Adaptive Activation Functions for Deep Learning-based Power Flow Analysis

Kaseb, Zeynab; Xiang, Yu; Palensky, Peter; Vergara, Pedro P.

DOI
10.1109/ISGTEUROPE56780.2023.10407913

Publication date
2024

Document Version
Final published version

Published in
Proceedings of 2023 IEEE PES Innovative Smart Grid Technologies Europe, ISGT EUROPE 2023

2023 IEEE PES Innovative Smart Grid Technologies Europe (ISGT EUROPE) | 979-8-3503-9678-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/ISGTEUROPE56780.2023.10407913

Adaptive Activation Functions for Deep Learning-based Power Flow Analysis

Zeynab Kaseb[1], Yu Xiang[2], Peter Palensky[1], Pedro P. Vergara[1]
[1]Intelligent Electrical Power Grids, Delft University of Technology, The Netherlands
[2]Alliander N.V., The Netherlands
Emails: Z.Kaseb@tudelft.nl, Tony.Xiang@alliander.com,
P.Palensky@tudelft.nl, P.P.VergaraBarrios@tudelft.nl

Abstract—**This paper investigates the impact of adaptive activation functions on deep learning-based power flow analysis. Specifically, it compares four adaptive activation functions with state-of-the-art activation functions, i.e., ReLU, LeakyReLU, Sigmoid, and Tanh, in terms of loss function error, convergence speed, and learning process stability, using a real-world distribution network dataset. Results indicate that the proposed adaptive activation functions improve learning capability over state-of-the-art activation functions. Notably, adaptive ReLU activation shows the most improved learning process, with convergence speed up to twice as fast as ReLU. Adaptive Sigmoid and Tanh activation functions exhibit superior performance in terms of loss function error, outperforming ReLU and LeakyReLU by up to two orders of magnitude. Furthermore, the proposed adaptive activation functions lead to smoother and more stable learning processes, especially during early training, improving convergence. The practical implications of this study include the potential application of these adaptive activation functions in distribution network modeling, forecasting, and control, leading to more accurate and reliable power system operation.**

Index Terms—**Machine learning, model-based neural networks, energy systems, power flow, distribution networks.**

I. Introduction

Ensuring the safe and efficient operation of distribution networks requires the fundamental task of power flow analysis. Traditional techniques for power flow analysis rely on iterative numerical algorithms, which can be computationally expensive for large-scale distribution networks, or inaccurate under certain circumstances (e.g., [1], [2]). In recent years, deep learning has shown great potential in this field, with successful applications to power flow analysis and modeling (e.g., [3]), optimal power flow (e.g., [4]), and unit commitment (e.g., [5]). By training deep neural networks on large datasets, it becomes possible to learn highly complex nonlinear relationships between input and output variables. As a result, accurate and fast solutions for power flow formulations can be obtained [6].

The trainable parameters of deep neural networks depend heavily on the derivative of the loss function. Likewise, the derivative of the loss function depends on the derivative of the activation function. Thus, the activation functions play a crucial role in introducing non-linearity to the neural network model, which ultimately increases their generalization capabilities [7]. The speed of convergence, avoidance of local minima, generalization to new data, and ability to capture complex relationships between input and output variables, among others, are all affected by the type of activation function. Therefore, choosing the right activation function is crucial in achieving high accuracy and efficiency in deep learning applications, such as distribution networks analysis and modeling [8].

While traditional activation functions such as ReLU, LeakyReLU, Sigmoid, and Tanh have been widely used in deep learning models, they have limitations that can negatively affect model performance. For example, ReLU and LeakyReLU suffer from the ”dying ReLU” problem, where a large portion of neurons may become inactive and output zero during training. This can slow down or even prevent convergence and decrease the model's expressive power [9]. Sigmoid and Tanh can suffer from vanishing gradient problems, which can make it difficult to train deep neural networks [10]. Additionally, these traditional activation functions have fixed functional forms that cannot adapt to the data distribution or the training process [7].

To address these limitations, adaptive activation functions have been proposed to improve the learning process. These functions can dynamically adjust their shape based on the input data distribution, which allows them to better capture complex relationships between input and output variables. By adaptively modifying their functional form, these activation functions can mitigate the vanishing and exploding gradient problems and improve the model's stability and convergence. As a result, exploring adaptive activation functions has become an active area of research in deep learning, with promising results in various applications [11]–[13].

