

Slime Mold Algorithm Based Hybridized Artificial Neural Network Model for Efficient Automatic Voltage Regulation Control

Papri Ghosh¹, Ritam Dutta^{2,*}, V. Muthulakshmi³

¹ CSE Dept., Surendra Institute of Engineering & Management, MAKAUT, West Bengal 734 009, India

² ECE Dept., Surendra Institute of Engineering & Management, MAKAUT, West Bengal 734 009, India

³ IT Dept., St. Joseph's College of Engineering, Anna University, Tamil Nadu 600 119, India

(Received 15 March 2021; revised manuscript received 15 June 2021; published online 25 June 2021)

A novel approach for efficient automatic voltage regulation (AVR) control using hybridized artificial neural network (ANN) model has been proposed in this research work. The novel automatic voltage regulator tuning using an improved neural network has been proposed in this paper. Artificial neural network has been used as focused time delay neural network (FTDNN). Validation is performed by comparing with the methods of feed-forward backpropagation neural networks, cascade-forward backpropagation neural networks, Elman-recurrent neural networks, focused time-delay neural networks and Distributed Time Delay Neural Networks (DTDNN). This hybridized ANN model incorporated a metaheuristic method namely Slime Mold Algorithm (SMA) for obtaining improved result on AVR control. SMA has characteristics that uses adaptive weights to simulate the process to generate feedback from the movement of bio-oscillator-based slime molds in foraging, exploring, and exploiting areas. The performance of the proposed method is focused on speed and rotor angle. The proposed method is compared with other neural network methods in a broad set of benchmarks to verify system efficiency. Promising results were obtained in tuning the AVR under 30 %, 60 % and 90 % loading conditions at slime mold count of 50.

Keywords: Artificial neural network, Slime mold algorithm, Artificial intelligence, Efficient automatic voltage regulator control, Bio-oscillator.

DOI: [10.21272/jnep.13\(3\).03038](https://doi.org/10.21272/jnep.13(3).03038)

PACS number: 07.05.Tp

1. INTRODUCTION

The diversity of electricity grid loading has received more attention in recent years. Electrical system load is a function of the demand for active and reactive power which depends on various variables including time, weather, geography, and economy. This is influenced by the increase in renewable resources and increasing number of electric vehicles that are widespread. The complexity of power systems caused by interconnection and high demand for charging through transmission and distribution networks [1]. Stability has been a major problem in the performance of the operation of the power system. Instability results in island formation, voltage drop, and blackout. This often occurs due to overloads, lightning strikes, faults etc. [2]. Controllers with high stability performance are designed to maintain stable operation in various operating conditions. This is useful to reduce unwanted conditions. Electrical power systems display unfamiliar patterns when experiencing major disruptions. This depends on the loading conditions of the system structure and the location of the disturbance. The modern electric power systems have a high multi-variable character that used to be dynamic with different response rates [3]. Electrical system is designed to operate at a specified value. Changes in the system have been allowed with tolerance limits [4].

A constant output voltage on a generator is very important to produce the expected power supply. Changes in the output voltage of a generator are influenced by various kinds of interference factors, one of which is a

change in load. Therefore, a special regulator equipment is needed to keep the generator output voltage constant even when the generator is affected by these interference factors. In addition, with the aim of maintaining the stability of the system, this regulator must be able to regulate the production or absorption of reactive power from the network at every change in load. This voltage regulator can be controlled both manually and automatically. In large-scale interconnection systems, manual regulators have never been used. Moreover, an automatic regulator device called an Automatic Voltage Regulator (AVR) has been installed in each generator. AVR is set under certain operating conditions. AVR is useful for maintaining the frequency and magnitude of the voltage within its limits. Small changes depend on changes in rotor angle and speed [5].

In several recent research works, the researchers have updated methodologies to improve AVR control more efficiently such as Particle Swarm Optimization (PSO) [6-8], Salp Swarm Optimization algorithm [9, 10], Teaching-Learning-Based Optimization (TLBO) [11, 12], Cuckoo Search Algorithm [13-14], Sine-cosine algorithm [15-16], and neural network [17-19].

