of hybrid metal matrix composites. The strategy employed here for the genetic algorithm based optimization has been inspired by the work carried out in [13, 14].

2 Experimentation

The data used for developing the regression models was obtained by turning the CW614N brass samples (tensile strength—456 MPa and hardness—154 HB) of mean diameter 24 mm and mean length 35 cm with tungsten carbide (WC) cutting tools on an in-house lathe machine. In accordance with the full-factorial-based design-of-experiments methodology, a total of 27 experiments (each on a separate sample) were carried out, while varying each of the process variables (feed-rate, DOC, and cutting speed) at three levels.

Surface roughness of all the turned brass samples was measured using a portable microprocessor based roughness tester. The threefold measurement of the typical roughness indices, i.e., (R_a, R_z and R_t), was done and the mean value of the three was recorded. The various input combinations of the turning parameters along with the mean values of the surface roughness indices are provided in Table 1.

3 Multi-linear Regression Modeling

In this section, three multi-linear regression models are developed that relate the response variable (R_a) and the predictor variables (feed-rate, DOC, and cutting speed) using explicit empirical relationships. In fact, this particular feature of the regression analysis motivated its use here as compared to other techniques like artificial neural networks, which do not provide an explicit mathematical relationship between the input and output variables. In accordance with the standard statistical terminology, we refer to feed-rate, DOC, and cutting speed collectively as predictors or regressors and the surface roughness (R_a) as the response variable. Before proceeding any further, we partition the experimental data (Table 1) into two subsets, the first, henceforth referred to as in-sample data, consists of 21 data points, and the second, hereafter termed as out-of-sample data, has the remaining 6 data points. The reason for the number of data points assigned to each would become evident later, but it is important to mention that the assignment of particular data points into the two subsets has been made arbitrarily. The in-sample data is used for the computation of regression

coefficients and the development of regression models, while the the out-of-sample data is used for validating the same.

Table 1 Experimental data based on the full-factorial design methodology

S. No.	Feed-rate (mm/rev)	Depth of cut (mm)	Cutting speed (m/min)	R_a (μm)	R_z (μm)	R_t (μm)	In-sample/ Out-of-sample
1	0.1	0.3	60	1.415	8.681	10.173	In-sample
2	0.1	0.3	84	0.935	6.405	7.345	Out-of-sample
3	0.1	0.3	105	0.851	7.149	931	In-sample
4	0.12	0.3	60	1.201	7.864	8.858	In-sample
5	0.12	0.3	84	1.112	7.382	8.419	In-sample
6	0.12	0.3	105	1.204	7.713	8.869	Out-of-sample
7	0.16	0.3	60	1.405	9.401	10.223	In-sample
8	0.16	0.3	84	1.174	8.717	10.553	In-sample
9	0.16	0.3	105	1.165	8.365	8.97	In-sample
10	0.1	0.25	60	1.561	8.555	11.275	In-sample
11	0.1	0.25	84	1.234	8.947	9.423	In-sample
12	0.1	0.25	105	1.263	7.065	7.803	Out-of-sample
13	0.12	0.25	60	1.798	8.847	11.647	In-sample
14	0.12	0.25	84	1.522	8.563	9.402	In-sample
15	0.12	0.25	105	1.551	8.289	8.861	In-sample
16	0.16	0.25	60	1.755	11.34	13.324	Out-of-sample
17	0.16	0.25	84	1.656	9.783	11.671	In-sample
18	0.16	0.25	105	2.198	11.5	13.039	In-sample
19	0.1	0.2	60	1.403	7.707	8.594	In-sample
20	0.1	0.2	84	0.956	6.114	7.024	In-sample
21	0.1	0.2	105	1.123	8.7	9.837	In-sample
22	0.12	0.2	60	1.397	8.36	9.819	Out-of-sample
23	0.12	0.2	84	1.29	8.621	10.215	In-sample
24	0.12	0.2	105	1.168	6.446	7.917	In-sample
25	0.16	0.2	60	1.432	10.33	11.729	In-sample
26	0.16	0.2	84	1.49	9.657	10.639	In-sample
27	0.16	0.2	105	1.644	9.259	10.992	Out-of-sample

3.1 First-Order Regression Model

We assume the following form for the predicted value of the surface roughness ($\hat{R}_a$):

$$\hat{R_a} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 \tag{1}$$

$$= \beta_0 + \sum_{i=1}^{3} \beta_i x_i \tag{2}$$

where the terms β_0, β_1, β_2, and β_3 denote the unknown regression coefficients, and x_1, x_2 and x_3, are the turning parameters of feed-rate, DOC, and cutting speed, respectively. Noting that the in-sample data contains 21 data points, we accordingly have a system of 21 simultaneous equations conforming to the form of Eqs. (1, 2). For the sake of brevity, we write all the 21 equations in a vector-matrix form as:

$$\begin{Bmatrix} \hat{R}_{a,1} \\ \hat{R}_{a,2} \\ \hat{R}_{a,3} \\ \vdots \\ \hat{R}_{a,21} \end{Bmatrix} = \begin{bmatrix} 1 & x_{1,1} & x_{2,1} & x_{3,1} \\ 1 & x_{1,2} & x_{2,2} & x_{3,2} \\ & \vdots & & \\ 1 & x_{1,21} & x_{2,21} & x_{3,21} \end{bmatrix} \begin{Bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{Bmatrix} \tag{3}$$

More compactly, we can write:

$$[\hat{\mathbf{R}}_\mathbf{a}]_{\mathbf{21\times 1}} = [\mathbf{X}]_{\mathbf{21\times 4}}[\beta]_{\mathbf{4\times 1}} \tag{4}$$

From Eq. 4, it is evident that solving for the vector β requires the inversion of a non-square matrix, i.e, **X**, which is formidably complex analytically. We thereby resort to Matlab® for the purpose and make use of the pseudo-inverse command to obtain the same. Based on the computed value of the vector β, the model (Eq. 1) becomes:

$$\hat{R_a} = 1.7563 + 7.4583x_1 - 3.7428x_2 - 0.0039x_3 \tag{5}$$

3.2 Second-Order Regression Model

In this case, we assume:

$$\hat{R_a} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_1^2 + \beta_5 x_2^2 + \beta_6 x_3^2 + \beta_7 x_1 x_2 + \beta_8 x_1 x_3 + \beta_9 x_2 x_3 \tag{6}$$

As was done before, all the 21 equations conforming to Eq. 6 are lumped together and written compactly as:

$$[\hat{\mathbf{R}}_{\mathbf{a}}]_{\mathbf{21}\times\mathbf{1}} = [X]_{\mathbf{21}\times\mathbf{10}}[\beta]_{\mathbf{10}\times\mathbf{1}} \tag{7}$$

The model obtained for this case is:

$$\begin{aligned}\hat{R_a} = {} & -6.3601 + 42.3094x_1 + 66.1265x_2 - 0.0736x_3 - 89.7917x_1^2 - 102.4333x_2^2 \\ & + 0.0004x_3^2 - 108.1667x_1x_2 + 0.2000x_1x_3 - 0.0620x_2x_3\end{aligned} \tag{8}$$

3.3 Third-Order Regression Model

Here, we assume:

$$\begin{aligned}\hat{R_a} = {} & \beta_0 + \beta_1x_1 + \beta_2x_2 + \beta_3x_3 + \beta_4x_1^2 + \beta_5x_2^2 + \beta_6x_3^2 + \beta_7x_1x_2 \\ & + \beta_8x_1x_3 + \beta_9x_2x_3 + \beta_{10}x_1^3 + \beta_{11}x_2^3 + \beta_{12}x_3^3 \\ & + \beta_{13}x_1x_2^2 + \beta_{14}x_1x_3^2 + \beta_{15}x_2x_1^2 + \beta_{16}x_2x_3^2 \\ & + \beta_{17}x_3x_1^2 + \beta_{18}x_3x_2^2 + \beta_{19}x_1x_2x_3\end{aligned} \tag{9}$$

And accordingly, we obtain the model as:

$$\begin{aligned}\hat{R_a} = {} & -0.0042217 - 27.91x_1 + 36.747x_2 - 0.11084x_3 - 130x_1^2 - 93.324x_2^2 \\ & - 0.001642x_3^2 - 91.981x_1x_2 + 1.9655x_1x_3 + 0.64156x_2x_3 - 48.081x_1^3 \\ & - 76.791x_2^3 + 0.0001x_3^3 - 59.85x_1x_2^2 + 0.0019249x_1x_3^2 + 1341.7x_2x_1^2 \\ & - 0.0038741x_2x_3^2 - 3.7408x_3x_1^2 + 0.73938x_3x_2^2 - 4.0474x_1x_2x_3\end{aligned} \tag{10}$$

4 Optimization of Surface Roughness Using Genetic Algorithm

Genetic algorithm is a metaheuristic optimization approach that belongs the class of evolutionary algorithms. It works on the principle of '*survival of the fittest*'. Genetic algorithm begins with a randomly generated population of solutions and in each generation (optimization step) evaluates the fitness of each individual solution. The fitness in most cases is the value of the candidate objective function that is sought to be optimized. The biologically inspired operations of selection, crossover, and mutation are performed during each step, and the iterations are stopped when no further improvement in the population can be attained. The solution so obtained is the global optimum of the objective function. However, unlike the conventional optimization methods, for instance, the gradient-based methods, the convergence of the solution to an optimum is not guaranteed. The algorithm can also stop prematurely in case the

number of iterations reaches a certain specified threshold or any other performance criterion is attained. The solution obtained in such an instance cannot be deemed as an optimum. The genetic algorithm is particularly useful for the optimization of functions that are nonlinear in nature, have discontinuities, and are non-differentiable [15].

In this section, we employ a single-objective genetic algorithm to determine the minimum surface roughness based on the regression models obtained in the previous section. The regression models here serve as the candidate objective functions, and the ranges of the regressor variables, i.e., feed-rate, DOC, and cutting speed, serve as the optimization constraints. Although, the values of the regressor variables are discrete in nature, for analytical convenience, we assume that they lie in the following continuous ranges:

$$x_1 \in [0.1, 0.16]; \qquad x_2 \in [0.2, 0.3] \quad \text{and} \quad x_3 \in [60, 105]$$

The single objective optimization problem can thus be formulated as:

$$\begin{aligned} &\text{minimize} && \hat{R_a}(\mathbf{X}) \\ &\text{subjected to} && \mathbf{LB} \leq \mathbf{X} \leq \mathbf{UB} \end{aligned}$$

where the vector $\mathbf{X}$ represents the regressor variables, and the vectors **LB**, **UB** are the lower and upper bounds on the regressors, respectively. These vectors are given by:

$$\mathbf{X} = \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix}; \qquad \mathbf{LB} = \begin{Bmatrix} 0.1 \\ 0.2 \\ 60 \end{Bmatrix}; \quad \text{and,} \quad \mathbf{UB} = \begin{Bmatrix} 0.16 \\ 0.3 \\ 105 \end{Bmatrix}$$

All the three developed regression models were optimized using the optimization toolbox of Matlab®, and the minimum surface roughness was obtained. Few of the important parameters that were used during the optimization are: population size = 200; generations = 300; elite count = 10; crossover fraction = 0.8. Further, the average change of the objective function was used as the stall condition, and its value was chosen to be 10^{-6}.

5 Results and Discussions

A comparison of the surface roughness predicted by multi-linear regression models, developed in Sect. 3, and the experimentally determined values (see Table 1), is depicted in Fig. 1, for the *in-sample* dataset, and in Fig. 2, for the *out-of-sample* dataset, respectively. It can be seen that the models developed based on the *in-sample* dataset are able to predict the surface roughness very closely to that of the experimentally determined values. This is particularly true, for the second and the third-order

models, which yield better fits as compared to the first-order model. This is expected as the second and the third-order models have more number of regressor variables. Surprisingly, as can be seen from Fig. 2, the models developed using the *out-of-sample* dataset do not follow the same trend in predicting the surface roughness for the *in-sample* dataset. While the third-order regression model has the best fit for the *in-sample* dataset, it yields the worst fit for the *out-of-sample* dataset. The second-order regression model, however, exhibits a good correspondence with the experimental values for both the *in-sample* and the *out-of-sample* datasets. The statistical measures of mean-squared error and coefficient of determination also suggest that among the three regression models, the second-order model yields the best fit with the experimental outcomes. The mean-squared error of the models in the order first, second, and third are: 0.067, 0.015, and 0.005 for the *in-sample* dataset, and

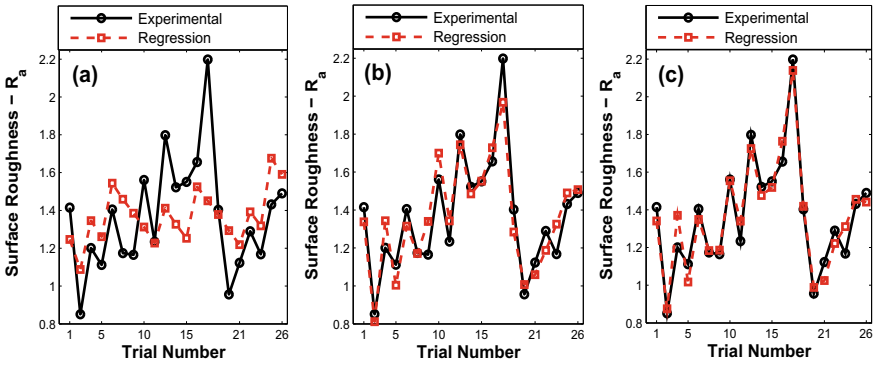


Fig. 1 A comparison of the surface roughness predicted by multi-linear regression models and the experimentally determined values, for the *in-sample* dataset—subfigures **a–c** correspond to first-, second-, and third-order regression models, respectively