To the best of the authors' knowledge, the impact of adaptive activation functions on deep learning-based power flow analysis has not yet been systematically investigated. The significance of conducting a systematic investigation lies in the potential to provide valuable insights into the effectiveness of adaptive activation functions in enhancing the accuracy and efficiency of deep learning-based power flow analysis, which, in turn, contributes to the safe and stable operation of distribution networks. In this paper, therefore, the focus is on the impact of four adaptive activation functions on the loss function error, convergence speed, and learning process stability of deep learning-based power flow analysis. The proposed activation functions are then systematically compared with the state-of-the-art activation functions that are conventionally used for power flow applications, i.e., ReLU, LeakyReLU, Sigmoid, and Tanh.

II. Deep Learning-based Power Flow Analysis

A. Power Flow Formulation

The power flow analysis is a fundamental task in distribution network modeling, which aims to calculate the voltage magnitude and phase angle at each bus of the network. In this study, the power flow analysis is performed based on the AC power flow equations, which are a set of nonlinear equations that relate the complex voltage, current, and power at each bus of the network. The AC power flow equations can be formulated as follows:

$$p_i = \sum_{j=1}^{n} (v_i v_j (g_{ij} \cos\delta_{ij} + b_{ij} \sin\delta_{ij})) \tag{1}$$

$$q_i = \sum_{j=1}^{n} (v_i v_j (g_{ij} \sin\delta_{ij} - b_{ij} \cos\delta_{ij})) \tag{2}$$

where i and j are the indices of the buses, n is the total number of buses in the network, v_i and δ_i are the magnitude and phase angle of the complex voltage at bus i, p_i and q_i are the active and reactive power injection at bus i, g_{ij} and b_{ij} are the real and imaginary parts of the admittance between buses i and j, and $\delta_{ij} = \delta_i - \delta_j$ is the phase angle difference between the voltages at buses i and j.

The power flow equations represent the physical laws that govern the flow of power in a network and are typically solved iteratively until a convergence criterion is met. In this study, the Newton-Raphson method is used to solve the power flow equations. This method is widely used in power system analysis due to its efficiency and robustness in handling both radial and meshed networks (e.g., [14], [15]).

B. Neural Network Architecture

The neural network architecture is designed to accurately capture the nonlinear relationship between the input and output variables while preventing overfitting and ensuring the stability of the learning process. Two fully-connected neural networks, $NN_{|v|}$ and NN_{δ}, are developed to approximate the voltage magnitude and voltage angle of all the buses involved in a distribution network, respectively, based on the active and reactive power injections at all the buses. Accordingly, the number of neurons in the input layer is twice the number of buses involved, while the number of neurons in the output layer is equal to the number of buses involved. The neural networks representation is given by equation (3):

$$y_i = \sigma(\sum(w_i x_i) + b) \tag{3}$$

where x_i and y_i are the i^{th} input and output vectors, respectively, w and b are the network weights and bias, respectively, and σ is the nonlinear activation function, which is applied to the output of each (input/hidden) layer before the next (hidden/output) layer.

1) Loss function: The mean squared error (MSE) is used as the evaluation metric, i.e., loss function, to assess the performance of the trained neural networks. The goal of training the deep neural networks is to find optimal weights and biases that minimize the loss function. The loss function is defined as a supervised loss term, which is represented by equation (4):

$$MSE = \frac{1}{N} \sum_{i=1}^{n} (y_i - f(p_i, q_i))^2 \tag{4}$$

where y_i represents the ground-truth data obtained from the Newton-Raphson method, which corresponds to the ith voltage magnitude or voltage angle for $NN_{|v|}$ and NN_{δ}, respectively. N is the total number of samples, p_i and q_i denote the active and reactive power at node i, and $f(p_i, q_i)$ is the output obtained from the neural network. The loss function is minimized by adjusting the weights and biases during the training process, which in turn improves the accuracy of the power flow analysis.