An automatic voltage regulator tuning using an improved neural network has been proposed in this paper. The method to improve the neural network is slime mold algorithm. Artificial neural network has been used in this study of focused time delay neural network (FTDNN). The generator is modeled using Heffron-Philips. This work mainly focuses on rotor speed and rotor angle to find out the best suitable performance of the proposed method.

* ritamdutta1986@gmail.com

The results were presented at the International Conference on Innovative Research in Renewable Energy Technologies (IRRET-2021)

Validation is performed by comparing with the methods of feed-forward backpropagation neural networks, cascade-forward backpropagation neural networks, Elman-recurrent neural networks, focused time-delay neural networks and Distributed Time Delay Neural Networks (DTDNN).

2. PROPOSED METHODOLOGY

2.1 Slime Mold Algorithm

Slime Mold Algorithm (SMA) method is an optimization algorithm that duplicates the changing habits and morphology of the slime mold physarum polycephalum in finding food. Slime mold is eukaryotes that occupy cold and damp areas. The main stage of the slime mold namely plasmodium, which is the stage of organic matter in the slime mold finding for food, twining it, and detaching enzymes to eat it. Slime mold has unique patterns that can grow more than 900 square centimeters if there is an excess of food resources in the surrounding environment. Slime mold has the ability to optimize for finding food. Slime mold can choose the most food sources. On the other hand, slime mold also takes into account speed, accuracy and risk in finding food [20].

Slime mold with inadequate information must decide when the right time to move to other areas when looking for food. When slime mold occupies areas of high food supply, the pressure to move decreases. Slime mold has unique characteristics that can utilize a variety of food sources simultaneously. The adaptability of slime molds can adjust search patterns based on food quality. Slime molds will search in restricted areas when food supplies are plentiful. However, the slime mold will enlarge the search area when the food supply in the area is little. The slime mold algorithm generally consists of following parts:

Phase 1: Approach food

Slime mold searches for and recognizes food from odors through the air. This method can be modelled as follows:

$$\overrightarrow{P}(t+1) = \begin{cases} \overrightarrow{P}_b(t) + \overrightarrow{vl} \cdot (\overrightarrow{W} \cdot \overrightarrow{P}_A(t) - \overrightarrow{P}_B(t)), r < p \\ \overrightarrow{vd} \cdot \overrightarrow{P}(t), r \geq p \end{cases} \quad (1)$$

$$p = \tanh|H(i) - BF| \quad i \in 1, 2, \dots, n, \quad (2)$$

$$\overrightarrow{vl} = [-a, a], \quad (3)$$

$$a = \operatorname{arctanh} \left(-\left(\frac{t}{\max_t} \right) + 1 \right), \quad (4)$$

where, the unique spot with the best smell concentration currently established is \overrightarrow{P}_b , \overrightarrow{vl} is measurable variable with a limit $[-a, a]$, linearly decreased variables ranging from 1 to 0 is \overrightarrow{vd} , t is the current iteration, the position of the slime mold is \overrightarrow{P} , two values chosen randomly are \overrightarrow{P}_A and \overrightarrow{P}_B , \overrightarrow{W} is the weight of slime mold. $H(i)$ is the fitness of \overrightarrow{P} , BF is the best fitness obtained in all iterations.

The formula of \overrightarrow{W} is listed as follows:

$$\overrightarrow{W}(\overrightarrow{SmellIndex}(i)) = \begin{cases} 1 + r \cdot \log \left(\frac{bF - H(i)}{bF - wF} + 1 \right), \text{condition} \\ 1 - r \cdot \log \left(\frac{bF - H(i)}{bF - wF} + 1 \right), \text{others} \end{cases} \quad (5)$$

$$\overrightarrow{SmellIndex} = \operatorname{sort}(S), \quad (6)$$

where condition parameter is indicating that $H(i)$ occupies the top of the population, R is a random value between 0 and 1, bF is the best fitness measure in the current iteration process, wF is the lowest fitness value in the currently iteration process, $\overrightarrow{SmellIndex}$ is the series of fitness parameters sorted.