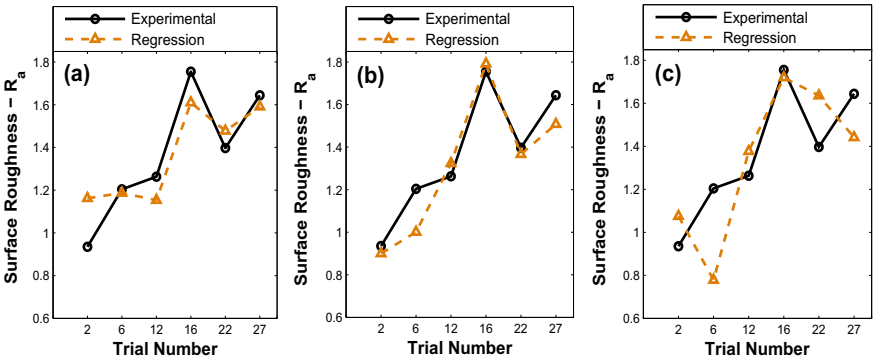


Fig. 2 A comparison of the surface roughness predicted by multi-linear regression models and the experimentally determined values, for the *out-of-sample* dataset—subfigures **a–c** correspond to first-, second-, and third-order regression models, respectively

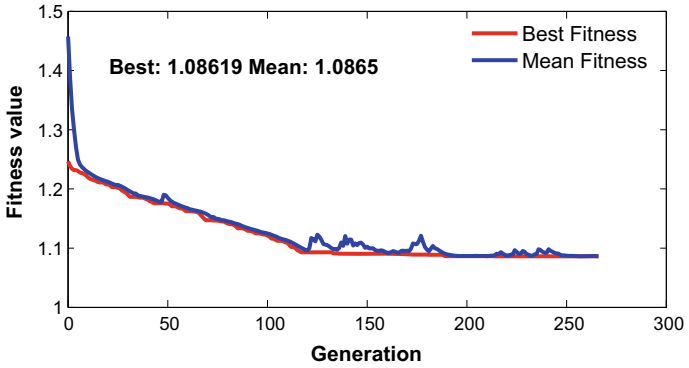


Fig. 3 Genetic algorithm iterations for the best and mean surface roughness values in the candidate population solutions—*first-order regression model*

0.018, 0.011, and 0.053 for the *out-of-sample* datasets, respectively. The coefficient of determination in the same order is: 0.25, 0.851, 0.963, for the *in-sample* dataset, and 0.795, 0.892, and 0.324 for the *out-of-sample* dataset, respectively. The second-order model can thus be deemed as the best regression model for the prediction of the surface roughness of brass during the turning operation.

The results of genetic algorithm based optimization are shown in Figs. 3, 4, and 5. As evident from the results, there is a good match between the mean values of the candidate population and the optimum values of the population. In particular, the results for the first and the second-order regression models are in close agreement, while there are appreciable deviations in the results of the third-order regression models. Furthermore, as depicted in the figures, the global minimum for the surface roughness obtained by the genetic algorithm is 1.08619, for the first-order model, 0.798218, for the second-order model, and 0.744137, and for the third-order model. The minimum surface roughness in the experimental results (Table 1) can be observed to be 0.851, which is very close to the global minimum predicted by the genetic algorithm based on the second-order regression model. This again implies that the second-order regression model is a good predictor of the surface roughness of brass during a typical turning operation.

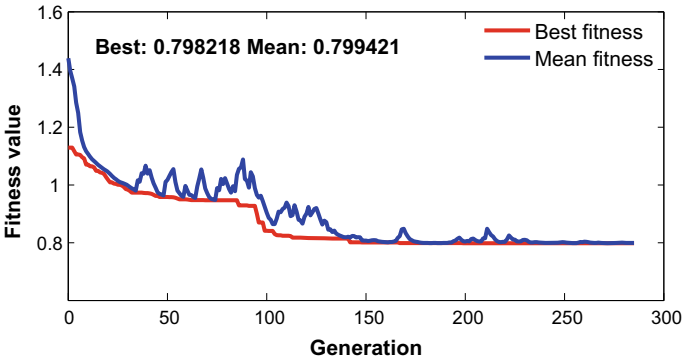


Fig. 4 Genetic algorithm iterations for the best and mean surface roughness values in the candidate population solutions—*second-order regression model*

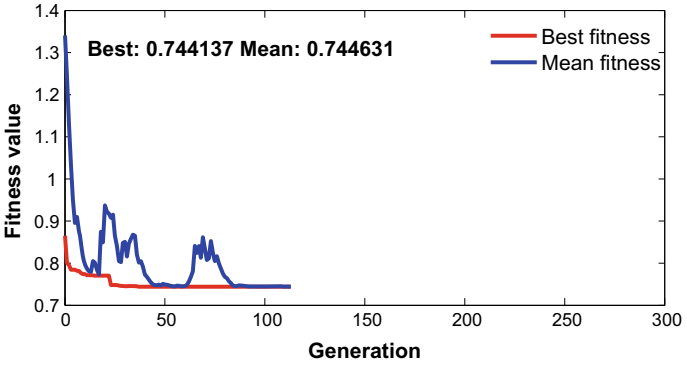


Fig. 5 Genetic algorithm iterations for the best and mean surface roughness values in the candidate population solutions—*third-order regression model*

6 Conclusions

In this work, mathematical models based on the multi-linear regression technique were developed for estimating and predicting the surface roughness of a brass alloy (CW614N brass) during a turning process. The results obtained herein reveal that the second order regression model is statistically more reliable in modeling and subsequently predicting the surface roughness in turning of brass. The adequacy of the models was also assessed by determining the minimum surface roughness (global minimum) using genetic algorithm based optimization technique, and again, it was found that the second-order model is better at modeling the surface roughness during turning than the other two models viz the first-order and the third-order regression models. The findings of this work can be utilized in other experimental settings

wherein brass samples are turned with machining parameters other than the ones used here.

References

1. M.L. Rahaman, L. Zhang, M. Liu, W. Liu, Surface roughness effect on the friction and wear of bulk metallic glasses. Wear **332**, 1231–1237 (2015)
2. J.H. Siu, L.K. Li, An investigation of the effect of surface roughness and coating thickness on the friction and wear behaviour of a commercial mos_2-metal coating on aisi 400c steel. Wear **237**(2), 283–287 (2000)
3. I. Kim, Y. Yang, G. Park, Effect of titanium surface roughness on oxygen catalytic recombination in a shock tube. Acta Astronautica **166**, 260–269 (2020)
4. X. Li, M. Dong, D. Jiang, S. Li, Y. Shang, The effect of surface roughness on normal restitution coefficient, adhesion force and friction coefficient of the particle-wall collision. Powder Technol. **362**, 17–25 (2020)
5. B. Podgornik, F. Kafexhiu, A. Nevosad, E. Badisch, Influence of surface roughness and phosphate coating on galling resistance of medium-grade carbon steel. Wear 203180 (2020)
6. J.T. Black, R.A. Kohser, *DeGarmo's Materials and Processes in Manufacturing* (Wiley, 2017)
7. I. Asiltürk, M. Çunkaş, Modeling and prediction of surface roughness in turning operations using artificial neural network and multiple regression method. Expert Syst. Appl. **38**(5), 5826–5832 (2011)
8. M.C. Cakir, C. Ensarioglu, I. Demirayak, Mathematical modeling of surface roughness for evaluating the effects of cutting parameters and coating material. J. Mater. Process. Technol. **209**(1), 102–109 (2009)
9. N. Vaxevanidis, N. Fountas, A. Koutsomichalis, J. Kechagias, Experimental investigation of machinability parameters in turning of cuzn39pb3 brass alloy. Proc. Struct. Integr. **10**, 333–341 (2018)
10. X. Wang, C. Feng, Development of empirical models for surface roughness prediction in finish turning. Int. J. Adv. Manuf. Technol. **20**(5), 348–356 (2002)
11. V.S. Sharma, S. Dhiman, R. Sehgal, S. Sharma, Estimation of cutting forces and surface roughness for hard turning using neural networks. J. Intell. Manuf. **19**(4), 473–483 (2008)
12. A.M. Bongale, S. Kumar, T. Sachit, P. Jadhav, Wear rate optimization of al/sicnp/e-glass fibre hybrid metal matrix composites using taguchi method and genetic algorithm and development of wear model using artificial neural networks. Mater. Res. Express **5**(3), 035005 (2018)
13. G. Rangajanardhaa et al., Development of hybrid model and optimization of surface roughness in electric discharge machining using artificial neural networks and genetic algorithm. J. Mater. Process. Technol. **209**(3), 1512–1520 (2009)
14. P. Suresh, P.V. Rao, S. Deshmukh, A genetic algorithmic approach for optimization of surface roughness prediction model. Int. J. Mach. Tools Manuf. **42**(6), 675–680 (2002)
15. S. Sumathi, S. Sivanandam, *Introduction to Data Mining and its Applications*, vol. 29 (Springer, 2006)

A Novel AUTOSAR-Compliant Approach to Develop Resolver as Complex Device Driver for Electric Vehicle Inverter

Sakshi Sharma, Bhanu Pratap, Ajit Safai, and Raghav Ankur

1 Introduction

Automobile exports have grown 14.5 during fiscal year 2019. Also, many programs by the Government of India and the major automobile companies in the Indian market are expected to make India a leader in the automotive market in the world in near future. All these development in the automotive market seems to be a failure when India stands among bottom five countries on the Environmental Performance Index (EPI) 2018. According to Central Pollution Control Board (CPCB), the vehicular pollution is a major reason for the degrading quality of air in the country. Fueling with electricity therefore offers the replacement for conventional internal combustion engine vehicles.

Conventional powertrain provides the parts to vehicle that delivers power to it, typically involving engine, transmission, and drivetrain. The inverter controls electric motor in the electric drivetrain. This is a main component of the electric vehicle (EV) same as combustion vehicle's engine management system (EMS). An integrated PCB controls the inverter, which should be designed to maximize thermal efficiency and

S. Sharma (✉) · B. Pratap
Electrical Engineering Department, National Institute of Technology Kurukshetra, Kurukshetra, India
e-mail: 08aahanasharma96@gmail.com

B. Pratap
e-mail: bhanumnnit@gmail.com

A. Safai · R. Ankur
KPIT Technologies Ltd., Pune, India
e-mail: ajit.safai@kpit.com

R. Ankur
e-mail: raghav.ankur@kpit.com

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_16

minimize switching losses. As a result, vehicle's range is directly related to the efficiency and performance of the main inverter.

To implement solution to increasing complexity due to growth of software sharing and functionality and also the rise in number of ECUs, the inverter design is AUTOSAR compliant [1]. AUTOSAR serves as a common standard across the whole automotive industry that enhances the scope of applications of vehicle functionalities without affecting the existing model. To design or develop a function outside of the normal AUTOSAR basic software stack which includes complex sensor evaluation, CDD are taken into usage. At present, permanent-magnet synchronous motors (PMSM) are extensively used in EVs because of their high power and torque capacities. A PMSM is a rotating electric machine where the stator is a classic three-phase stator. In order to control PMSM, a resolver (AD2S1210) is attached to the motor shaft. It provides the angular velocity and position data for further control action in digital format and uses serial peripheral interface (SPI) for the data transmission to multicore microcontroller. Resolver data is required as feedback for field-oriented control (FOC) of AC motors.

The FOC consisting of Clarke–Park transformations is a commutation algorithm designed for inverter circuitry in order to control the position and velocity of PMSM and other related operations like controlling faults of microcontroller, temperature regulation of IGBT, etc. For the suggested resolver as CDD, the FOC collects samples at every 100 μs. Therefore, using SPI protocol to fetch position and velocity data serves advantageous. Resolver provides data over SPI at a baud rate of 10 Mbps which means it just takes maximum of 10 μs. Thus, interfacing resolver via serial communication protocol not only optimizes the timing requirements but also helps in achieving inverter's optimal performance.

2 Theory of Operation

2.1 Resolver

A resolver, which is a rotating transformer, has windings on the rotor as well as on the stator. There is no winding present on the rotor in variable reluctance resolver, as shown in Fig. 1. The primary winding is present on the stator; however, sinusoidal changes in the secondary coupling with the angular position is achieved by the saliency in the rotor design.

The resolver output voltages (S3 − S1, S2 − S4) are given by, $S3 - S1 = E\sin(\omega t) \times \sin(\theta)$, $S4 - S2 = E\sin(\omega t) \times \cos(\theta)$, where θ is the shaft angle,E denotes the input amplitude of rotor, and $\sin(\omega t)$ symbolizes the rotor excitation frequency. The windings of stator are displaced mechanically by 90°, and AC reference excites the primary winding. Position of the rotor (shaft) relative to the stator is responsible for the amplitude change of subsequent coupling onto the stator

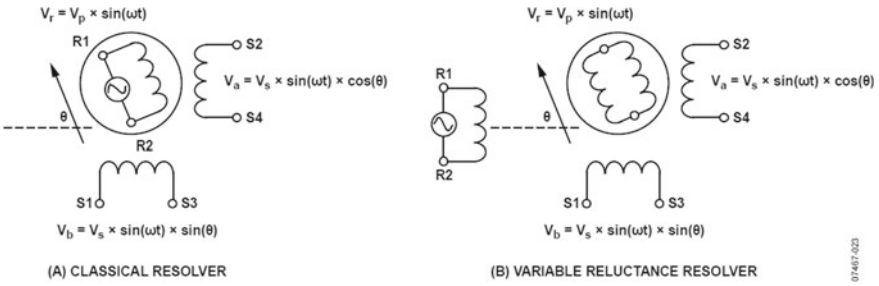


Fig. 1 Types of resolver

secondary windings. Finally, the sine and cosine of the shaft angle modulates the resultant output voltages (S3 − S1, S2 − S4) of resolver [2].

Resolver to Digital Conversion: AD2S1210 supports 3125 rps maximum tracking rate for a 10 bit resolution and provides serial as well as parallel 10 bits to 16 bits absolute position and velocity outputs. The operating principle of AD2S1210 is a Type II tracking closed-loop approach. The resolver position is continuously tracked by the output without requiring any wait states and external conversion. The output is incremented by one least significant bit when resolver is moved along a position identical to LSB weighting. The converter follows the shaft angle θ and produces an angle ϕ which gets fed back. This angle ϕ is compared to the input angle θ. Hence, when converter is correctly tracking θ, the resulting error between ϕ and θ is driven toward 0. For measuring error, S3 − S1 gets multiplied by, $\cos(\phi)$ and S2 − S4 gets multiplied by $\sin(\phi)$ as follow, $E\sin(\omega t) \times \sin(\theta)\cos(\phi)$ (for S3 − S1), $E\sin(\omega t) \times \cos(\theta)\sin(\phi)$ (for S4 – S2). The difference is taken as, $E\sin(\omega t)(\sin(\theta)\cos(\phi) - \cos(\theta)\sin(\phi))$. The resultant signal is demodulated using the internally generated reference, which yields, $E(\sin(\theta)\cos(\phi) - \cos(\theta)\sin(\phi)) = E\sin(\theta - \phi)$, where $\theta - \phi$ is the angular error.