2) Activation function: Activation functions are used to introduce nonlinearity into the neural networks. The following commonly-used activation functions are used:

- **Rectified Linear Unit (ReLU)**: $f(x) = \max(0, x)$
- **Leaky Rectified Linear Unit (LeakyReLU)**: $f(x) = \max(0, x) + v \min(0, x)$
- **Sigmoid**: $f(x) = \frac{1}{1+\exp(-x)}$
- **Hyperbolic Tangent (Tanh)**: $f(x) = \frac{\exp(x)-\exp(-x)}{\exp(x)+\exp(-x)}$

In this study, adaptive activation functions are employed to perform model-based nonlinear transformations for the two developed neural networks. Two trainable parameters, α and β, are introduced to implement the nonlinearity by scaling and shifting the output results of the activation function based on the input features. α is a positive trainable parameter that scales the input (e.g., [7]), and β is a trainable parameter that adds to it. Figures 1 and 2 show the impact of α and β on the activation functions. The formulations of the adaptive activation functions are given by:

- **Adaptive ReLU**: $f(x) = \max(0, \alpha x + \beta)$
- **Adaptive LeakyReLU**: $f(x) = \max(0, \alpha x + \beta) + v \min(0, \alpha x + \beta)$
- **Adaptive Sigmoid**: $f(x) = \frac{1}{1+\exp(-(\alpha x+\beta))}$
- **Adaptive Tanh**: $f(x) = \frac{\exp(\alpha x+\beta)-\exp(-(\alpha x+\beta))}{\exp(\alpha x+\beta)+\exp(-(\alpha x+\beta))}$

During each iteration of the optimization process, the gradients of the loss function with respect to the trainable parameters, $\nabla_{w,b,\alpha,\beta} L$, are computed using the chain rule of differentiation. The resulting gradients are then backpropagated through the network to update and optimize the weights w, biases b, scaling parameter α, and shifting parameter β. This process enables the network to learn the optimal values of α and β, in addition to w and b, leading to improved performance in approximating the output variable. More detailed information about adaptive activation functions can be found in [16].

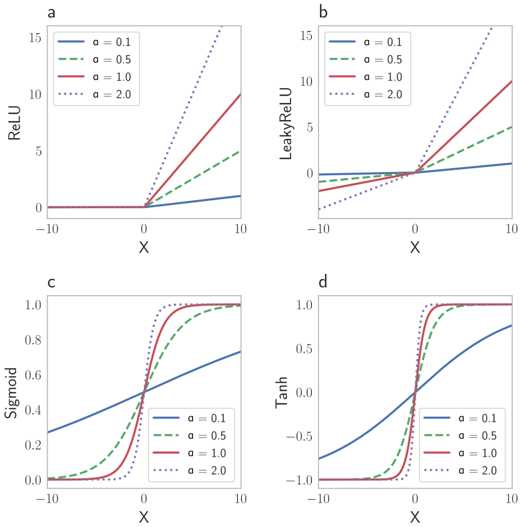

Fig. 1. Impact of the scaler α on the activation functions: (a) ReLU, (b) LeakyReLU, (c) Sigmoid, and (d) tanh.

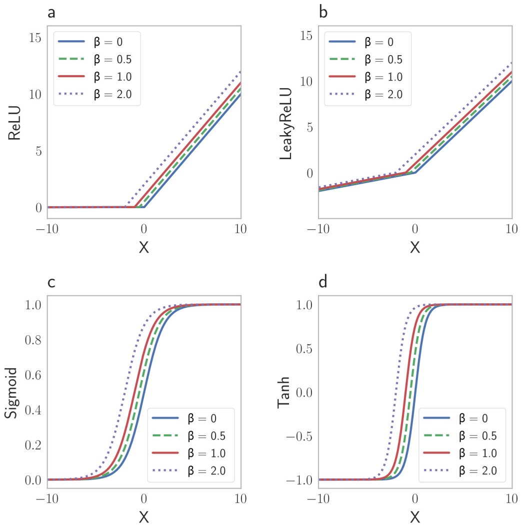

Fig. 2. Impact of β on the activation functions: (a) ReLU, (b) LeakyReLU, (c) Sigmoid, and (d) tanh.

III. Results

A. Model Setup

The neural networks are trained using the Adam optimization algorithm with a learning rate of 1×10^{-4} and a batch size of 128. A real-world medium-voltage distribution network comprising 369 buses is used as the basis for the dataset, which includes a total of 10,000 data points randomly sampled from a given time series in 2021. The input features of the dataset are the active and reactive power injection/consumption of all the buses, while the output features consist of the voltage magnitude and voltage angle of all the buses. The high-resolution Newton-Raphson numerical method is employed to generate the ground-truth data using the Power Grid Model package [17]. The dataset is divided into two subsets, where 60% of the data is used for training, and the remaining 40% is used for testing.