Phase 2: Wrap food

The phase is to duplicate the contraction mode of the slime mold structure mathematically when looking for food. The more food supply that is touched by the slime mold network. The signal generated by the bio-oscillator is getting stronger, the faster the cytoplasm flows and the thicker the net.

$$S\overrightarrow{P}^* = \begin{cases} \operatorname{rand} \cdot (ub - lb) + lb, \operatorname{rand} < z \\ \frac{\operatorname{rand} \cdot (ub - lb) + lb}{\overrightarrow{P}_b(t) + \overrightarrow{vl} \cdot (\overrightarrow{W} \cdot \overrightarrow{P}_A(t) - \overrightarrow{P}_B(t))}, r < p \\ \overrightarrow{vd} \cdot \overrightarrow{P}(t), r \geq p \end{cases} \quad (7)$$

where lb and ub are the lower and upper limits of the seeking, rand and r are the random values with range 0 to 1, z is an experiment to parameter settings.

Phase 3: Grabble food

Slime mold is very dependent on the information signal sent by the biological oscillator. This will make the slime mold in the best position. For the duplication of variant widths of the slime mold parameters are used \overrightarrow{W} , \overrightarrow{vl} , and \overrightarrow{vd} .

\overrightarrow{W} represents a duplication of the slime mold oscillation frequency when at different food concentrations. This makes slime mold reach food faster. On the other hand, when the concentration of food is low from the position of the slime mold. Slime mold will slow down. So that the movement of the slime mold is more efficient. The selective character of the slime mold is duplicated in \overrightarrow{vl} and \overrightarrow{vd} to find the latest food sources. If the slime mold finds the latest food source. Slime mold still have organics that can be used to explore the latest and highest quality food sources.

\overrightarrow{vl} process that duplicates the behavior of the mold slime to provide information whether to approach the food or find other food sources. The period will be influenced by environmental factors. On the other hand, this will increase the impulse of the slime mold to find high-quality food and avoid local traps.

2.2 Focused Time Delay Neural Networks (FTDNN)

Focused Time Delay Neural Networks (FTDNN) are dynamic ANN types. FTDNN has a topology consisting

of forward feed which has a time delay. This is placed on the network through which the input is passed. FTDNN was developed primarily to process temporary patterns [19]. Time delay placed on topology will support in predicting and controlling problems effectively and efficiently in dynamic ANN, the results have several variables that can be used to maintain the contextual part of the input unit. The part is available in local memory in the form of tapped time delay. The basic FTDNN topology consists of memory structures and non-linear associations [21]. The FTDNN architecture can be seen in Fig. 1.

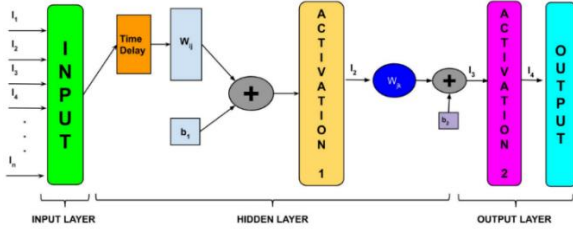


Fig. 1 – Focused time delay neural network: different layer topologies

The memory structure functions are to store records of related past information and the associator functions is to predict future events with the help of memory structures. Memory structure contains a time delay signal that originates from the input. Whereas the associator contains a conventional network of feed-forward. The main structure of FTDNN is to have a special memory structure that is in the input layer. The advantage of FTDNN is having a simple topology. Where L is the length of the delay path memory, unit input is $I_n(t)$, the network processes is $I_n(t)$, $I_n(t-1)$, $I_n(t-2)$... and $I_n(t-L)$. Therefore, the input signal $X_1(t)$ to neuron i (Fig. 1) is given as:

$$X_1(t) = \sum_{i=1}^j W_{ij} I_n(t-L) + b_1. \quad (8)$$

Output $X_2(t)$ is the result of processing $X_1(t)$ using a non-linear activation function. A function that is often used is the sigmoid activation function

