AFRICAN BUFFALO OPTIMIZATION APPROACH TO THE DESIGN OF PID CONTROLLER IN AUTOMATIC VOLTAGE REGULATOR SYSTEM

Julius Beneoluchi Odili
Faculty of Computer Systems and Software Engineering,
Universiti Malaysia Pahang,
Gambang, Kuantan 26300, Malaysia
Odili_julest@yahoo.com

Mohd Nizam Mohmad Kahar
Faculty of Computer Systems and Software Engineering,
Universiti Malaysia Pahang,
Gambang, Kuantan 26300, Malaysia

Abstract—: In the past few decades, researchers have focused so much attention on the application of bio-inspired techniques for solving real-life problems in science, engineering and industrial processes. This paper presents the African Buffalo Optimization (ABO) approach to tuning the Proportional, Integral and Derivative (PID) parameters of an Automatic Voltage Regulator. The ABO is a simulation of the movement of African Buffalos in the vast African landscape in search of lush grazing pastures using two basic sounds, namely the waaa and the maaa vocalizations. The ABO has so far been effective and efficient in providing solutions to a number of optimization problems as a result of its use of relatively fewer parameters, straight-forward fitness function as well as regular communications using the star communication topology. Simulation results from the application of the ABO to tuning the parameters of an AVR has proven to be very competitive when compared with the results of other bio-inspired optimization techniques such as the GA-PID, ACO-PID, PSO-PID, BFO-PID, PID-PSO and PID Tuner.

Keywords— African Buffalo Optimization, Proportional, Integral and Derivative, Tuning, Automatic Voltage Regulator

1. INTRODUCTION

Optimization is at the heart of the biggest scientific breakthroughs that the world experienced from the 19^{th} century till date. Optimization basically is the economics of science, technology and engineering as it emphasizes the need to achieve optimal results with minimum input. Optimization has wide applications in every aspect of design and decision-making ranging from industrial designs and production to business management, political decision-making to pharmacy and bio-technology, bio-medical engineering etc.

The overwhelming popularity of optimization has given rise to the design and development of a number of optimization algorithms such as the Simulated Annealing (SA) [1], Genetic Algorithm [2], Particle Swarm Optimization [3], Ant Colony Optimization [4], African Buffalo Optimization [5, 6] etc. Application areas of the these bio-inspired techniques include

Travelling Salesman's Problems [7, 8], administration [9], computer networking [10], numerical function optimization [5], Proportional, Integral and Derivative (PID) parameters tuning of Automatic Voltage Regular (AVR) [11] etc.

One of the first successful methods of tuning of PID parameters was the Ziegler and Cohen's method. The traditional tuning methods like the Ziegler Nichols PID and the Cohen-Coon PID requires that the process models should be minimised so as to become less complex [12]. If this concern is not appropriately addressed, there cannot be satisfactory tuning, thus leading to system overshoot, system-delays, steady-state errors and eventual system instability. This way, the primary objective of using a PID controllers which is to ensure the effective and efficient tuning of parameters is defeated. Moreover, Ziegler and Cohen's method requires lots of technical expertise to ensure appropriate tuning, even though it does not guarantee satisfactory performance. These weaknesses drove researchers to investigate the possibility of metaheuristic tuning of PID parameters leading to the design and development of Genetic Algorithm PID (GA-PID) [13], Particle-Swarm Optimization PID (PSO-PID) [14], Ant Colony Optimization PID (ACO-PID) [15], PID-Tuner [16], Bacteria-Foraging Optimization PID (BFO-PID) [17] and now the African Buffalo Optimization (ABO-PID). The popularity of metaheuristic tuning of PID paramters in systems control results from their ease of use, simplicity of operation, ease of maintenance, low cost, ease of implementation, dynamism, effectiveness and efficiency .

This study presents the application of the African Buffalo Optimization algorithm to tuning the PID parameters of an AVR. The study highlights the ease with which the ABO tunes the PID parameters of an AVR to ensure effectiveness and stability of the tuning process.

The rest of this paper is organzed thus: section two discusses the PID; section three is concerned with a discussion on AVR and the effects of PID on an AVR and section four discusses the experiemental results. This is followed by the acknowledgemnet of support for the study and then references.