The number of neurons in the hidden layers is chosen to be 512, 256, 256, and 128, respectively, resulting in a total of four hidden layers and 655,985 trainable parameters. Dropout regularization is also implemented after each hidden layer with a probability of 20% to prevent overfitting. The training process is stopped after 1000 epochs. Note that the architecture of the neural networks is achieved through a sensitivity analysis conducted on a subset of the dataset, which aims to find the optimal number of hidden layers, neurons per hidden layer, dropout percentage, learning rate, and batch size.

B. Model Performance

This section evaluates the impact of using the proposed adaptive activation functions on the deep learning model performance by comparing their impact on the learning process with that of four commonly-used activation functions, namely ReLU, LeakyReLU, Sigmoid, and Tanh. The evaluation is based on three criteria: (i) the loss function error, measured by the mean squared error after the first 400 epochs, (ii) the convergence speed, and (iii) the stability of the learning process. The results are presented in the following subsections, providing insights into the effectiveness of the adaptive activation functions in improving the performance and efficiency of deep learning models for power flow analysis.

1) Loss function error: The impact of the adaptive activation functions is evaluated based on the model's ability to reduce the loss function error during the training process. The loss function error is measured after 400 epochs for all the commonly-used and the adaptive activation functions investigated in this study. The results presented in Table I show that the proposed adaptive activation functions outperform the corresponding commonly-used activation functions in terms of loss function error for both voltage magnitude and voltage angle. Particularly, the adaptive Sigmoid and Tanh activation functions show the lowest loss function error after 400 epochs. It should be noted that Tanh and Sigmoid are prone to vanishing gradients, which may affect the performance of the model in deeper neural networks.

Figure 3 compares the actual voltage magnitude of all the buses for an extreme point in the testing dataset with those predicted by ReLU and adaptive ReLU after 400 epochs. Using an adaptive version of ReLU, the neural network $NN_{|v|}$ can provide better approximations of voltage magnitude at extreme points, as evidenced by a lower maximum error of

TABLE I

Loss function error after 400 epochs.

Activation Function	MSE $[pu^2]$	MSE $[rad^2]$
ReLU	5.645×10^{-5}	6.081×10^{-5}
Adaptive ReLU	1.2×10^{-6}	2.443×10^{-6}
LeakyReLU	1.408×10^{-3}	2.923×10^{-3}
Adaptive LeakyReLU	3.262×10^{-4}	1.882×10^{-4}
Sigmoid	9.538×10^{-5}	9.842×10^{-6}
Adaptive Sigmoid	7.164×10^{-6}	1.016×10^{-6}
Tanh	3.047×10^{-6}	3.262×10^{-5}
Adaptive Tanh	2.33×10^{-7}	2.59×10^{-6}

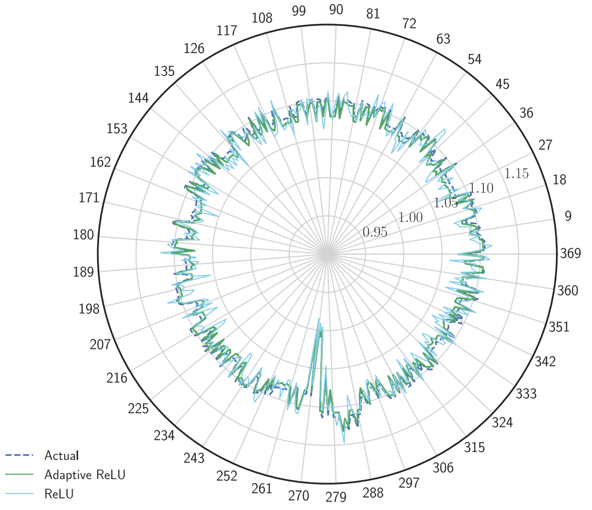

Fig. 3. Comparison of ReLU and adaptive ReLU performance in approximating voltage magnitudes at all buses for an extreme point in the testing dataset.

0.012 pu compared to the error of 0.031 pu observed with ReLU after 400 epochs. This is of great importance in the operation of distribution networks where unexpected events can cause voltage magnitude to deviate significantly from nominal values, and accurate predictions are necessary for safe and stable operation.