$$X_2(t) = f(X_1(t)) = \frac{1}{1 + \exp^{-X_1}}. \quad (9)$$

At layer 2 there is no time delay. Output $X_2(t)$ is becomes input at layer 2. It becomes output $X_3(t)$. The results are then activated using the activation function

$$X_3(t) = \sum_{j=1}^k W_{jk} X_2(t) + b_2, \quad (10)$$

$$X_4(t) = f(X_3(t)) = \frac{1}{1 + \exp^{-X_3}}. \quad (11)$$

One of FTDNN's features is a topology that has not backpropagation to calculate network gradients. The tapped time delay is only in the input layer and does not contain a feedback loop.

2.3 Automatic Voltage Regulator

The generator control consists of two parts, namely

Automatic voltage regulators (AVR) and power system stabilizers (PSS). AVR and PSS have functions to maintain generator stability [22]. AVR functions to regulate the terminal voltage at a predetermined value. Simple AVR is composed of four components, namely amplifier, exciter, generator and sensor. The schematic of AVR can be seen in Fig. 2. For mathematical modeling, components are assumed to be linear. This takes into account the main time constant and ignores non-linearity. The control of the generator is composed of the AVR and PSS. An automatic voltage regulator (AVR) and power system stabilizer (PSS) are used to repair the transient stability.

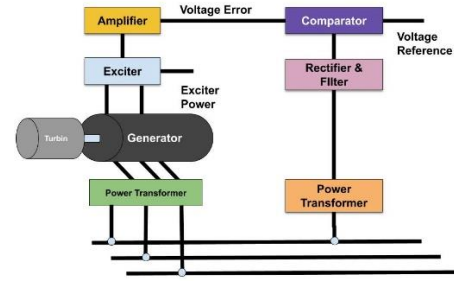


Fig. 2 – Automatic voltage regulation control schematic diagram

3. PROPOSED SMA-FTDNN MODEL

The application of SMA and FTDNN in the tuning of AVR can be illustrated as shown in Fig. 3. The first step is to model the generator in a single machine type Heffron-Philips. The simulation results of the modeling are data speed and rotor angle. This data is used as input in a neural network. Random initial weight data from FTDNN is sampled. The results are processed using SMA which has three phases. It is a food approach, wrapping food and garbled food. SMA results will be a potential weight for FTDNN that can be used in the network.

4. RESULTS AND DISCUSSION

The proposed method was applied to optimize conventional controller in order to see the enhancement stability of the power system. The application for the SMA-FTDNN was written in MATLAB. The dynamic and transient condition of the proposed regulator was compared with the operation of the other method controller under three different loading conditions: 30, 60 and 90 %. The validation of the proposed method will be carried out by comparing it with the FFBNN, CFBNN, Elman-RNN, FTDNN and DTDNN methods. Each of these was investigated for speed and rotor angle.

The variables of SMA are also required to be set up before optimization of the FTDNN model, including the number of Slime Mold (Search Agent), maximum number of iterations, and lower-upper limit of the optimization (L). Fig. 4 is the performance of search agent with 100 iterations. The use of the highest search agent gets the best score for fitness. The peak value is 0.2093. Meanwhile, the settling time value is 15.581. The detailed results can be seen in Table 1.