2. PROPORTIONAL-INTEGRAL-DERIVATIVE

Proportional-Integral-Derivative (PID) controller is a generic control loop feedback technique that is used in industrial control systems. The PID works optimally in systems with accurate mathematical models. The PID controller calculates involves three separate parameters, namely, proportional, integral and derivative coefficients. The proportional component of the PID computes the value of the current error. Similarly, the integral component determines the result of the sum of recent errors and the derivative component determines the reaction based on the rate at which the errors have been changing. The weighted sum of these three actions is imported into the control system.

Figure 1: Picture of an AVR type [18]

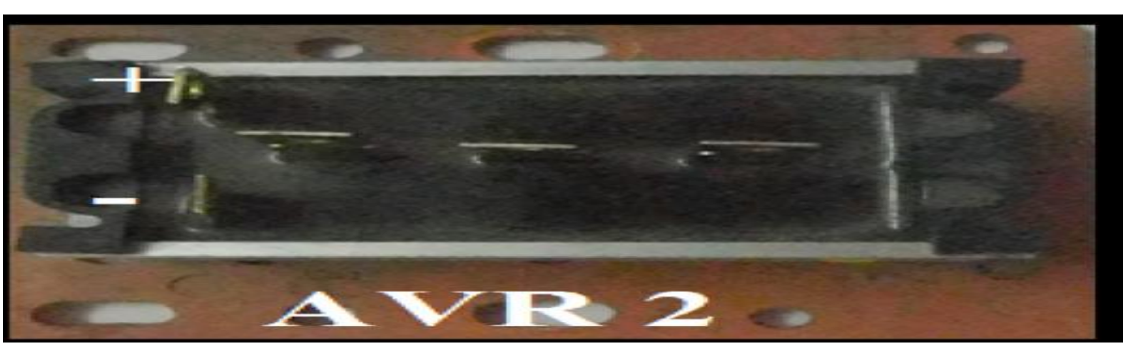

Figure 2: Picture of another type of an AVR

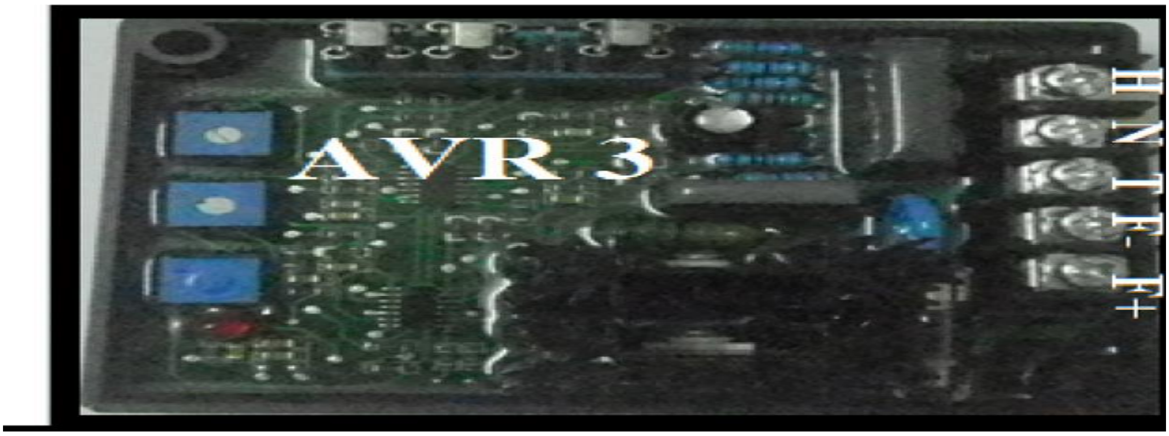

Figure 3: Picture of a third type of AVR [18]

A. Effects of PID on an AVR

The primary duty of a PID controller is to control the dynamic response of the AVR in addition to reducing or completely eliminating the steady-state error. The PID has three major parts: the Proportional, Integral and Derivative components. The duty of the Proportional component (G_p) of a PID controller system is to minimize the rise time of the power system. It is also helpful of reducing the steady state error but quite incapable of totally eliminating the steady-state error. The primary duty of the integral component (G_i) is to reduce or possibly eliminate the steady-state error for a step input. This component also helps in slowing down the transient response of the power system. Finally, the main function of the derivative control (G_d) component is to increase the system stability by reducing or completely eliminating the system overshoot. This way, the derivative helps to improve the transient response of the system. The cumulative effect of the PID on a closed loop system is represented diagrammatically in Table 1. It is worthy of note, however, that there could be slight changes to the effects listed in Table 1 in some special circumstances.