A closed-loop system is formed by phase-sensitive demodulator, a compensation filter to nullify error, and some integrators. When the error is achieved, ϕ becomes equivalent to the resolver angle θ which lies in the converter rated accuracy. As the sustained velocity inputs can be tracked with ease without any implicit error, this serves the reason of using type II tracking loop. The circuit that detects fault of AD2S1210 can sense loss of resolver signals, input signal mismatch, out-of-range input signals, or loss of position tracking. However, in case of a fault, there may be a significant difference in the position given by AD2S1210 from the actual shaft position of the resolver [2]. The conditions to analyze fault is presented in Table 1.

Modes of operation: The resolver AD2S1210 operates in two modes: configuration mode and normal mode. To set the resolution, excitation frequency, and fault detection thresholds of AD2S1210, requires register programming, and hence, configuration mode is used. Configuration mode can also be used to read back data in the position and velocity registers. The AD2S1210 is operated either completely in configuration mode or in normal mode on the completion of the initial configuration. Then, this part can be separated out of configuration mode. When resolver operates

Table 1 AD2S1210 decoding of fault detection

Condition	DOS pin	LOT pin	Order of priority
Loss of signal (LOS)	0	1	1
Degradation of signal (DOS)	0	1	2
Loss of tracking (LOT)	1	0	3
No fault	1	1	N/A

Table 2 Inputs setting for configuration mode

A0	A1	Result
0	0	Normal mode—position output
0	1	Normal mode—velocity output
1	0	Reserved
1	1	Configuration mode

in normal mode, angular position or angular velocity data is provided as data output. To obtain the mode of AD2S1210 and also to determine the type of data which gets through outpin pin, A0 and A1 inputs settings are used as per Table 2.

2.2 Multicore Microcontroller

The multicore technology consists of two or more cores or computational/processing units which function in parallel to read and write instructions. The major advantage of using multicore microcontroller is that it gives the similar performance as the single faster processor at reduced power dissipation and less clock frequency simultaneously handling more tasks parallel. Automotive real-time integrated next-generation architecture (AURIX) is a 32-bit Infineon microcontroller family targeting automotive industry in term of performance and safety (QM to ASIL-D) [3]. AURIX TC277 has a triple TriCore with operating frequency as 200 MHz, flash memory of 4 MB, and a powerful generic timer module (GTM). These features aim for a reduced complexity, best-in-class power consumption, and significant cost savings. TC277 supports reduced instruction set computing (RISC) processor architecture and includes on-chip memories and peripherals. In order to meet the needs of highly demanding embedded control systems applications, TC277 is designed which deals with the competing issues of price/performance, computational power, data bandwidth, power consumption, and real-time responsiveness [4].

2.3 Complex Driver Development

CDD**.** AUTOSAR-layered software architecture consists of a software module CDD which is not standardized by AUTOSAR. basic software (BSW) modules APIs or AUTOSAR interfaces can access or get accessed by CDD. Complex driver's layer of BSW encapsulates CDD which interacts with RTE or BSW modules [5]. The main objective of CDD is the implementation of critical sensor and actuator control through directly accessing the microcontroller's specific interrupts along with external devices, complex microcontroller peripherals for fulfilling the certain timing requirements. Also, it can be used to apply enhanced protocols with non-AUTOSAR system's encapsulated legacy functionality. CDD implementation may depend on ECU, application, and μC. Lastly, the CDD serves as a mechanism of migration for introducing new concepts in the AUTOSAR software architecture [6].

Interfacing with MCAL modules: CDD directly accesses the microcontroller resources. When there are no rigid constraints or if MCAL manages the needed resources, CDD must use MCAL. This is strongly recommended to avoid any challenging conflicts further (e.g., disallowing the parallel access for the same channel as data input–output services are not re-entrant). Hence, CDD takes in use the MCAL modules standard APIs for accessing MCAL different modules (SPI, DMA, etc., in case of resolver). CDD for microcontroller driver can be developed and practiced, but it is unable to perform other standard module as it lies in the lower layer of the architecture [5].

2.4 SPI

SPI is a four-line, synchronous, serial bus, which is generally used to connect a microcontroller to peripheral circuitry. SPI operates in a master–slave mode where the microcontroller is usually the master while the peripheral chips respond as slaves. An SPI protocol has four signal wires:

1. Master Out Slave In (MOSI)—MOSI signal is transferred and produced by master, and is received by slave.
2. Master In Slave Out (MISO)—MISO signal is sent by slaves and is received by the master.
3. Serial Clock (SCLK or SCK)—SCK is generated by the master for synchronizing data transfer between master and slave.
4. Chip Select (CS)—CS signal is generated by master for selecting individual peripheral devices.

This communication protocol is always initiated by the master. Initially, the master configures the clock, with the frequency, which is less than or equal to the maximum frequency supported by the slave device. Afterwards, the desired slave for communication is selected by the master by asserting the CS pin of that particular slave. At

last, any of the data transmission modes can occur during each clock cycle. Thus, the master transfers a bit on the MOSI line; the slave receives it from that same line; the slave sends a bit on the MISO line; the master reads it from that same line [7].

SPI specifications in AUTOSAR standard: A hierarchical structure is defined as per AUTOSAR standard which includes channels, jobs, and sequences to describe data transmission process on a SPI bus. A channel actually holds place for the transmitted data. A job consists of one or more channels and has same CS signals attached to its definition. A sequence has one or more jobs. In Fig. 2 [8], the sequence is composed of two jobs, job n and m. Firstly, job *n* mediates the bus. On the completion of transfer of data by channel *x*, the other channel of same job gets started without leaving the bus. Once, both channels are transferred completely, job *n* releases bus, and job *m* starts sending data until sequence *a* is completed [8].

QSPI of AURIX TC277 microcontroller: The main purpose of the QSPI module is to provide synchronous serial communication with external devices using clock, data-in, data-out, and slave select signals. The focus of the module is set to fast and flexible communication: either point-to-point or master to many slaves communication. Parallel requests from on-chip bus masters to a module will be executed sequentially via the on-chip bus system [7]. Read–modify–write feature provides an automatic read/write sequence where no other master can access the module in between. The essential features include the 4–32 programmable number data bits possible for 50 Mbps, flexible baud rate and delays, control and data handling by the DMA controller, seven slave select inputs and sixteen programmable slave select outputs in slave mode and master mode, respectively. The present task uses the SPI communication in master mode, full duplex, short data mode, and single move modes of TC277 QSPI module [9].

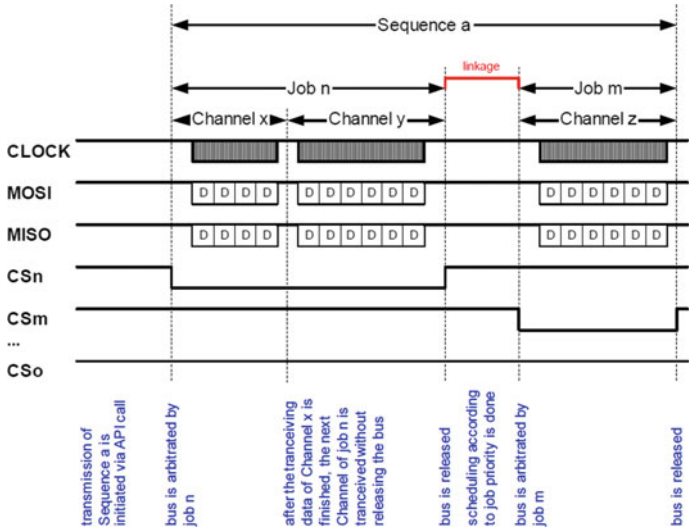


Fig. 2 SPI data transmission according to the AUTOSAR specifications

3 Methodology

To develop CDD for resolver in order to control PMSM velocity and position, the resolver data has to be timely fetched and provided to the tricore microcontroller. The discussed approach uses two channels (one each for velocity and position), one job and one sequence.

The flowchart in Fig. 3 demonstrates the normal mode data obtained from resolver and its serial interface.

The steps involved in the AUTOSAR-compliant resolver driver development are:

1. Resolver sensor AD2S1210 fetches the 16-bit position and velocity data from PMSM.
2. The data is then propagated to tricore microcontroller via SPI, DMA, and IRQ drivers of MCAL.
3. The respective necessary configurations are performed in AUTOSAR 4.0.3 complaint configuration tool EB-Tresos to generate the .c and .h files.

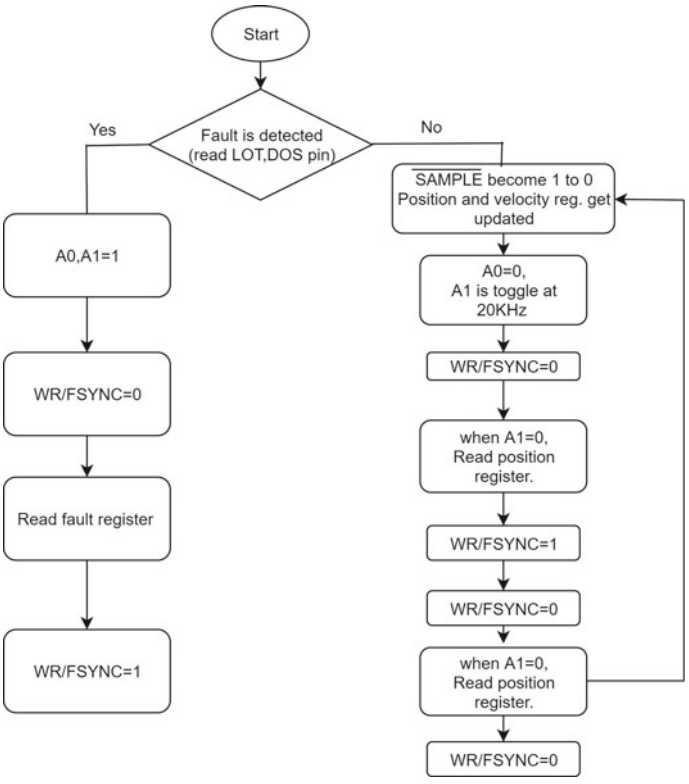


Fig. 3 Flowchart for normal mode (serial interface)

Table 3 Configuration tool settings

SPI parameter	Appropriate configurations for CDD
Baud rate	10 Mbps
CS polarity	Low
Data shift edge	Leading
Shift clock idle level	High
Channel configurations	
Channel type SPI data width Spi EB max length SPI transfer start	External buffers (EB) 32 bits 12 MSB

4. The driver code for resolver is developed which includes resolver fault handler, resolver functions, resolver MCAL, and resolver SPI files.
5. All the configured and developed files are then compiled in high-tech IDE compiler to obtain .elf file.
6. The Trace32 debugger is taken in use for debugging the driver.

4 Outcome

The approach to develop resolver as CDD is given in Table 3 which involved the configuration settings of MCAL module—SPI on the tool EB-Tresos:

Data transfer from resolver to μC via SPI is initiated with the application program interface (API) call. APIs are the high-level functions that access the peripherals and devices irrespective of their physical location and connectivity. The configuration code generated by the settings for resolver and following APIs along with their type definitions are invoked in the development of resolver as CDD: (i) Spi_ConfigType, (ii) Spi_Init, (iii) Spi_DeInit, (iv) Spi_SetupEB (SPI_CHANNEL_24BIT,…), (v) Spi_AsyncTransmit (SPI_SEQ_24BIT), (vi) Spi_GetSequenceResult (SPI_SEQ_24BIT). Hence, with the resolver, tool settings, and code generated for the interface, the position, velocity, and fault data are obtained by microcontroller TC277.

The relative SPI bus timing diagram is shown in Fig. 4 [2], and practically obtained position and fault data signals on MISO pin and SCLK of AURIX from resolver is shown in Fig. 5.

5 Conclusion

Resolver, a complex sensor, is developed as CDD which fulfills the special functional and timing requirements in the inverter designing for EV. In addition, it also encapsulates legacy functionality of a non-AUTOSAR system. The microcontroller

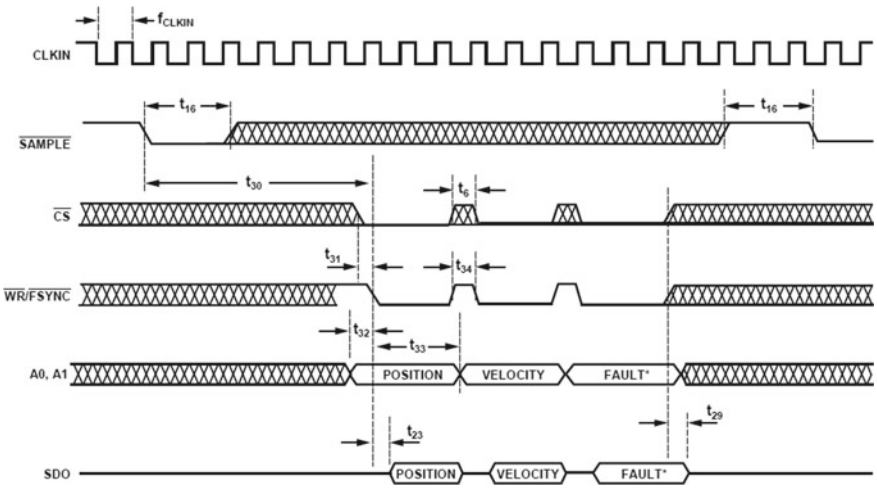


Fig. 4 Serial communication timing diagram of resolver

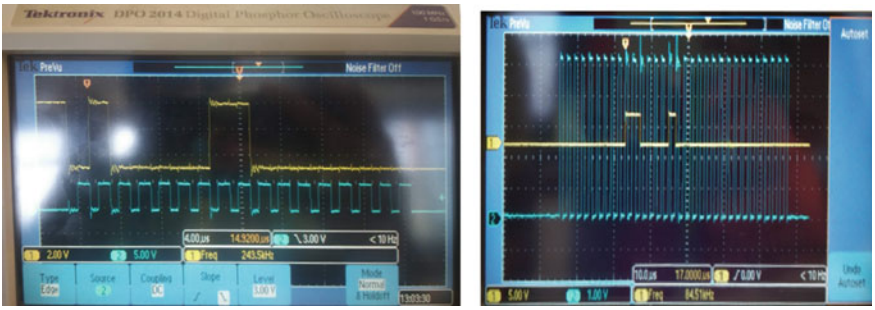


Fig. 5 Position data of resolver with clock cycles and position data with fault signals captured

features are extracted for controlling the PMSM position and velocity. The entire driver is building in such a way that it is easy to upgrade with the future upgrading of EVs generation. Through the program interface of SPI protocol, the entire data of PMSM can be extracted with the maximum of 8–10 μs. This optimizes the timing requirement of FOC of PMSM build for the inverter.