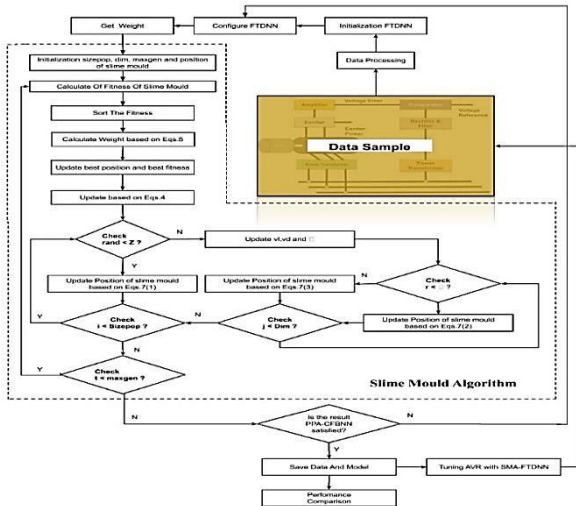


Fig. 3 – Proposed SMA-FTDNN model flowchart

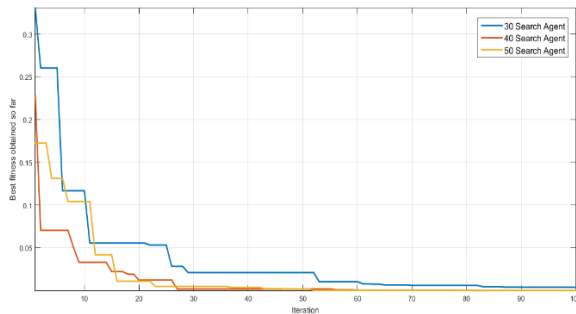


Fig. 4 – Convergence graph of various search agents used for SMA-FTDNN process

Table 1 – Parameter values for various search agents

| Search agent | Rise time | Settling time | Peak |
|--------------|-----------|---------------|--------|
| 30 | 24.32 | 52.298 | 0.3406 |
| 40 | 13.78 | 26.723 | 0.2380 |
| 50 | 07.09 | 15.581 | 0.2093 |

The lower-upper parameter also needs to be set to get a value that can be used for optimization. The test uses a random range value, namely the range $[-1, 1]$, $[-5, 5]$, $[-10, 10]$ and $[-20, 20]$. Fig. 5 shows the convergence curve value for each lower-upper using the SMA-FTDNN method. In Fig. 5, it can be seen that the lower-upper value quickly converges when using the value $[-1, 1]$. Testing of slime mold and lower-upper parameters uses 100 iterations.

Measurement of the performance of the generator is the speed and rotor angle. The first test is carried out by overloading the generator with a loading of 30%. The results of the loading can be seen in Fig. 6 and Fig. 7. From Fig. 6, the undershoot values of the FFBNN, CFBNN, FTDNN and SMA-FTDNN methods are obtained the same value. The value is -0.53 : the worst score for undershoot belongs to the E-RNN method. Value -0.53 ; meanwhile, the overshoot value of the CFBNN, FTDNN and DTDNN methods has the same value, namely 0.39 . the worst score belongs to the E-RNN method. The value is 0.42 .