2) Convergence speed: The impact of the adaptive activation functions on the convergence speed is evaluated based on the point at which the loss function reached a plateau, defined as a tolerance of 10^{-3}, indicating negligible changes in the loss. Figures 4 and 5 compare the convergence speeds of the adaptive activation functions with the corresponding commonly-used activation functions for $NN_{|v|}$ and NN_{δ}, respectively. The findings demonstrate that the adaptive activation functions achieved faster convergence compared to their commonly-used counterparts. Specifically, in the case of voltage magnitude, the ReLU function did not converge even after 1000 epochs, whereas the adaptive ReLU function converged after 600 epochs, as shown in Figure 4a. Similar results were obtained for the voltage angle. Note that the y-axis of the diagrams is scaled to better visualize the changes.

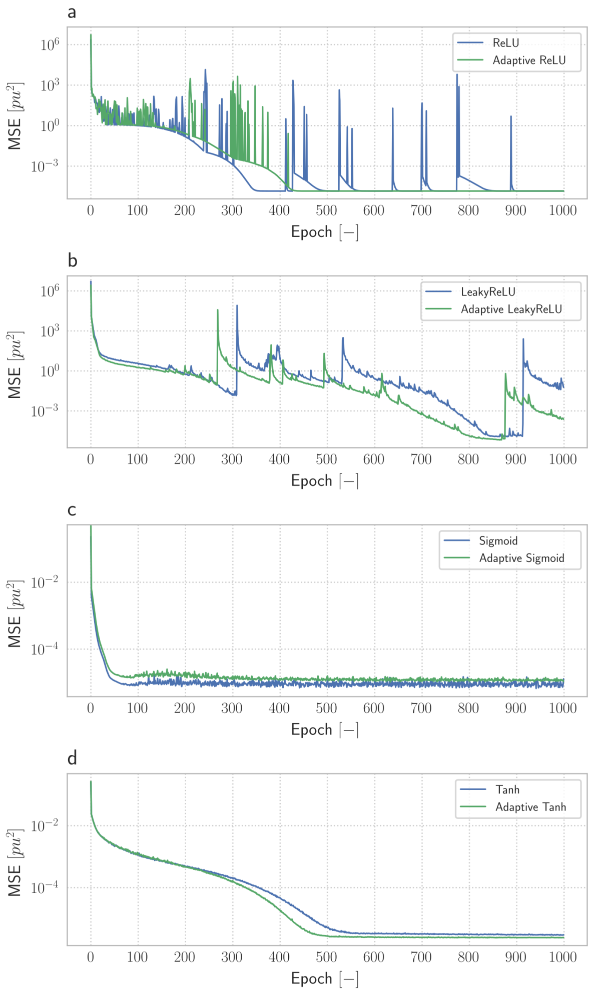

Fig. 4. Comparison of the convergence speed of the neural network to approximate voltage magnitude for: (a) ReLU and Adaptive ReLU, (b) LeakyReLU and Adaptive LeakyReLU, (c) Sigmoid and Adaptive Sigmoid, and (d) Tanh and Adaptive Tanh.

3) Learning process stability: The stability of the learning process is analyzed for both the commonly-used activation functions and the adaptive activation functions. A closer look at Figures 4 and 5 reveals that the adaptive activation functions exhibit greater stability than their commonly used counterparts. Specifically, the adaptive activation functions were less prone to oscillations during the learning process, which led to a smoother convergence. This improved stability of the adaptive activation functions is an important advantage, as it can help prevent the model from getting stuck in local minima and contribute to the model's overall accuracy.

IV. Conclusion

The present study evaluates the impact of using adaptive activation functions on deep learning-based power flow analysis of distribution networks. The performance of the deep learning models is evaluated based on the loss function error, convergence speed, and stability of the learning process,