References

1. P. Nag, U. Ghanekar, J. Harmalkar, A novel multi-core approach for functional safety compliance of automotive electronic control unit according to ISO 26262, in *International Conference for Convergence in Technology* (Pune, India, 2019)
2. Resolver AD2S1210 datasheet

3. R. Tripathy, J. Harmalkar, A. Kumar, A functionally safe dual-bus platoon architecture for future smart cities, in *International Conference on Trends in Electronics and Informatics* (Tirunelveli, India, 2019)
4. M. Ashwathnarayan, J. Guddeti, A methodology to validate the on-chip buses of a microcontroller, in *IEEE East-West Design & Test Symposium* (Kazan, Russia, 2018)
5. AUTOSAR, Complex Driver design and integration guideline, AUTOSAR Official, Identification No. 622, (2017)
6. W. Dafang, Z. Jiuyang, Z. Guifan, H. Bo, L. Shiqia,g. Survey of the AUTOSAR complex drivers in the field of automotive electronics, in *International Conference on Intelligent Computation Technology and Automation* (Changsha, China, 2010)
7. AURIX SPI Handler Driver, Release v5.3 (2018)
8. AUTOSAR, specifications of SPI Handler/Driver, AUTOSAR Official, Identification No. 038, (2017)
9. AURIXTM TC27x D-Step User's Manual

Role of Power Electronics and Optimization Techniques in Renewable Energy Systems

Abrar Ahmed Chhipa, Shripati Vyas, Vinod Kumar, and R. R. Joshi

1 Introduction

The present and future energy situations are getting more complicated and the urge for a smart grid. Since, the electric grid is growing very fast and changing continuously, new challenges arise that need to be resolved. The integration of modern renewable energy sources (RESs), energy storage systems (ESSs), and better optimization intelligent techniques are solutions to challenges of preserving power quality, government power regulatory, interconnection of electric grids and sub grids, reliability, and overall grid stability [1–3].

Energy consumption by commercial and non-commercial increases day by day which leads to greater energy demand. To overcome this demand, installation of the new power plants with higher power generation capacity or purchase the energy from other regions which are nearer. This will require the complex interconnection of different nature power generating stations. This leads to greater challenges faced during interconnection and power sharing such as the reliability and stability of the overall distribution system.

In addition, use of fossil fuels emits hazardous gases and nuclear waste from nuclear power generating units to affect global warming. These factors affect the

A. A. Chhipa (✉) · S. Vyas · V. Kumar · R. R. Joshi
College of Technology and Engineering, Udaipur 313001, Rajasthan, India
e-mail: abrarahmed@ieee.org

S. Vyas
e-mail: Shripativyasjoni@gmail.com

V. Kumar
e-mail: vinodcte@yahoo.co.in

R. R. Joshi
e-mail: rrJoshi_iitd@yahoo.com

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_17

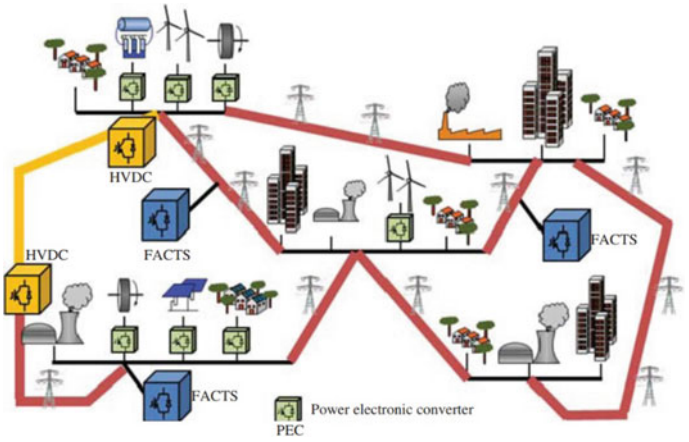


Fig. 1 Structure of modern distributed electrical power grid

growth of such types of new energy generation system installation. Such a new system should be generating energy efficiently along with minimum impact on global warming. Therefore, during recent decades new RESs are integrated into the grid primarily photovoltaic and wind power plants. Figure 1 shows the structure of the modern distributed electric power grid.

This chapter reviews the modern distributed grid concept and efficient solution utilizing power electronic converters and optimization techniques. Further, this chapter focuses on RESs like wind power and photovoltaic power energy storage systems and optimization techniques for maximum power point tracking. Different topologies and technologies are reviewed which are already in existence as well as available for replacement or under research.

2 Renewable Energy Systems

2.1 Categorization of Renewable Energy Sources

The categorization of main renewable energy sources is shown in Fig. 2.

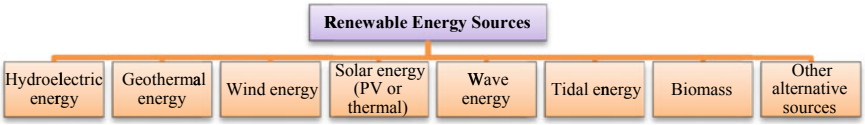


Fig. 2 Categorization of renewable energy sources

	Housing Applications		Solar Energy		Marine/fluvial Energy			Wind Energy	Geothermal Energy
Renewable Energy Systems	Roof PV	Solar Thermal Roof	PV Plant	Solar Thermal	Wave Energy	Tidal Energy	Hydroelectric		
Maximum Nominal Power	50kW		250MW	370MW	150kW	256MW	14GW	6MW per Turbine	720MW
Usual Converter Topologies	DC/AC	-	DC/AC	-	AC/DC, DC/DC, DC/AC		-	Back to back	-
	DC/DC	-	DC/DC	-					
Typical Power Semiconductors	MOSFET	-			IGBT, IGCT		-	IGBT IGCT	-
		-							
Availability	Solar irradiance dependence				Random	Intermittent	Seasonal	Random	Constant
Technology trend	↑ Power density ↑Efficiency		↑ Nominal power ↑ Efficiency		↑ Robustness Under storms		-	Gearless, Offshore, ↑ Power	↑ Development cost

Fig. 3 Categorization of present RESs

The categorization of present RESs is shown in Fig. 3. This mainly depends on the maximum nominal power, usual converter topologies, typical power semiconductor, availability of sources, and technology trends [4].

3 Wind Energy and Photovoltaic Energy

3.1 Wind Energy

In recent decades, wind energy technology has grown enough to accomplish energy market demand [5]. Windmills were invented long ago, in the nineteenth century, the first wind turbine (WT) was designed which can produce electricity. In the twentieth century, different types of variable speed and fixed speed wind turbines (WTs) were designed. Fixed speed wind generators are connected directly to the electric grid via transformer, and this arrangement is simple to deploy and cheap. Because of lack of power electronics converter, it will suffer mechanical stress and current oscillates.

To overcome these mentioned operational issues, in the 1990s, several inventions of new wind energy generation technologies emerged to operate with variable and fixed wind speed along with pitch control. Figure 4a presents the simplest and most efficient doubly fed induction generator (DFIG) design with a reduced capacity converter.

The new modern WTs comprise of back-to-back full power capacity power converter topology act as a bridge between the grid and a permanent magnet synchronous generator (PMSG) shown in Fig. 4b. This case meets the regulation requirements as all quantiles like active and reactive power have a wide range of control. Drop back of this configuration is that the cost of the converters gets increased.

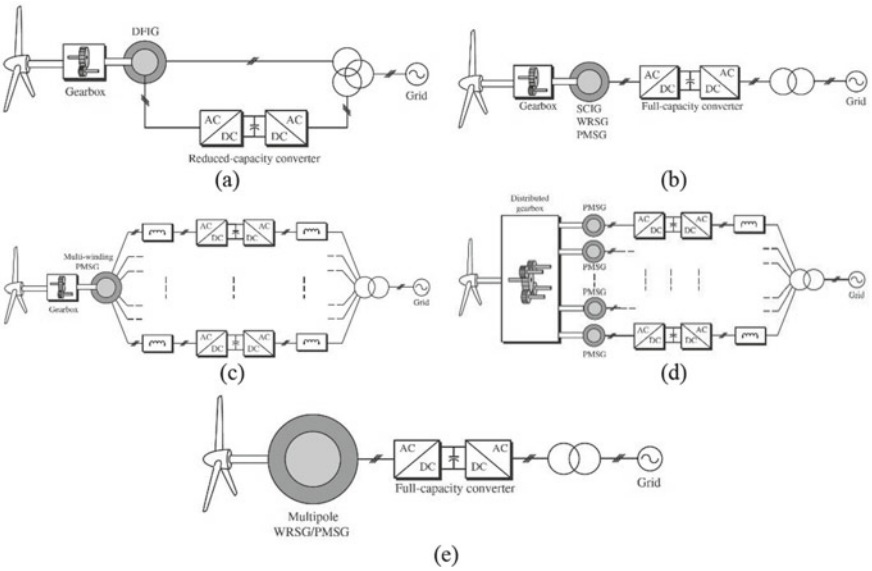


Fig. 4 Different configurations of WT integration in the electric power grid: **a** reduced capacity back-to-back converter for DFIG, **b** maximum power capacity back-to-back converter with the electrical machines like PMSG or squirrel-cage induction generator (SCIG), **c** number of distributed back-to-back converters for multi-winding PMSG configuration, **d** dispersed drive train for multiple PMSGs configuration, **e** maximum power capacity back to back converter for multipole WRSG/PMSG configuration

Figure 4c shows the multi-winding PMSG linked to numerous back-to-back power converters. Main advantage of this configuration is that power is shared, so the fault tolerance gets improvised. Another possible configuration with distributed gearboxes is shown in Fig. 4d. This configuration is more complex compared with Fig. 4c. Finally, Fig. 4e show the gearless WT configuration. This utilizes the multipole machines.

3.2 Photovoltaic Energy

Cost of photovoltaic modules is drastically decreasing day by day and the government in every country is also strongly supporting this technology. This led to the consideration of this technology for future renewable energy scenarios. One of the main advantages of this technology is it can be installed for residential applications (<5 kW). Apart from this application, in recent years, large PV plants are also built to meet the power grid demand.

PV panels produce the power in DC form and industry and non-industry require AC power. To convert AC voltage into DC voltage, power converter is required.

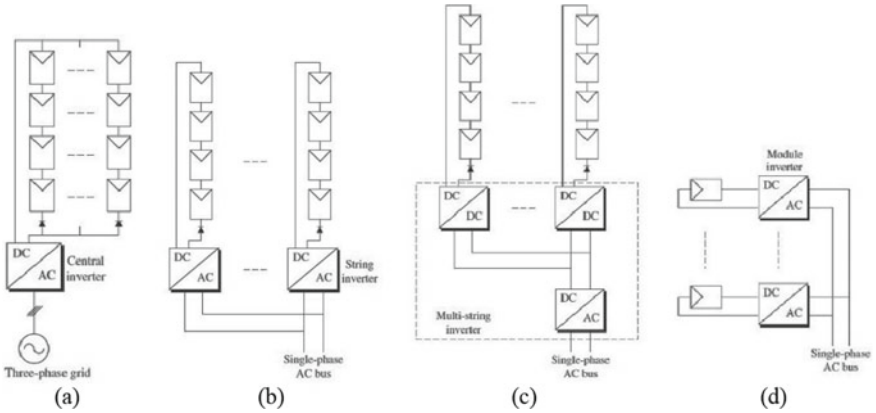


Fig. 5 Photovoltaic modules and power converter configurations: **a** series–parallel, **b** string inverter, **c** multi-string, and **d** module inverter

Hence, there is a wide variety of PV inverters and converters on the market with emerging technologies and topologies. The following possible configurations are presented in Fig. 5: (1) series–parallel (10–250 kW), (2) string inverter (<5 kW), (3) multi-string (several KW), (4) module inverter (<500 W).

It must be taken into account that different countries have their different power grid regulations like, grid operating frequency, and voltage requirements, also reactive and active power limits. The control strategy for converters and the power conversion stage must be taken into account. Study presented in [6, 7] shows the different topologies of inverters.

4 Energy Storage Systems

As power electronics converters are required to integrate various types of energy generation along with energy storage systems technologies. For efficient grid power management, ESSs are the best suited option to adopt. They provide solutions to the problems like ride through which are present in wind energy, intermittent disconnection characteristics of renewable energy [9, 10].

4.1 Technologies

ESSs devices technologies classification is shown in Figs. 6 and 7. Different ESSs devices characteristics are presented in Table 1.