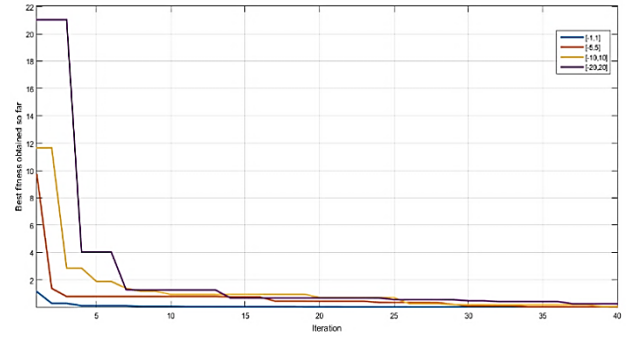


Fig. 5 – Convergence graph of lower-upper

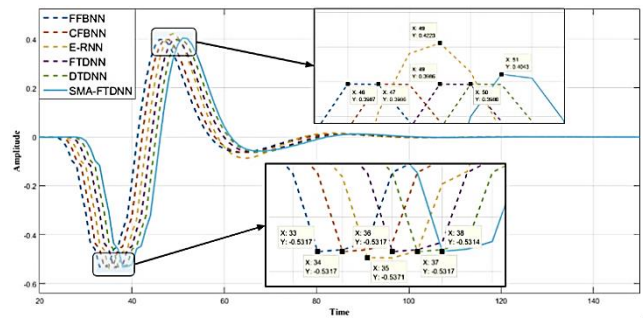


Fig. 6 – AVR speed response with 30 % load

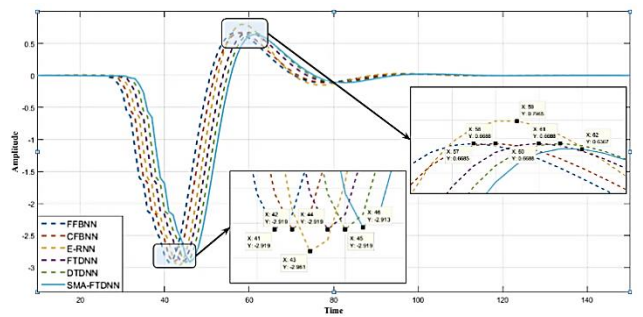


Fig. 7 – AVR rotor angle response with 30 % load

Fig. 7 is the result of the rotor angle with a loading of 30%. In Fig. 7, the worst values for the overshoot and undershoot are 0.7965 and -2.961 . This value belongs to the E-RNN method. Meanwhile, the undershoot value of the FFBNN, CFBNN, FTDNN and DTDNN methods obtained the same value that is equal to -2.919 .

In Table 2, it can be seen that the undershoot, overshoot and settling time of the rotor angle in the SMA-FTDNN method is the best. The values are -2.913 , 0.6367 and 86.944 . The undershoot value of speed in the SMA-FTDNN method is the same as for the FFBNN, CFBNN and FTDNN methods. Meanwhile, the overshoot value is slightly better than the E-RNN method. The value is 0.404 .

Experiments with a load of 60%, the results of the SMA-FTDNN method obtained the best overshoot and undershoot values of the rotor angle. The values are 0.7050 and -3.1524 . Meanwhile, the speed results get the best score in the undershoot section. The value is -0.5687 . The details of the results of the 60% loading can be seen in Table 3.

Table 2 – Speed and rotor angle response at 30 % of load

| Methods | Speed response | | | | Rotor angle response | | | |
|-----------|----------------|------------|-----------------------|-------------------|----------------------|------------|----------------------|-------------------|
| | Under-shoot | Over-shoot | Rise time (s) | Settling time (s) | Under-shoot | Over-shoot | Rise time (s) | Settling time (s) |
| FFBNN | -0.5317 | 0.3987 | 0.453 | 86.783 | -2.919 | 0.6685 | $2.82 \cdot 10^{-7}$ | 84.427 |
| CFBNN | -0.5317 | 0.3986 | 0.5475 | 86.849 | -2.919 | 0.6688 | $2.91 \cdot 10^{-7}$ | 84.453 |
| E-RNN | -0.5371 | 0.4223 | 0.6947 | 86.744 | -2.9861 | 0.7965 | $2.91 \cdot 10^{-7}$ | 84.453 |
| FTDNN | -0.5317 | 0.3986 | 0.444 | 86.844 | -2.919 | 0.6688 | $2.90 \cdot 10^{-7}$ | 84.451 |
| DTDNN | -0.5317 | 0.3986 | 0.444 | 86.844 | -2.919 | 0.6688 | $2.90 \cdot 10^{-7}$ | 84.45 |
| SMA-FTDNN | -0.5314 | 0.4043 | $1.23 \cdot 10^{-10}$ | 86.944 | -2.913 | 0.6367 | 0.7659 | 84.302 |