3. ABO FOR PID

In testing the performance of the ABO-PID, an AVR system with a PID controller was implemented. The primary benefit of an AVR is to ensure a constant voltage level. Regulating a voltage may require a feed-forward system as in 'open loop' mechanisms or a closed loop system as in 'feedback' systems. A feedback system is designed in a way to send back some feedback with information on the actual output so as to ensure that that the predetermined output is same as originally desired. The AVR may employ an electronic mechanism or an electromechanical component in its operation and has the capacity to regulate one or more DC or AC voltages. A basic AVR is made up of the amplifier, exciter, generator and sensor, all working in harmony to ensure the output voltage of the power system is at a specified range. It is to be noted that an increase in the generator reactive power load usually leads to a drop in the terminal voltage. A PID controller is, therefore, useful in minimizing the error and ensuring improved dynamic response. As a result of this, the stability or otherwise of the AVR system seriously affects the security of the power system. The Block diagram of an ABO-PID is presented in Figure 3.

The function of these AVR components can be can be represented linearly and represented mathematically as:

$$\text{Amplifier } (G_A) = VR(s) \,/\, Ve(s) \;=\; G_A \;\;/\;\; 1 + \tau_A S \tag{1}$$

where G_A is the Amplifier gain and τ_A is the Time constant in the S domain. Typical values of G_A are in the range of 10 to 400. The amplifier time constant is very small ranging from 0.02 to 0.1 s. The transfer function of a modern exciter may be represented by a gain $G_E\, K_E$ and a single time constant

Table 1: Effects of a PID on a closed loop system

RESPONSE	OVERSHOOT	S-S ERROR	RISE TIME	SETTLING TIME
G_p	Increase	Decrease	Decrease	Small Change
G_i	Increase	Eliminate	Decrease	Increase
G_d	Decrease	No Change	Small Change	Decrease

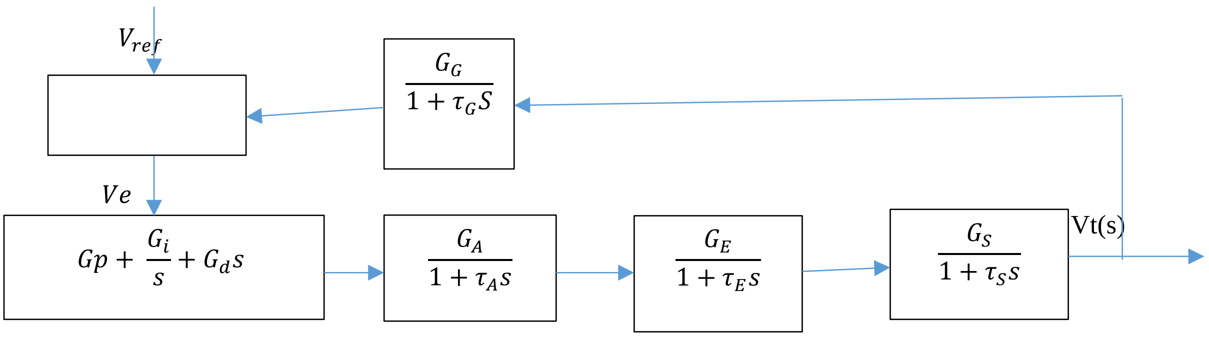

Figure 3: Block diagram of AVR system with a PID.

$$\text{Exciter} = VF(S)/\,VR\,(S) = G_E\,/1 + \tau_E(S) \quad (2)$$

Here, K_E represents Exciter gain and $\tau_E(S)$ Time constant in the S domain. Typical values of G_E are in the range of 10 to 400. The time constant is in the range of 0.5 to 1.0 s.

Similarly, the transfer function relating the generator terminal voltage to its field voltage can be represented by a gain G_G and a time constant

$$\text{Generator} = \quad Vt\,(S)\,/\,Vf\,(S) = G_G\,/1 + \tau_G(S) \quad (3)$$

These constants are load dependent, G_G may vary between 0.7 and 1.0, and between 1.0 and 2.0 s from full load to no load. Finally, the sensor is modelled by a simple first-order transfer function, given by

$$\text{Sensor} = \quad Vs\,(S)\,/Vt\,(S) = G_S\;\;/\,1 + \tau_S\,(S) \quad (4)$$

τ_S is usually very small, ranging from of 0.001 to 0.06 s.