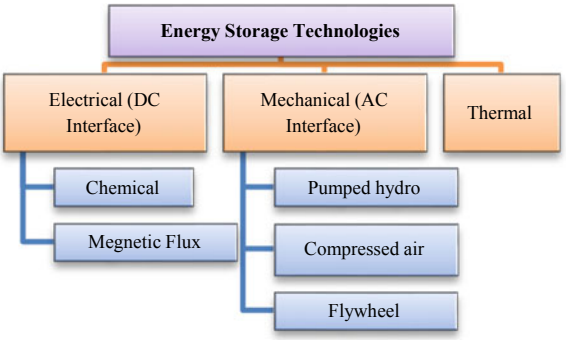


Fig. 6 Classification of energy storage devices technology

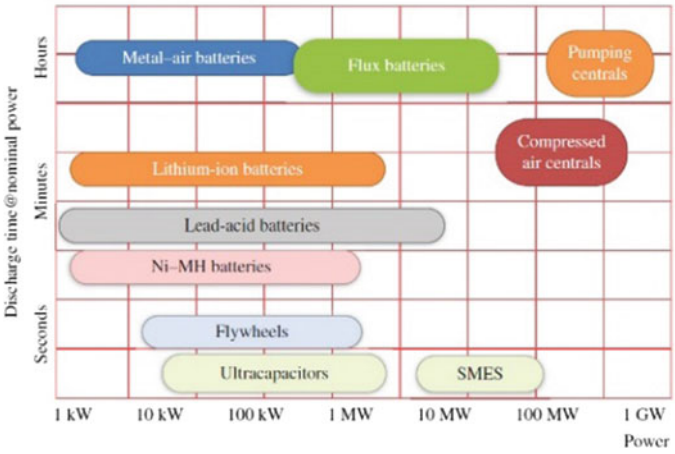


Fig. 7 Energy storage classification based on the power density [8]

- Lead-acid batteries are the oldest type of ESSs. It had been in existence even before the power converters were invented. Cost is low compared with other ESSs.
- Li-ion batteries have high-energy density, making them most common in low-energy consumer applications such as cell phones, wearables, hybrid electric vehicles, etc. Their energy efficiency advantage makes it popular for applications. In comparison with lead-acid batteries they are much expensive, this shortcoming makes it unaffordable for consumer applications.
- In earlier days, NiMH or NiCd was most widely used, but in the current situations, they are being substituted by Li-ion batteries. Reason behind is the high-energy densities and high-power carrying capability. Utilization of these batteries drastically increased in consumer applications. Unfortunately, their high cost is stopping them in applications where cost factor is very sensitive.
- Electrochemical double-layer capacitors (EDLCs): Principle of storing energy in EDLCs is the same as a traditional condenser. Hence, no chemical reaction occurs.

Table 1 Energy storage classification and features summary [8]

Type	Energy efficiency (%)	Energy density (Wh = kg)	Power density (W = kg)	Cycle life (cycles)	Self-discharge
Pb-acid	70–80	20–35	25	200–2000	Low
Ni–Cd	60–90	40–60	140–180	500–2000	Low
Ni–MH	50–80	60–80	220	<3000	High
Li-ion	70–85	100–200	360	500–2000	Medium
Li-polymer	70	200	250–1000	>1200	Medium
NaS	70	120	120	2 000	–
VRB	80	25	80–150	>16,000	Negligible
EDLC	95	<50	4000	>50,000	Very high
Pumped hydro	65–80	0.3	–	>20 years	Negligible
CAES	40–50	10–30	–	>20 years	–
Flywheel (steel)	95	5–30	1000	>20,000	Very high
Flywheel (composite)	95	>50	5000	>20,000	Very high

- Regenerative fuel cells (RFCs): This type of ESSs somewhat work in an analogous way to flow batteries. In RFCs, reactive is either gas–liquid or gases.
- Compressed air energy systems (CAESs): In this ESSs, energy is stored in the form of compressed air. Gas turbines are used to recover this stored energy.
- Flywheel energy storage systems (FESSs): In this ESSs, highly inertial rotating mass stores the kinetic energy.
- Superconductive magnetic energy storage systems (SMESs): Energy is stored in the form of magnetic field flux. Among all the other available ESSs, this technology has the highest power density and dynamic response.
- Thermoelectric energy storage (TEES): This type of system store energy in molten salt or synthetic oil by means of increasing temperature and keeping the temperature at same level in a heat isolated tanks, which can be utilized later.

5 Optimization Techniques

Since the weather conditions such as wind speed, moisture, and sunlight have an unpredictable nature, this leads to varied power output of wind energy generation and photovoltaic generation continuously. Thus, to reach high efficiency and superior

output performance from wind systems and PV systems MPPT algorithms are incorporated with these systems. Several studies have presented the different MPPT algorithm, each has its own merits and demerits. Classification of the MPPT algorithms for wind systems is shown in Fig. 8, and for PV systems are shown in Fig. 9.

Among all available MPPT algorithms for the wind and PV systems, intelligent control technique is on urge because of its intelligence, computational speed, robustness, dynamic response, and superior performance for complex nonlinear systems. The fuzzy logic controller (FLC) has the advantage that it does not require any mathematical model of the plant. Block diagram of the fuzzy logic controller is shown in Fig. 10a.The controllers based on neural networks are reliable replacements for the conventional controller for MPPT. Figure 10b presents the structure of an artificial neural network (ANN). Only downside of ANN is that the neural network must be trained using available historical data.

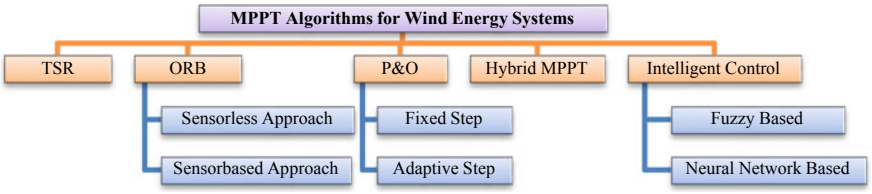


Fig. 8 Classification of MPPT algorithms for wind systems

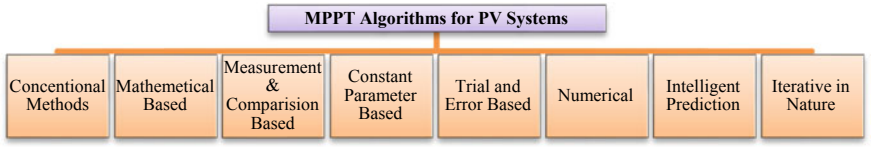


Fig. 9 Classification of MPPT algorithms for PV systems

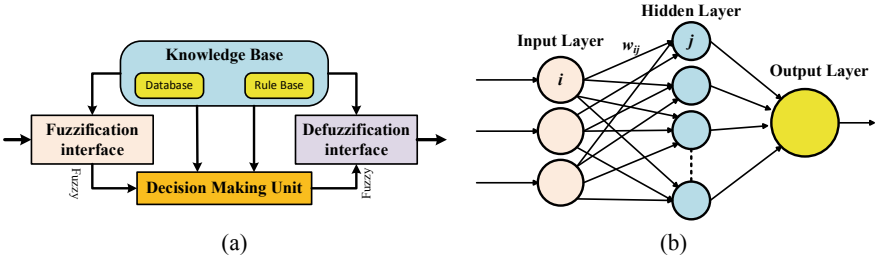


Fig. 10 Block diagrams of intelligent controllers: **a** fuzzy logic controller, **b** artificial neural network

6 Conclusion

Integration of RESs leads to improvement in stability, reliability, and fault tolerance of the overall energy generation and distribution of modern distributed electric power grids. Interconnection of different grids, power regulatory imposed by governments, and integration of RESs leads to challenges in operation. To overcome the challenges faced by the distributed power grid, power electronics is the solution to these challenges. Power electronic converters are the most appropriate solution to incorporate RESs with electric power grids. Flow of power can be fully controlled by using intelligent optimization techniques like fuzzy logic, neural networks. Lastly, ESSs are the best solution to the arbitrary disconnection of renewable energy, ride through, power quality demand by grid codes problems. Different ESSs technologies are discussed in the previous section. In summary, research on developing power optimization techniques and power control strategies for the power electronics converter using artificial intelligence is in urgent need for power grid stability and reliability.

References

1. B.K. Bose, Power electronics and motor drives—recent progress and perspective. IEEE Trans. Industr. Electron. **56**(2), 581–588 (2009)
2. F. Bouhafs, M. Mackay, M. Merabti, Links to the future: communication requirements and challenges in the smart grid. IEEE Power Energy Mag. **10**(1), 24–32 (2012)
3. M. Liserre, T. Sauter, J.Y. Hung, Future energy systems: integrating renewable energy sources into the smart power grid through industrial electronics. IEEE Ind. Electron. Mag. **4**(1), 18–37 (2010)
4. J.M. Carrasco, L.G. Franquelo, J.T. Bialasiewicz et al., Power-electronic systems for the grid integration of renewable energy sources: a survey. IEEE Trans. Industr. Electron. **53**(4), 1002–1016 (2006)
5. B. Wu, Y. Lang, N. Zargari, S. Kouro, *Power Conversion and Control of Wind Energy Systems*, 1st edn. (Wiley, Ltd-IEEE Press, 2011)
6. R. Teodorescu, M. Liserre, P. Rodriguez, *Grid Converters for Photovoltaic and Wind Power Systems*, 1st edn. (Wiley, Ltd-IEEE Press, 2011)
7. E. Romero-cadaval, G. Spagnuolo, L.G. Franquelo, C.A. Ramos-Paja, T. Suntio, W.M. Xiao, Grid-connected photovoltaic generation plants: components and operation. Ind. Electron. Mag. IEEE **3**(7), 6–20 (2013)
8. A. Emadi, S.S. Williamson, A. Khaligh, Power electronics intensive solutions for advanced electric, hybrid electric, and fuel cell vehicular power systems. IEEE Trans. Power Electron. **21**(3), 567–577 (2006)
9. S. Vazquez, S.M. Lukic, E. Galvan, L.G. Franquelo, J.M. Carrasco, Energy storage systems for transport and grid applications. IEEE Trans. Industr. Electron. **57**(12), 3881–3895 (2010)
10. S. Lukic, Charging ahead. IEEE Ind. Electron. Mag. **2**(4), 22–31 (2008)

Rotor Resistance Estimation for Improved Performance of MRAS-Based Sensorless Speed Estimation of Induction Motor Drives

Priya Misra and Bhavnesh Kumar

Nomenclature

v_s	three-phase input voltage
i_s	three-phase input current
V_{ds}, V_{qs}	d, q-axis voltage components of stator
i_{ds}, i_{qs}	d, q-axis current components of stator
$i_{\alpha s}$, $i_{\beta s}$	α, β-axis current components of stator
λ_{dr}, λ_{qr}	d, q-axis flux components of rotor
$\lambda_{r\alpha}$, $\lambda_{r\beta}$	α, β-axis flux components of rotor
$\bar{\lambda}$	estimated rotor flux in the adaptive model
L_{ss}, L_{rr}	stator and rotor self-inductance
M	mutual inductance
R_s, R_r	stator and rotor resistance
τ_r	$\frac{L_{ss}}{R_r}$; rotor time constant
σ	leakage factor
p	derivative operator
$\acute{\omega}_r$	estimated rotor angular speed
ϵ	speed adaption signal
R_r^e	estimated rotor resistance
$MRAS^U$	Updated MRAS model including rotor resistance estimator

P. Misra (✉) · B. Kumar
Department of ICE, Netaji Subhas University of Technology, New Delhi, India
e-mail: misrapriya12@gmail.com

B. Kumar
e-mail: kumar_bhavnesh@yahoo.co.in

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_18

1 Introduction

Cage-type induction motor is one of the most popular motors due to its advantageous features like robust design, less maintenance along with less cost. In general, induction motor drives use mechanical sensors or speed encoders mounted on the rotor shaft for control strategies used for determining the angular speed of the drive. The disadvantages of this approach are [1, 2]:

1. The price of the sensor may exceed the price of the motor itself in a low power range of drive.
2. Space constraints also arise in certain applications.
3. Additional wiring required for speed encoder may cause issues in certain applications such as vacuum compressors.
4. The added on wiring may also result in some interference.

Hence, there is a need for low maintenance, low cost, and robust electrical drives which resulted in the current popularity of AC induction motors. To address the reliability and maintenance issues, sensorless speed control is very effective. Therefore, a vector controlled system of induction motors without sensors for sensorless speed estimation is being continuously improved to obtain a robust model [3].

By the use of some new control strategies, like model reference adaptive control (MRAS), estimation and control of the rotor speed are possible [4–6]. Various schemes are constituted for realizing this method of speed estimation based on the quantity used to form the error vector that is the input to the adaptation mechanism, for example, rotor flux-based MRAS [7], back EMF-based MRAS [8], reactive power-based MRAS [9, 10], or stator current-based MRAS [11]. Rotor flux-based MRAS method is used in this chapter. These methods estimate the rotor speed by adjusting the gain of the system to reduce the error function produced due to the difference in the reference and adaptive model results as illustrated in Fig. 1. The majority of the models are machine parameter-dependent as they are constructed by using motor constants. The basic idea behind implementing the MRAS method is to control the state variable of the system using the error produced as a variation in the result of the

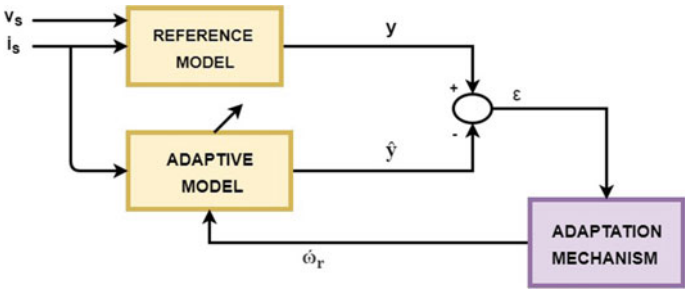


Fig. 1 Basic model for speed estimation based on the MRAS techniques

two models. The adaptive mechanism has a controller usually proportional–integral (P-I) type [12].

Machine parameters greatly affect these estimated speed, especially in the near-zero and low-speed region. Therefore, major issues occur in these speed estimation methods at near-zero and very low-speed regions and also in regenerative mode operation. The MRAS system suffers from the drawback that the accuracy of rotor speed measurement is directly affected by variation in system parameters. The major error is caused due to the variation in stator and rotor resistance, which is caused due to the variation in temperature [8]. A variation of 25 and 50% in rotor resistance is observed on the d-axis rotor flux, with a neural network-based adaptive model, proposed to minimize this effect [4]. Researchers have also tried to estimate the rotor resistance value along with speed estimation to improve the robustness of the drive [13]. A MRAS-based rotor resistance estimator is designed with its efficiency being tested by increasing the rotor resistance by 20% and observing the estimated rotor speed, having some error in the transient state but producing a satisfactory result in steady state [14].