Table 3 – Speed and rotor angle response at 60 % of load

| Methods | Speed response | | | | Rotor angle response | | | |
|-----------|----------------|------------|-----------------------|-------------------|----------------------|------------|----------------------|-------------------|
| | Under-shoot | Over-shoot | Rise time (s) | Settling time (s) | Under-shoot | Over-shoot | Rise time (s) | Settling time (s) |
| FFBNN | -0.5711 | 0.4296 | 0.4819 | 88.144 | -3.1694 | 0.7161 | $2.61 \cdot 10^{-7}$ | 85.671 |
| CFBNN | -0.571 | 0.4294 | 0.5937 | 88.17 | -3.1692 | 0.7165 | $2.77 \cdot 10^{-7}$ | 85.692 |
| E-RNN | -0.5795 | 0.4555 | 0.6714 | 88.001 | -3.2385 | 0.8592 | $2.48 \cdot 10^{-6}$ | 83.886 |
| FTDNN | -0.571 | 0.4294 | 0.3998 | 88.172 | -3.1692 | 0.7164 | $2.76 \cdot 10^{-7}$ | 85.692 |
| DTDNN | -0.571 | 0.4294 | 0.6879 | 88.176 | -3.1693 | 0.7164 | $2.77 \cdot 10^{-7}$ | 85.694 |
| SMA-FTDNN | -0.5687 | 0.4338 | $6.76 \cdot 10^{-10}$ | 91.243 | -3.1524 | 0.705 | 0.1695 | 87.361 |

Table 4 – Speed and rotor angle response at 90 % of load

| Methods | Speed response | | | | Rotor angle response | | | |
|-----------|----------------|------------|---------------|-------------------|----------------------|------------|----------------------|-------------------|
| | Under-shoot | Over-shoot | Rise time (s) | Settling time (s) | Under-shoot | Over-shoot | Rise time (s) | Settling time (s) |
| FFBNN | -0.6319 | 0.4793 | 0.6236 | 88.557 | -3.4696 | 0.7743 | $3.32 \cdot 10^{-7}$ | 86.004 |
| CFBNN | -0.6319 | 0.4792 | 0.6678 | 88.651 | -3.4696 | 0.7748 | $2.89 \cdot 10^{-7}$ | 86.009 |
| E-RNN | -0.6425 | 0.5084 | 0.6283 | 88.408 | -3.5531 | 0.9341 | $3.05 \cdot 10^{-7}$ | 84.194 |
| FTDNN | -0.6319 | 0.4792 | 0.3231 | 88.66 | -3.4696 | 0.7747 | $2.91 \cdot 10^{-7}$ | 86.012 |
| DTDNN | -0.6319 | 0.4792 | 0.6527 | 88.659 | -3.4696 | 0.7747 | $2.89 \cdot 10^{-7}$ | 86.01 |
| SMA-FTDNN | -0.6164 | 0.4743 | 0.8365 | 96.784 | -3.4159 | 0.7821 | $2.76 \cdot 10^{-8}$ | 91.093 |

Detailed results of 90 % loading have been portrayed in Table 4. With a system loading of 90 %, the output value of overshoot and undershoot of the speed respond for the SMA-FTDNN method provides highest value. The values are -0.6164 and 0.4743. On the other hand, the results of overshoot and undershoot of the rotor angle are different. The undershoot value of the SMA-FTDNN method is the best.

5. CONCLUSIONS

The development of the latest hybrid methods based on artificial intelligence techniques for controlling AVR has been discussed in this paper. The comparison and evaluation of the methods developed with other methods are carried out in this article. It can be concluded that the metaheuristic method has a better performance in optimizing artificial neural network. By using the new method, the slime mold algorithm to optimize HANN with the focused time delay, the best value is obtained in automatic voltage regulation control.

REFERENCES

- J.G. Vlachogiannis, *IEEE Trans. Power Syst.* **24** No 4, 1808 (2009).
- X. Zhao, X. Zhang, B. He, *Energy Convers. Manag.* **50** No 3, 658 (2009).
- S. Kumar, Nallagalva, M.K. Kirar, Ganga Agnihotri, *International Journal of Scientific Engineering and Technology* 1 No 3 (2012).
- S. Majumdar, K. Mandal, N. Chakraborty, *2014 Annual IEEE India Conference (INDICON)*, 1 (Pune: India: 2014).
- H. Saadat, *Power System Analysis* (New York: McGraw-Hill: 1999).
- H. Gozde, M.C. Taplamacioğlu, M. Ari, *Proceedings of the 2014 6th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, 23 (Bucharest: Romania: 2014).
- J. Femmy Nirmal, D. Jeraldin Auxillia, *International Conference on Circuits, Power and Computing Technologies (ICCPCT)*, 661 (Nagercoil, India: 2013).
- R. Indhuja, N. Kamaraj, *International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, 159 (Kumaracoil: India: 2016).
- IA. Khan, A.S. Alghamdi, T.A. Jumani, A. Alamgir, A.B. Awan, Khidrani, *Electronics* **8**, 1472 (2019).
- P. Sirsode, A. Tare, V. Pande, *Sixth Indian Control Conference (ICC)*, 431 (Hyderabad: India: 2019).
- A. Mishra, N. Singh, S. Yadav, *Advances in Computing and Intelligent Systems*, 153 (Singapore: Springer: 2020).
- B. Ataşlar Ayyıldız, O. Karahan, *J. Sci. Technol. A* **21** No 1, 128 (2020).