4. EXPERIMENTS AND DISCUSSION OF RESULTS

ABO-PID was implemented using a MATLAB code and executed using MATLAB Compiler 2012b on a PC: Intel Duo Core i7-3770CPU, 3.40 Ghz with 4GBRAM. The parameters are presented in Table 2:

Table 2: PID tuning parameters

Parameters	Values
G_A	10
G_E	1.0
G_G	1.0
G_S	1.0
τ_A	0.1
τ_E	0.4
τ_G	1.0
τ_S	0.01

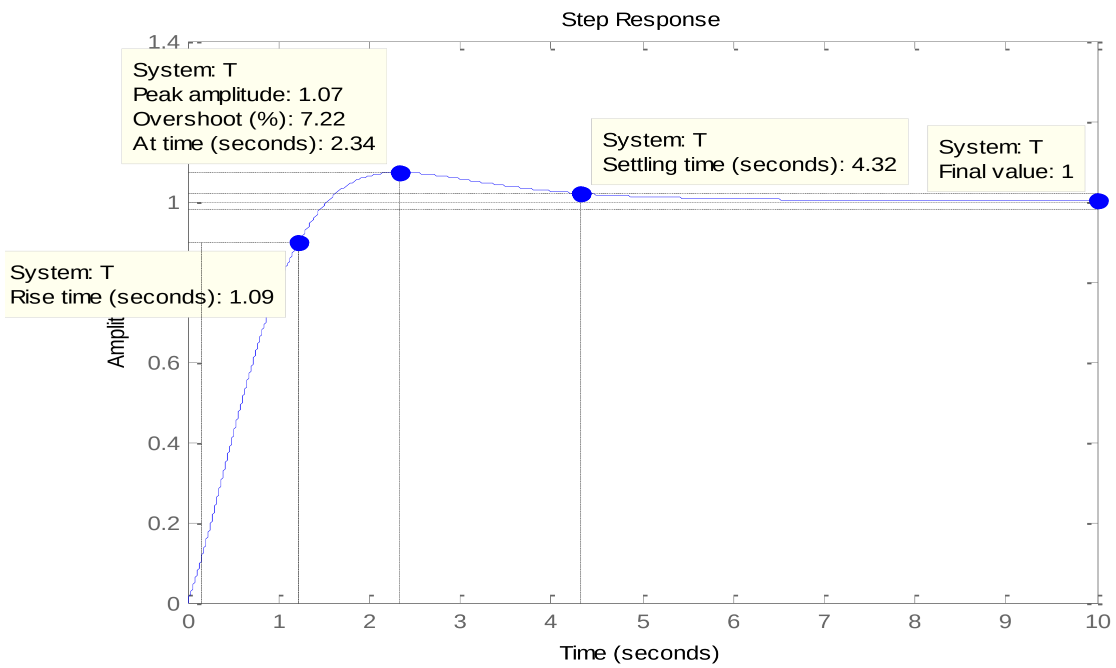

Figure 4: ABO-PID on time domain

It should be noted that the variable Ve in the Block diagram (Figure 3) is obtained by Equation 5

$$Ve = Vt(s) - Vref\ (s) \tag{5}$$

In the above Equation represents, Ve represents the tracking error and it is obtained by subtracting the reference (input) signal ($Vref\ (s)$) from the output signal ($Vt(s)$). The error signal (Ve) is then sent to the PID controller whose duty it is to calculate the proportional, the integral and the derivative of this error signal.

Similarly, The PID controller transfer function is:

$$Gp + Gi/s\frac{Gi}{s} + Gd\ (s) \tag{6}$$

Table 3. Simulation Results

Gain overshoot (%)	**Type of controller**	**Parameters**			**Rise time (secs)**	**Settling time (secs)**
		Gp	Gi	Gd		
7.22	ABO-PID	4.1380	2.0672	0.5779	1.090	4.320
8.99	PID_PSO	0.6125	0.4197	0.2013	0.684	3.087
3.80	PID_TUNER	0.2736	.1723	0.1150	1.360	4.400
36	LQR-PID	1.0100	0.5000	0.1000	0.500	2.335
5	BC-GA	0.5692	0.2484	0.1258	0.893	1.7019
.004	RC-GA	0.6820	0.2660	0.1790	1.0668	1.2682
0	FO-PSO-PID	0.1700	0.0300	0.0140	27.00	52.00
60	GA-PID	3.1563	0.9463	0.4930	0.493	8.900
60.5	ACO-PID	2.9917	1.1053	0.3085	0.493	7.100
59	PSO-PID	3.3172	0.8993	0.2814	0.4993	10.200
62.1	BFO-PID	3.0725	1.1054	0.2601	0.4993	6.800