In this chapter, the effect of variation in rotor resistance on the estimation of rotor speed and errors produced due to these variations are observed. Also, a rotor resistance computation block with an as accurate estimation as possible is designed to minimize the effect of these variations on our system.

2 Mathematical Model of MRAS Estimator

A rotor flux-based MRAS (RF-MRAS) speed estimator adaptation mechanism was presented in detail in the paper [2]. Schematic diagram of RF-MRAS is illustrated in Fig. 2. Final mathematical expressions of the estimator in d-q reference frame are as follows:

Reference model:

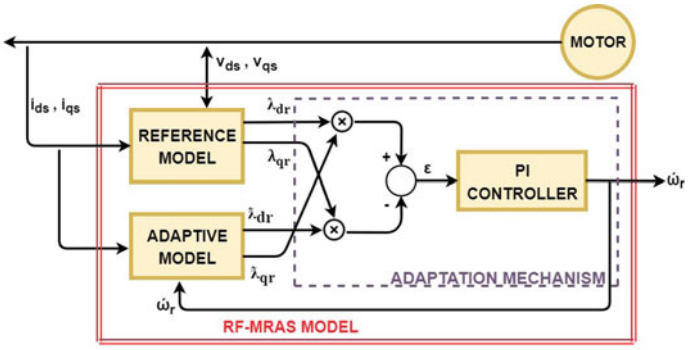


Fig. 2 Basic scheme of RF-MRAS estimator

$$\lambda_{dr} = \frac{L_{rr}}{M} \int [V_{ds} - (R_s + \sigma L_{ss} p) i_{ds}] \mathrm{d}t \tag{1}$$

$$\lambda_{qr} = \frac{L_{rr}}{M} \int \left[V_{qs} - (R_s + \sigma L_{ss} p) i_{qs}\right] \mathrm{d}t \tag{2}$$

Adaptive model:

$$p\bar{\lambda}_{dr} = -\omega_r \bar{\lambda}_{qr} - \frac{1}{\tau_r} \bar{\lambda}_{dr} + \frac{M}{\tau_r} i_{ds} \tag{3}$$

$$p\bar{\lambda}_{qr} = +\omega_r \bar{\lambda}_{dr} - \frac{1}{\tau_r} \bar{\lambda}_{qr} + \frac{M}{\tau_r} i_{qs} \tag{4}$$

For minimizing the error function for the rotor flux-based MRAS method given by Eq. (5), a PI controller is used as an adaptation mechanism.

$$\varepsilon = \lambda_{dr} \bar{\lambda}_{qr} - \lambda_{qr} \bar{\lambda}_{qr} \tag{5}$$

2.1 Effect of Rotor Resistance Variation

The rotor resistance of the machine varies with time due to the increase in temperature as the drive goes on operating. The heating effect caused due to copper losses of the machine results in the increase of temperature of the machine. Relation in heat energy produced in the drive due to copper losses incurring overtime can be expressed as:

$$\Delta H = i^2 R \Delta t \tag{6}$$

where ΔH is heat energy produced in the drive (joules), I is current flowing in the drive (amperes), R is equivalent resistance of the drive (ohms), and Δt is time elapsed since the drive started.

Heat energy produced increases the temperature, which in turn changes the resistance of the motor. Change in resistance can be calculated as:

$$R_r = R_{r0}(1 + \alpha \Delta T) \tag{7}$$

where R_r is the current rotor resistance of the motor, R_{r0} is the reference or initial rotor resistance, α is the temperature coefficient, and ΔT is the temperature change.

Hence, the rotor resistance of the motor may get changed with time as the drive operates. However, the MRAS speed estimator uses the reference rotor resistance only for the rotor speed estimation, which is included in the reference and adaptive

model equations of the estimator. This, thereby, causes the error in the rotor speed estimation.

2.2 Rotor Resistance Computation

The general scheme of the rotor resistance estimator considered in this research is shown in Fig. 3. The estimator uses the motor currents and rotor fluxes estimated in reference and adaptive model as inputs for rotor resistance estimation, and the estimated rotor resistance value is used as feedback for the adaptive model, forming a closed-loop structure for rotor resistance estimation.

The mathematical expressions used for computing the rotor resistance are [11, 14]:

$$R_r{}^e = \int A_3 \mathrm{d}t + A_4 \tag{8}$$

where

$$A_3 = K_3\left[\left(\frac{-\lambda_{r\alpha} + L_{ss} i_{\propto s}}{L_{rr}}\right)\varepsilon_\alpha + \left(\frac{-\lambda_{r\beta} + L_{ss} i_{\beta s}}{L_{rr}}\right)\varepsilon_\beta\right] \tag{9}$$

$$A_4 = K_4\left[\left(\frac{-\lambda_{r\alpha} + L_{ss} i_{\propto s}}{L_{rr}}\right)\varepsilon_\alpha + \left(\frac{-\lambda_{r\beta} + L_{ss} i_{\beta s}}{L_{rr}}\right)\varepsilon_\beta\right] \tag{10}$$

In the above expression, parameters K_3 and K_4 are positive constants and error vectors ε_α and ε_β are obtained as:

$$\varepsilon_\alpha = \lambda_{r\propto} - \lambda_{r\alpha} \tag{11}$$

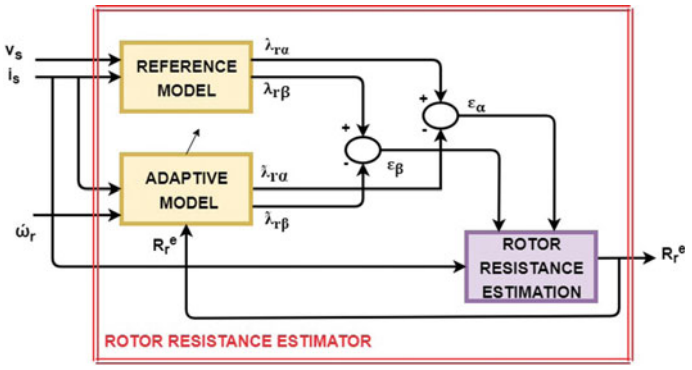


Fig. 3 Basic scheme of proposed rotor resistance estimator

$$\varepsilon_\beta = \lambda_{r\beta} - \bar{\lambda}_{r\beta} \tag{12}$$

2.3 Results and Discussion

Performance of the drive with and without rotor resistance variations is examined with the simulation model developed in MATLAB software.

Figure 4 shows the rotor speed plot for the nominal value of rotor resistance in the conventional MRAS model; hence, the estimated rotor speed gives a satisfactory result. Figure 5a shows the rotor speed plot for a 50% increase in rotor resistance in the conventional MRAS model. An error of about 2% can be observed in the estimated rotor speed. Figure 5b shows the rotor speed plot after the rotor resistance is incremented by 100% in the conventional MRAS model. An error of about 8% can be observed in the estimated rotor speed. This concludes that, with the increase in rotor resistance, the rotor speed estimation error increases for the conventional MRAS model.

Figure 6a gives the result of the proposed rotor resistance estimator included in MRAS^{U} model given in Sect. 2. The obtained result is able to track the reference rotor resistance satisfactorily with an error of about 0.03 at the steady state. However, the result shows high transients in the near-zero speed region and further work could be done to eliminate these transients.

Figure 6b shows the rotor speed plot for the MRAS^{U} model including the rotor resistance estimator. For the near-zero speed range, the rotor resistance estimator has been disabled because the complete system with speed and rotor resistance adaptation cannot provide stable results in that region. The estimated rotor speed is able to track the reference speed exactly.

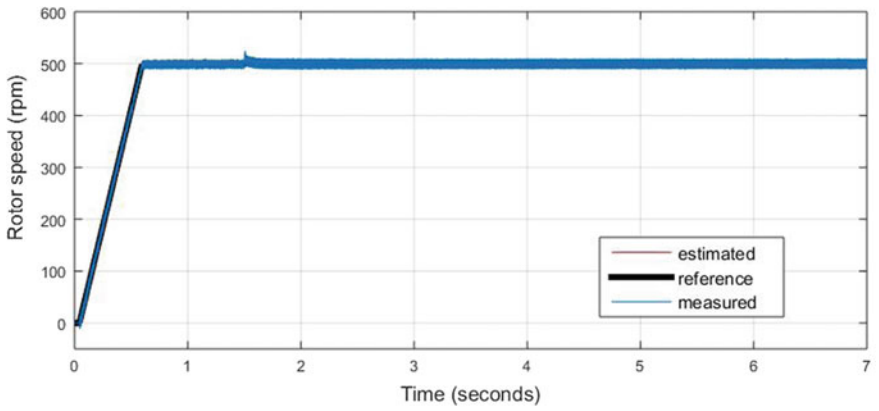


Fig. 4 Rotor speed for conventional MRAS model with nominal rotor resistance value

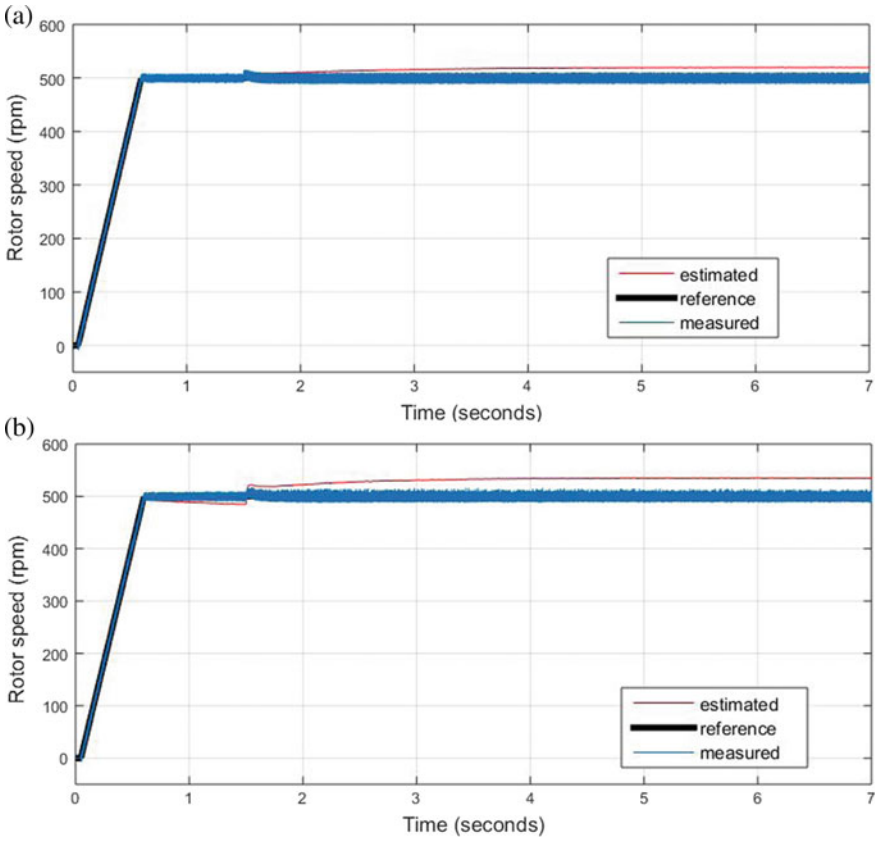


Fig. 5 Rotor speed for conventional MRAS model with **a** 50% **b** 100% increment in rotor resistance value

Figure 7a shows the result for the rotor resistance estimator for a 50% increment in the rotor resistance value. The estimated rotor resistance tracks the reference resistance satisfactorily with an error of about 0.05 at steady state. Figure 7b shows the rotor speed plot for the MRAS^{U} model with an increment in rotor resistance of 50%. For the near-zero speed range, the rotor resistance estimator has been disabled because the complete system with speed and rotor resistance adaptation cannot be stable in that region. Although the estimated rotor speed tracks the reference speed satisfactorily, certain ripples are observed in the result obtained. Also, we can compare the results of Figs. 5a and 7b to observe the advantage of including a rotor resistance estimator in the conventional MRAS model.