13. A. Sikander, P. Thakur, R.C. Bansal, S. Rajasekar, *Comput. Elect. Eng.* **70**, 261 (2018).
14. B. Zafer, O. Karahan, *J Frankl. Inst.* **355** No 13, 5534 (2018).
15. J. Bhookya, R.K. Jatoth, *Evolutionary Intelligence* **12** No 4, 725 (2019).
16. B. Hekimoğlu, *T.I. Meas. Control.* **41** No 6, 1761 (2019).
17. G. Bal, O. Kaplan, S.S. Yalcin, *8th International Conference on Renewable Energy Research and Applications (ICRERA)*, 1032 (Brasov: Romania: 2019).
18. M. Elsis, *Neural Comput. Appl.* **31**, 5017 (2019).
19. W. Aribowo, *EMITTER International Journal of Engineering Technology* **7** No 1, 34 (2019).
20. S. Li et al., *Future Gener. Comput. Syst.* (2020).
21. M. Abed, A. El-Shafie, S. Siti, *J. Comput. Sci.* **6**, 597 (2010).
22. S. Essallah, A. Bouallegue, A. Khedher, *J. Mod. Power System. Clean Energy* **7** No 5, 1115 (2019).

Гібридизована модель штучної нейронної мережі на основі алгоритму слизової цвілі для ефективного автоматичного регулювання напруги

Papri Ghosh¹, Ritam Dutta², V. Muthulakshmi³

¹ CSE Dept., Surendra Institute of Engineering & Management, MAKAUT, West Bengal 734 009, India

² ECE Dept., Surendra Institute of Engineering & Management, MAKAUT, West Bengal 734 009, India

³ IT Dept., St. Joseph's College of Engineering, Anna University, Tamil Nadu 600 119, India

У роботі запропоновано новий підхід для ефективного автоматичного регулювання напруги (AVR) за допомогою гібридизованої моделі штучної нейронної мережі (ANN), а саме запропоновано нове налаштування AVR з використанням вдосконаленої нейронної мережі. ANN була використана як нейронна мережа зі сфокусованою затримкою часу (FTDNN). Перевірка проводилася шляхом порівняння з методами нейронних мереж зворотного поширення з прямим зв'язком, нейронних мереж зворотного поширення з каскадним зв'язком, рекурентних нейронних мереж Елмана, FTDNN і нейронних мереж з розподіленою затримкою часу (DTDNN). Ця гібридизована модель ANN включала метаевристичний метод, а саме алгоритм слизової цвілі (SMA) для отримання поліпшених результатів по контролю AVR. SMA має характеристики, які використовують адаптивні одиниці маси для моделювання процесу створення зворотного зв'язку від переміщення слизової цвілі на основі біо-осцилятора в місцях видобутку, дослідження і експлуатації. Ефективність запропонованого методу орієнтована на швидкість та кут ротора. Запропонований метод порівнюється з іншими методами нейронних мереж у широкому наборі тестів для перевірки ефективності системи. Були отримані перспективні результати при налаштуванні AVR для навантаження 30, 60 та 90 % і кількості слизової цвілі 50.

Ключові слова: Штучна нейронна мережа, Алгоритм слизової цвілі, Штучний інтелект, Ефективне автоматичне регулювання напруги, Біогенератор.