Also, the transfer function of the AVR components is:

$$\frac{\Delta Vt\ (s)}{\Delta Vref\ (s)} = \frac{(S^2\ Gd + S\ Gp + Gi)\ (K_A\ \ K_E\ \ K_{G}\ K_S)\ (1 + S\ T_S)}{S(1+S\ \tau_A\ )\ (1+S\ \tau_E)\ (1+S\ \tau_G)\ (1+\ S\tau_S)\ (S^2\ Gd + S\ Gp + Gi)\ (K_A\ \ K_E\ \ K_G\ K_S)} \tag{7}$$

The ABO-PID was implemented in a MATLAB code, executed using a MATLAB2012b compiler. The result of this experiment is compared with results from other optimization algorithms such as the GA, PSO, ACO, BFO [19], PID-PSO, PID-tuner and LQR tuner [20]. The result is presented in Table 3:

Table 3 presents an interesting fact that authenticates the No Free Lunch theorem of optimization algorithms [21]. The Table highlights the strengths and weaknesses of each of the comparative algorithms to the extent that a parameter that is of particular interest to a researcher will determine his choice of an algorithm. Using the percentage gain overshoot as a benchmark, it could be seen that this varies from 0% for FO-PSO-PID to 0.004% for Real-coded Genetic Algorithm; 3.8% for PID-tuner; 5% for Binary-coded Genetic Algorithm; 7.22% for ABO-PID, 8.99% for PID-PSO, 36% for LQR-PID; 59% for PSO-PID, 60% for GA-PID,; 60.5% for ACO-PID and 62.1% for BFO-PID.

Similarly, in terms of the rise time of the generating set, similar trend can be observed with the rise time ranging from 0.4993 second (GA-PID, ACO-PID, PSO-PID, BFO-PID) to 0.5 second (LQR-PID); 0.684 second (PID-PSO); 0.893 second (BC-GA); 1.0668 seconds (RC-GA); 1.09 (ABO-PID); 1.36 seconds (PID-tuner) and 27 seconds to FO-PSO-PID.

Also, the settling time parameter is not very different from the gain percentage overshoot and the generator rise time. In terms of the generator's settling time, RC-GA had the best settling time of 1.2682 seconds, followed by BC-GA (1.7019 seconds), LQR-PID (2.335 seconds), PID-PSO (3.087 seconds), ABO-PID (4.320 seconds), PID-tuner (4.400 seconds), BFO-PID (6.800 seconds), ACO-PID (7.100 seconds), GA-PID (8.900 seconds), PSO-PID (10.200 seconds) and, finally, FO-PSO-PID (52.00 seconds).

4. CONCLUSION

From the foregoing analysis, it can be adduced that a good tuner is one that does a fair trade off in balancing the performance of different parameters in its quest to obtain effective and efficient tuning. Of special interest is the performance of FO-PSO-PID which had no gain overshoot but has the slowest rise time and setting time. That is to say that it sacrificed the rise time and settling time in its attempt to have 0% gain overshoot. The strength of the ABO-PID stands out in its ability to maintain a good balancing of the power generating parameters, thus ensuring harmonious, effective and efficient working of the AVR system.

ACKNOWLEDGMENT

The authors appreciate the Faculty of Computer Systems and Software Engineering, Universiti Malaysia Pahang, Kuantan 26300, Malaysia for providing funding for this study through Grant GRS 1403118

REFERENCES

[1] K. A. Dowsland and J. M. Thompson, "Simulated annealing," in *Handbook of Natural Computing*, ed: Springer, 2012, pp. 1623-1655.

[2] M. H. Moradi and M. Abedini, "A combination of genetic algorithm and particle swarm optimization for optimal DG location and sizing in distribution systems," *International Journal of Electrical Power & Energy Systems,* vol. 34, pp. 66-74, 2012.

[3] M. A. Kunna, T. A. A. Kadir, A. S. Jaber, and J. B. Odili, "Large-Scale Kinetic Parameter Identification of Metabolic Network Model of E. coli Using PSO," *Advances in Bioscience and Biotechnology,* vol. 6, p. 120, 2015.