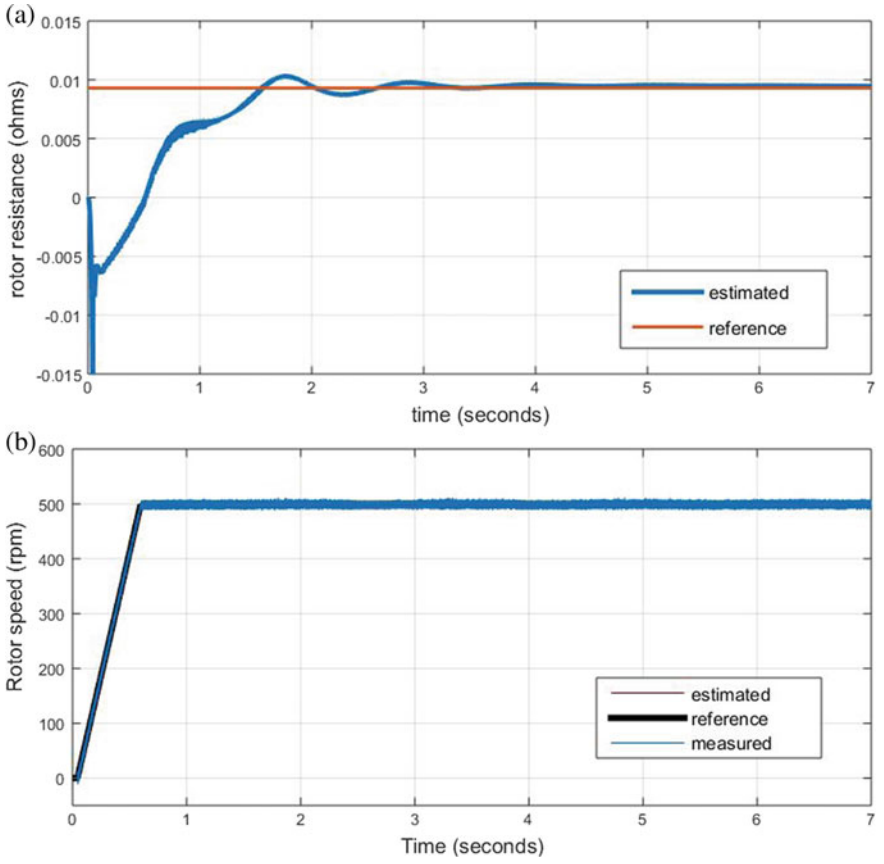


Fig. 6 **a** Estimated rotor resistance. **b** Estimated rotor speed with the proposed MRAS^{U} model

3 Conclusion

In this chapter, the conventional rotor flux-based MRAS method simulation and its results are studied. The effect of deviation of rotor resistance from its nominal value on the speed estimation and errors incurred due to this are also observed. To avoid any disturbance in rotor speed estimation due to variations in rotor resistance, an estimator for rotor resistance is designed with an accuracy of approximately 97% at steady state for nominal rotor resistance value and of about 95% for increased rotor resistance. The obtained result has significant transients in near-zero speed region, and future work can be done on this model so that the result obtained may have fewer oscillations in that region. Due to this, the rotor resistance estimator is required to be disabled in that region for rotor speed estimation in the updated MRAS model as it gives unstable results in that region. Also, rotor speed estimation with the updated

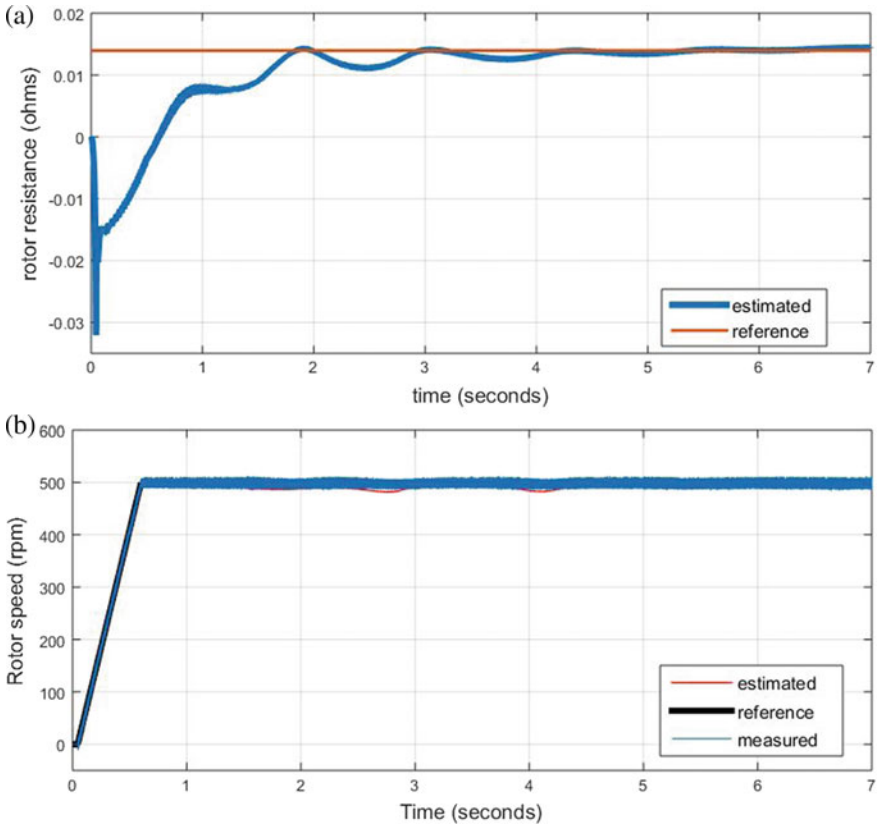


Fig. 7 **a** Estimated rotor resistance. **b** Estimated rotor speed with 50% increment in rotor resistance for MRAS^{U} model

MRAS model including the rotor resistance estimator is shown to give better results in comparison to the conventional MRAS model [15].

Appendix

See Tables 1 and 2.

Table 1 Motor ratings

Power = 149.2 KVA	Inertia = 3.1 kg m^2
Voltage (L-L) = 460 V	Friction factor = 0.08 N m s
Frequency = 60 Hz	Pole pairs = 2

Table 2 Parameters of induction motor drive

R_s	R_r	L_{ss}	L_{rr}	M
22.275mΩ	9.295mΩ	0.3027 mH	0.3027 mH	10.46 mH

Reference

1. W.P. Al-Bahadly, Sensorless speed control in induction motor drives, in *Proceedings First IEEE International Workshop on Electronic Design, Test and Applications*, vol. 1 (Christchurch, New Zealand, 2002) pp. 480–482
2. B. Kumar, T. Goyal, Investigations on flux estimation methods for stator current based MRAS speed estimator for induction motor drive, in *2018 IEEMA Engineer Infinite Conference (eTechNxT)* (New Delhi, 2018), pp. 1–5
3. P. Rao, J. Nakka, R. Shekar, Sensorless vector control of Induction machine using MRAS techniques, in *2013 International Conference on Circuits, Power and Computing Technologies (ICCPCT)* (Nagercoil, 2013), pp. 167–175
4. S.M. Gadoue, D. Giaouris, J.W. Finch, Sensorless control of induction motor drives at very low and zero speeds using neural network flux observers. IEEE Trans. Industr. Electron. **56**(8), 3029–3039 (2009)
5. M.S. Zaky, M.K. Metwaly, H.Z. Azazi, S.A. Deraz, A new adaptive SMO for speed estimation of sensorless induction motor drives at zero and very low frequencies. IEEE Trans. Industr. Electron. **65**(9), 6901–6911 (2018)
6. B.R. Teja, S. Mikkeli, Deployment of fuzzy based MRAS and IRFOC strategy with fast-tracking for the sensorless speed control of SPIM, in *2016 IEEE 6th International Conference on Power Systems (ICPS)* (New Delhi, 2016), pp. 1–6
7. S. Mohan Krishna, J.L. Febin Daya, MRAS speed estimator with fuzzy and PI stator resistance adaptation for sensorless induction motor drives using RT-lab, in *Perspectives in Science*, vol. 8 (2016), pp. 121–126
8. S.M. Gadoue, D. Giaouris, J.W. Finch, Stator current model reference adaptive systems speed estimator for the regenerating- mode low-speed operation of sensorless induction motor drives. IET Electric Power Appl. **7**(7), 597–606 (2013)
9. M. Ankarao, M. Vijaya Kumar, K. Trija, A comparative analysis of different MRAS schemes for speed sensorless induction motor drives employing PI and fuzzy controllers. Int. J. Appl. Eng. Res. ISSN **13**(18), 13898–13903 (2018)
10. K. Sedhuraman, S. Himavathi, A. Muthuramalingam, Neural learning adaptive system using simplified reactive power reference model based speed estimation in sensorless indirect vector controlled induction motor drives. Arch. Electric. Eng. **62**(1), 5–41 (2013)
11. M. Dybkowski, T. Orlowska-Kowalska, Speed sensorless induction motor drive system with MRAS type speed and flux estimator and additional parameter identification, in *11th IFAC International Workshop on Adaptation and Learning in Control and Signal Processing*, vol. 46, issue 11 (France, 2013), pp. 33–38
12. M.N. Gayathri, S. Himavathi, R. Sankaran, Performance of vector controlled induction motor drive with reactive power based MRAS rotor resistance estimator, in *2011 International Conference on Recent Advancements In Electrical, Electronics and Control Engineering* (Sivakasi, 2011), pp. 352–356
13. M.R. Dehbozorgi, H.M. Kojabadi, H. Vahedi, K. Al-Haddad, A comparative study of various MRAS-based IM's rotor resistance adaptation methods, in *IECON 2012—38th Annual Conference on IEEE Industrial Electronics Society* (Montreal, QC, 2012), pp. 4070–4075
14. Y. Agrebi Zorgani, Y. Koubaa, M. Boussak, Simultaneous estimation of speed and rotor resistance in sensorless ISFOC induction motor drive based on MRAS scheme, in *XIX International Conference on Electrical Machines, ICEM* (Rome, 2010), pp. 1–6

15. H. Li, W. Xuhui, C. Guilan, New general MRAS adaptive scheme to estimate stator and rotor resistance of induction motors, in *Conference Record of the 2006 IEEE Industry Applications Conference Forty-First IAS Annual Meeting* (Tampa, FL, 2006), pp. 1775–1780

Simple PID Controller Design for Non-minimum Phase FOPTD and SOPTD Processes

Ediga Chandramohan Goud, A. Seshagiri Rao, and M. Chidambaram

1 Introduction

For Proportional–integral–derivative (PID) controllers design and the dynamics of various processes are described by a FOPTD/SOPTD transfer function models. However, certain systems exhibit an inverse response. Such processes are defined by the transfer function model

$$\frac{y(s)}{u(s)} = \frac{K_P(-ps+1)e^{-\theta s}}{(1+\tau s)} \tag{1}$$

The method of designing PID controllers for the process given in Eq. (1) are given by Sun et al. [1], Jeng and Lin (2012), equating coefficient method (2003), etc.

If the initial response of a dynamic system is in the direction opposite to the steady state, it is known as an inverse response. Processes like boiler drum level, distillation column reboiler section, ammonium sulfate crystallizer, desalination process, and an exothermic tubular catalytic reactor, etc., exhibit non-minimum phase behavior. These classical examples are illustrated by Stephanopoulos [2], Ogunnaike and Ray [3] and Rathore et al. [4]. In case, the open-loop model has an RHP zero, then the model limits the value of the controller gain and hence the closed-loop model performance is limited. Methods of designing a PID controller for non-minimum phase processes have been suggested by many researchers. For the closed-loop process to be robust, the constraints on the gain margin (GM) as well as phase margin (PM) are as follows

E. C. Goud · A. Seshagiri Rao (✉) · M. Chidambaram
Department of Chemical Engineering, National Institute of Technology, Warangal 506004, India
e-mail: seshagiri@nitw.ac.in

R. Kumar et al. (eds.), *Intelligent Algorithms for Analysis and Control of Dynamical Systems*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-8045-1_19

$$\text{GM} \geq \frac{M_S}{(1 - M_S)} \tag{2}$$

$$\text{PM} \geq 2\sin^{-1}\left(\frac{0.5}{M_S}\right) \tag{3}$$

where the peak sensitivity (M_S) is given as

$$M_S = \max\left|\left[\frac{1}{(1 + KG(j)}\right]\right| \tag{4}$$

In the present work, the PID controller design for non-minimum phase FOPTD and SOPTD processes is proposed such that the open-loop transfer function is same for all the processes. This makes the peak sensitivity M_S same for all the processes.

2 Present Method

The given non-minimum phase system ($K_P(1 - ps)e^{-\theta s}/(1 + \tau s)$) is reduced by assuming $(1 - ps)$ as e^{-ps} and this delay is added to the already existed delay (θ) in the system and the effective delay is considered as L [=($\theta + p$)]. The non-minimum phase system is now reduced to $K_P e^{-Ls}/(1 + \tau s)$. In the PID controller design for the reduced model, a lag filter is added to compensate for the approximation. The value of the filter time constant is chosen based on the numerator time constant (p). The performance of the present controller is assessed in the original process. The output responses for both servo (y/y_r:y_r is set point) and load problems (y/y_d: y_d is load input) depend on the closed-loop characteristics equation ($1 + KG = 0 : KG$ is loop gain). The present method is derived from keeping a constant open-loop transfer function model. The present method is an extension of Haalman's method [5]. The controller parameter values for other transfer function models are obtained from this expression for KG. In the present work, integrating plus time delay model is considered as the basic model

$$K_P G_P = \frac{K_P e^{-Ls}}{s} \tag{5}$$

By minimizing the ISE value for a servo problem, the controller parameter values for the $K_C G_C$ are reported by Visioli and Zhong [6] as

$$K_C = 1/(K_P L)\tau_I = \infty, \tau_D = 0.5L \tag{6}$$

which is a PD controller. The loop gain $KG = K_P G_P K_C G_C$. The genetic algorithm is used to minimize ISE values to get the controller parameters as above. GA method is

time-consuming one. The same controller mode and the settings of the PD controller are obtained by Sai Varun and Padma Sree [7] by a simple analytical method (i.e., the equating coefficient method). The above method has been extended to the non-minimum phase stable processes. Consider the non-minimum phase integrating time delay process as

$$K_P G_P = \frac{K_P(1 - ps)\mathrm{e}^{-\theta s}}{s} \approx \frac{K_P \mathrm{e}^{-Ls}}{s} \tag{7}$$

The controller parameters for the $K_p G_p$ is PD controller with lag filter $\frac{K_C(1+\tau_D s)}{(\tau_F s+1)}$ and given by

$$K_C = 1/(K_P L),\ \tau_D = 0.5L \tag{8}$$

τ_F is a tuning parameter and is selected as 0.6p to 1.2p. As the tuning parameter increases the performance improves, but robustness decreases and variation in the manipulated variable increases and vice versa.

Hence, KG is obtained as $\left[K_P \mathrm{e}^{-Ls}/s\right] K_C(1 + \tau_D s)[\frac{1}{(\tau_F s+1)}]$. The expression for KG is retained as same for all other processes in this work. The expression of $K_C G_C$ for a non-minimum phase FOPTD model as follows

$$K_C G_C e^{-Ls}/[(\tau s + 1)] = \left[K_P \mathrm{e}^{-Ls}/s\right][1/(K_P L)][1 + 0.5Ls][\frac{1}{(\tau_F s + 1)}] \tag{9}$$

$$K_C G_C = \frac{[(1 + 0.5Ls)(1 + \tau s)]}{K_P Ls}[\frac{1}{(\tau_F s + 1)}] = K_C[1 + \{1/s\tau_I\} + \tau_D s][\frac{1}{(\tau_F s + 1)}] \tag{10}$$

where

$$K_C = [(0.5L + \tau)/(K_P L)]\tau_I = (0.5L + \tau) \text{and} \tau_D = [(0.5L\tau)/(0.5 + \tau)] \tag{11}$$

The method has the tuning parameter τ_F. Similarly, the controller for the non-minimum phase SOPTD model $K_P \mathrm{e}^{-Ls}/[(\tau_1 s + 1)(\tau_2 s + 1)]$ is obtained as

$$\begin{aligned} K_C G_C &= [1/(K_P L)]\left[\frac{(\tau_1 s + 1)(\tau_2 s + 1)(0.5Ls + 1)}{s}\right][\frac{1}{(\tau_F s + 1)}] \\ &= K_C[(\tau_1 s + 1)(0.5Ls + 1)/s][(\tau_2 s + 1)/(\tau_F s + 1)] \end{aligned} \tag{12}$$

The PID settings with lead-lag filter $[(s + 1)/(s + 1)]$ are given by

$$K_C = [(\tau_1 + 0.5L)/(K_P L)]\tau_I = (\tau_1 + 0.5L)\tau_D = \left[\tau_1(0.5L)/(\tau_1 + 0.5L)\right], = \tau_2, = \tau_F \tag{13}$$

3 Simulation Case Studies

In the simulation study, two case studies are presented and compared with the recent literature methods.

3.1 Case Study 1

Consider a non-minimum phase FOPTD with K_P =1; $\tau = 1$; $\theta = 0.2$; and $p = 1$

$$G_p(s) = \frac{(1-s)\mathrm{e}^{-0.2s}}{(1+s)}$$

The above approximate FOPTD model is given by $K_p\mathrm{e}^{-Ls}/(1+\tau s)$ with $L = 1.2$. The present method PID controller with filter parameters is $K_C = 0.933$, $\tau_I = 1.6$, $\tau_D = 0.375$, $\tau_F = 0.6$, Sai Varun and Padma Sree [7] PID controller with filter parameters is $K_C = 0.3021$, $\tau_I = 0.9977$, $\tau_D = 0.09$, $\tau_F = 0.0409$ and Shamsuzzoha et al. [11]. PI controller parameters are $K_C = 0.314$, $\tau_I = 0.418$. A unity step applied in the setpoint at $t = 0$ s, the closed-loop output response, and the control action are displayed in Fig. 1. The controlled variable behavior is also shown in Fig. 1. The closed-loop load/regulatory responses are presented in Fig. 2. The present method provides better performance both for servo and load problems. The evaluated closed-loop servo and load performance (IAE and TV), stability (GM, PM) and robustness values (M_S) are given in Table 1.

3.2 Case Study 2

The system reported by Bo et al. (2014) is chosen (an optimal analytical solution internal model control (IMC) theory) with $p = 3$, $k_P = 2$, $\tau_1 = 2$; $\tau_2 = 1$, $\theta = 0.5$.

$$G_p(s) = \frac{(-3s+1)\mathrm{e}^{-0.5s}}{(s+1)(s+0.5)}$$

The given process is reduced to that of the system without any zero $2\mathrm{e}^{-3.5s}/(s+1)(2s+1)$. The present method PID controller with lead-lag filter settings are $K_C = 0.3$, $\tau_\mathrm{I} = 3$, $\tau_\mathrm{D} = 0.6667$, $= 1.75$, $= 1.8$, Jeng and Lin [8] proposed settings are $K_C = 0.191$, $\tau_I = 3.027$, $\tau_D = 0.67$, $= 0.067$, Chen and Seborg [9] proposed settings are $K_C = 0.209$, $\tau_I = 3$, $\tau_D = 0.667$, $= 0.079$, Chien et al. [10] proposed controller settings are $K_C = 0.1561$, $\tau_I = 2$, $\tau_D = 1$ and evaluated in the original process. The present method provides the best performance for load rejection. The method by Chen and Seborg [9] gives the best closed-loop

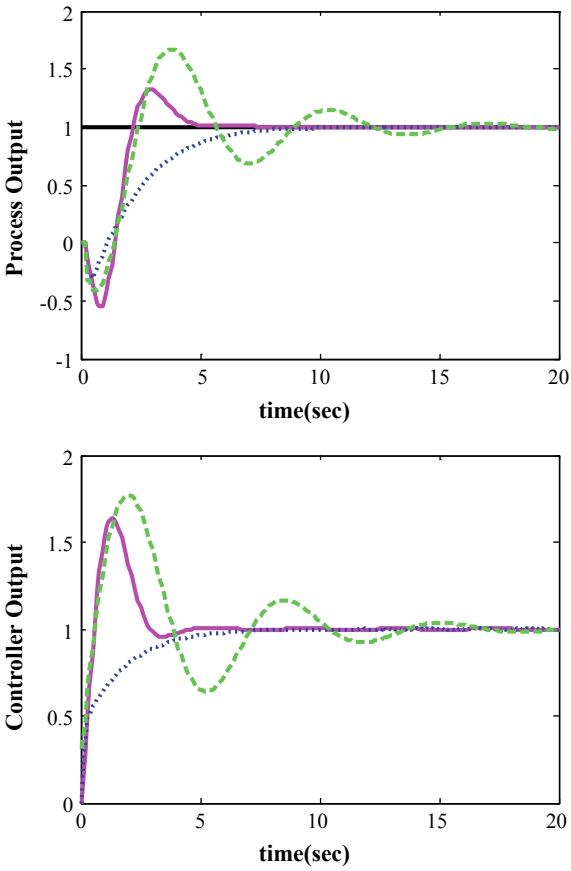


Fig. 1 Closed-loop servo response (above) and corresponding controller response (below) for the case study 1; dot-Sai Varun and Padma Sree [7], Shamsuzzoha et al. [11]. solid-present

servo performance. The servo responses are displayed in Fig. 3, and the regulatory responses are displayed in Fig. 4 along with the manipulated variable behavior. Stability, maximum sensitivity, and performance values are presented in Table 2.

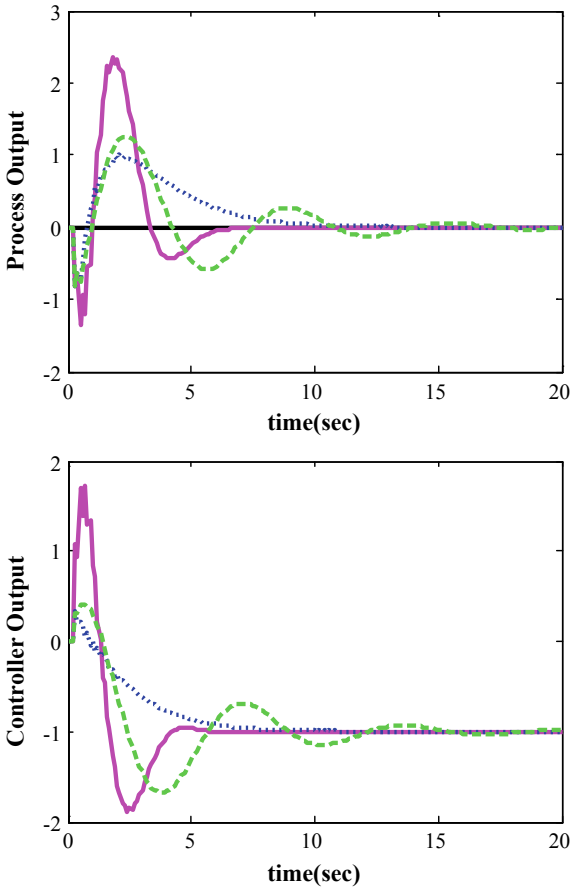


Fig. 2 Closed-loop regulatory response (above) and corresponding controller response (below) for the case study 1; dot-Sai Varun and Padma Sree [7]; dash-Shamsuzzoha et al. [11]; solid-present

Table 1 Stability, robustness, and performance values for case study 1

Methods	GM	PM	M_S	Servo response		Regulatory response	
				IAE	TV	IAE	TV
Present method	1.56	46.1	2.8	0.27	2.38	0.465	7.35
Sai Varun and Padma Sree [7]	1.51	68.3	2.6	0.33	1.15	0.408	2.28
Shamsuzzoha et al. [11]	1.49	22.5	3.9	0.48	3.54	0.513	4.33

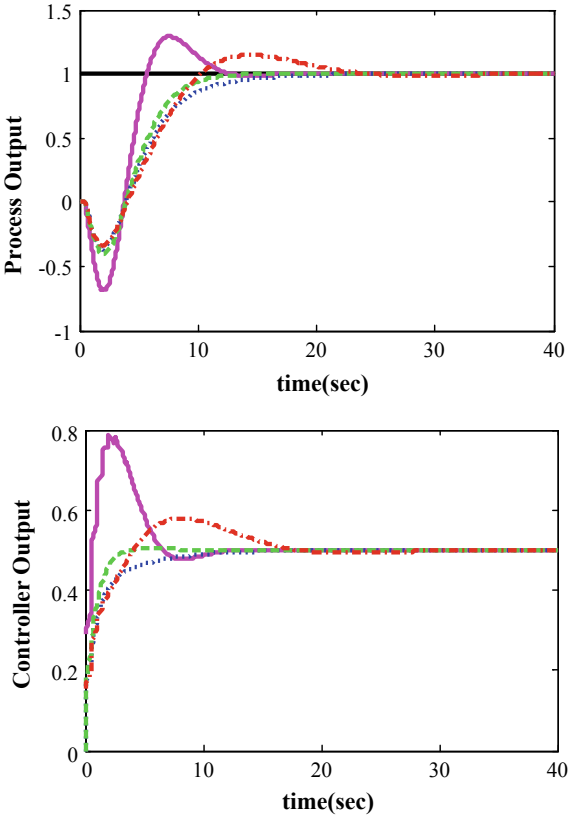


Fig. 3 Closed-loop servo response (above) and corresponding controller response (below) for the case study 2. dot-Jeng and Lin [8], dash-Chen and Seborg [9]; dash-dot-Chien et al. [10]; Solid-present

4 Conclusions

A method is proposed for the controller design for the stable non-minimum phase FOPTD and SOPTD models. In the design, a first-order filter is used for all case studies. The present method gives a simple analytical expression for the controller settings as a function of the process parameters. The controller (TV) performance, process performance (IAE), stability, and robustness of the present controller are compared with the recent methods.

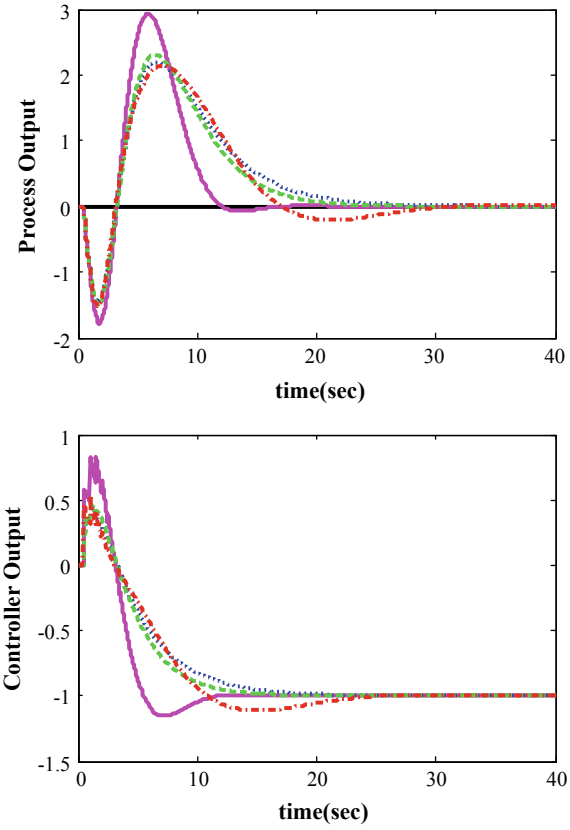


Fig. 4 Closed-loop servo response (above) and corresponding controller response (below) for the case study 2. dot-Jeng and Lin [8], dash-Chen and Seborg [9]; dash-dot-Chien et al. [10]; Solid-present

Table 2 Stability, robustness, and performance values for case study 2

Methods	GM	PM	M_S	Servo response		Regulatory response	
				IAE	TV	IAE	TV
Present method	1.56	45.6	2.83	7.21	0.89	16.59	3.94
Jeng and Lin [8]	2.39	63.5	1.74	7.93	0.5	20.95	1.875
Chen and Seborg [9]	2.18	60.2	1.87	7.185	0.51	19.61	1.875
Chien et al. [10]	2.14	50.9	1.874	8.722	0.517	21.135	2.92

References

1. B. Sun, W. Zhang, Z. Li, Optimal H2 input load disturbance rejection controller design for nonminimum phase systems based on algebraic theory. Ind. Eng. Chem. Res. **53**(4), 1515–1528 (2014)
2. G. Stephanopoulos, *Chemical Process Control* (PTR Prentice Hall, Englewood Cliffs, N.J., 1984)

3. B.A. Ogunnaike, Process dynamics, modeling, and control. No. 660.28 O48 (1994)
4. N.S. Rathore, V.P. Singh, B.D.H. Phuc, A modified controller design based on symbiotic organisms search optimization for desalination system. J. Water Supply Res. Technol. Aqua **68**(5), 337–345 (2019)
5. A. Haalman, Adjusting controllers for a deadtime process. Control Eng. 71–73 (1965)
6. A. Visioli, Q.-C. Zhong, PID control, in Control of integral processes with dead time (Springer, London, 2011), pp. 9-47
7. A. Sai Varun, R. Padma Sree, Tuning of PID controllers for first order stable/unstable time delay systems with a zero. IJCTA**8** (2015)
8. J. Jeng, S. Lin, Robust proportional-integral-derivative controller design for stable/integrating processes with inverse response and time delay. Ind. Eng. Chem. Res. **51**(6), 2652–2665 (2012)
9. D. Chen, D.E. Seborg, PI/PID controller design based on direct synthesis and disturbance rejection. Ind. Eng. Chem. Res. **41**, 4807−4822 (2002)
10. I.L. Chien, Y.C. Chung, B.S. Chen, C.Y. Chuang, Simple PID controller tuning method for processes with inverse response plus dead time or large overshoot response plus dead time. Ind. Eng. Chem. Res. **42**(20), 4461–4477 (2003)
11. M. Shamsuzzoha, Robust PID controller design for time-delay processes with a peak of maximum sensitivity criteria. J. Cent. S. Univ. **21**(10), 3777–3786 (2014). https://doi.org/10.1007/s11771-014-2362-0