[4] J. B. Odili, "Application of Ant Colony Optimization to Solving the Traveling Salesman's Problem," *Science Journal of Electrical & Electronic Engineering,* vol. 2013, 2013.

[5] J. B. Odili and M. N. M. Kahar, "African Buffalo Optimization (ABO): a New Meta-Heuristic Algorithm." *Journal of Advanced and Applied Sciences,* vol. 3(3), 2015, pp.101-106.

[6] J. B. Odili, M. N. M. Kahar, and S. Anwar, "African Buffalo Optimization: A Swarm-Intelligence Technique," *Procedia Computer Science,* 2015, vol. 76, pp.443-448.

[7] J. B. Odili and M. N. M. Kahar, "Solving the Traveling Salesman’s Problem using the African Buffalo Optimization" *Computational Intelligence and Neuroscience,* vol.2016, 2016, pp. 1-12.

[8] J. B. Odili, M. N. M. Kahar, S. Anwar, and M. A. K. Azrag, "A comparative study of African Buffalo Optimization and Randomized Insertion Algorithm for asymmetric Travelling Salesman's Problem," in *Software Engineering and Computer Systems (ICSECS), 2015 4th International Conference on*, 2015, pp. 90-95.

[9] J. B. Odili, "Staff development programs and job performance: Implications for productivity in Lagos state ministry of education ", 2013, pp. 330-334.

[10] S. Anwar, J. M. Zain, M. F. Zolkipli, Z. Inayat, A. N. Jabir, and J. B. Odili, "Response option for attacks detected by intrusion detection system," in *Software Engineering and Computer Systems (ICSECS), 2015 4th International Conference on*, 2015, pp. 195-200.

[11] P. Agrawal and S. Lakshminarayanan, "Tuning proportional-integral-derivative controllers using achievable performance indices," *Industrial & engineering chemistry research,* vol. 42, pp. 5576-5582, 2003.

[12] M. Korkmaz, Ö. Aydoğdu, and H. Doğan, "Design and performance comparison of variable parameter nonlinear PID controller and genetic algorithm based PID controller," in *Innovations in Intelligent Systems and Applications (INISTA), 2012 International Symposium on*, 2012, pp. 1-5.

[13] M. J. Neath, A. K. Swain, U. K. Madawala, and D. J. Thrimawithana, "An optimal PID controller for a bidirectional inductive power transfer system using multiobjective genetic algorithm," *Power Electronics, IEEE Transactions on,* vol. 29, pp. 1523-1531, 2014.

[14] M. I. Solihin, L. F. Tack, and M. L. Kean, "Tuning of PID controller using particle swarm optimization (PSO)," *International Journal on Advanced Science, Engineering and Information Technology,* vol. 1, pp. 458-461, 2011.

[15] M. Ünal, A. Ak, V. Topuz, and H. Erdal, *Optimization of PID controllers using ant colony and genetic algorithms* vol. 449: Springer, 2012.

[16] P. Aravind, M. Valluvan, and S. Ranganathan, "Modelling and Simulation of Non Linear Tank," *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering,* vol. 2, pp. 842-849, 2013.

[17] I. B. Manuaba, M. Abdillah, A. Priyadi, and M. H. Purnomo, "COORDINATED TUNING OF PID-BASED PSS AND AVR USING BACTERIAL FORAGING-PSOTVAC-DE ALGORITHM," *Control and Intelligent Systems,* vol. 43, 2015.

[18] A. J. Al Gizi, M. Mustafa, K. M. Al Zaidi, and M. K. Al-Zaidi, "Integrated PLC-fuzzy PID Simulink implemented AVR system," *International Journal of Electrical Power & Energy Systems,* vol. 69, pp. 313-326, 2015.

[19] A. S. 1K. Suresh Manic, V. Rajinikanth, "Refined Double Search Optimization Methodology to Design PID Controller for

Unstable Systems," *Australian Journal of Basic and Applied Sciences,* vol. 8, pp. 44-51, 2014.

[20] V. K. Bhatt and D. S. Bhongade, "Design Of PID Controller In Automatic Voltage Regulator (AVR) System Using PSO Technique," *International Journal of Engineering Research and Applications (IJERA),* vol. 3, pp. 1480-1485, 2013.

[21] H. Xu, C. Caramanis, and S. Mannor, "Sparse algorithms are not stable: A no-free-lunch theorem," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 34, pp. 187-193, 2012.