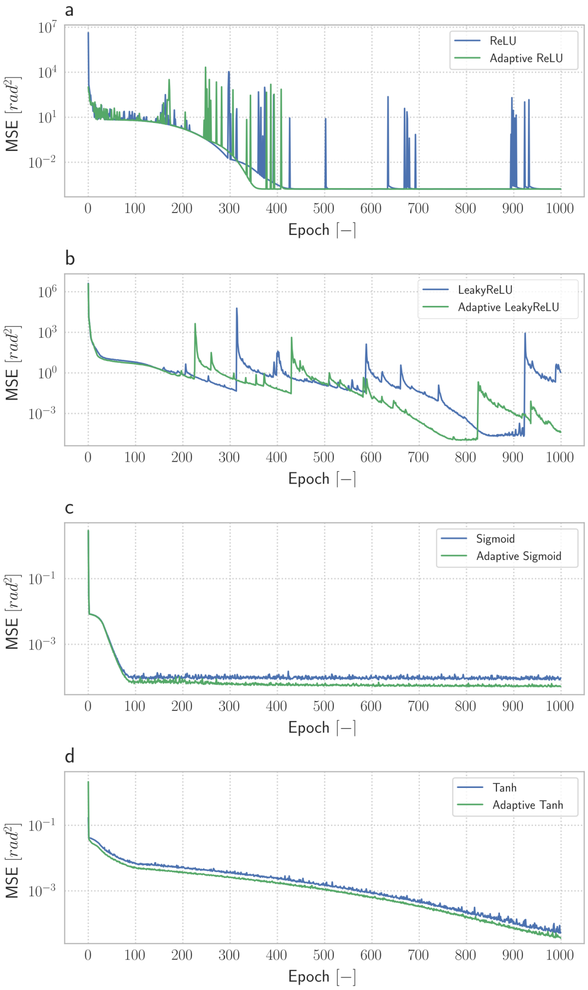

Fig. 5. Comparison of the convergence speed of the neural network to approximate voltage angle for: (a) ReLU and Adaptive ReLU, (b) LeakyReLU and Adaptive LeakyReLU, (c) Sigmoid and Adaptive Sigmoid, and (d) Tanh and Adaptive Tanh.

and is compared with that of four commonly-used activation functions, namely ReLU, LeakyReLU, Sigmoid, and Tanh.

Results indicated that the proposed adaptive activation functions significantly improve the accuracy and reliability of deep learning-based power flow analysis of distribution networks. This is achieved by adaptively modifying their shape to better fit the input data, thereby effectively capturing the nonlinear relationships between input and output. The study also highlights the increased stability of the learning process offered by the adaptive activation functions, resulting in less oscillations and more consistent performance.

Acknowledgements

This research is supported by the DATALESs project (482.20.602), funded by Netherlands Organization for Scientific Research (NWO), and National Natural Science Foundation of China (NSFC). Acknowledgments go to these organizations for their support.

References

[1] O. D. Montoya, W. Gil-González, and A. Garces, “Numerical methods for power flow analysis in DC networks: State of the art, methods and challenges,” *International Journal of Electrical Power & Energy Systems*, vol. 123, 2020.

[2] J. S. Giraldo, O. D. Montoya, P. P. Vergara, and F. Milano, “A fixed-point current injection power flow for electric distribution systems using Laurent series,” *Electric Power Systems Research*, vol. 211, 2022.

[3] T. Pham and X. Li, “Neural Network-based Power Flow Model,” in *2022 IEEE Green Technologies Conference (GreenTech)*, 2022.

[4] J. Rahman, C. Feng, and J. Zhang, “Machine Learning-Aided Security Constrained Optimal Power Flow,” in *2020 IEEE Power & Energy Society General Meeting (PESGM)*, 2020.

[5] Y. Yang and L. Wu, “Machine learning approaches to the unit commitment problem: Current trends, emerging challenges, and new strategies,” *The Electricity Journal*, vol. 34, 2021.

[6] A. Kumbhar, P. G. Dhawale, S. Kumbhar, U. Patil, and P. Magdum, “A comprehensive review: Machine learning and its application in integrated power system,” *Energy Reports*, vol. 7, 2021.

[7] A. D. Jagtap, K. Kawaguchi, and G. E. Karniadakis, “Adaptive activation functions accelerate convergence in deep and physics-informed neural networks,” *Journal of Computational Physics*, vol. 404, 2020.

[8] A. D. Jagtap and G. E. Karniadakis, “How important are activation functions in regression and classification? A survey, performance comparison, and future directions,” 2022.

[9] L. Lu, Y. Shin, Y. Su, and G. E. Karniadakis, “Dying ReLU and Initialization: Theory and Numerical Examples,” 2019.

[10] X. Wang, Y. Qin, Y. Wang, S. Xiang, and H. Chen, “ReLTanh: An activation function with vanishing gradient resistance for SAE-based DNNs and its application to rotating machinery fault diagnosis,” *Neurocomputing*, vol. 363, 2019.

[11] W. Haoxiang and S. S, “Overview of Configuring Adaptive Activation Functions for Deep Neural Networks - A Comparative Study,” *Journal of Ubiquitous Computing and Communication Technologies*, vol. 3, 2021.

[12] A. D. Jagtap, K. Kawaguchi, and G. Em Karniadakis, “Locally adaptive activation functions with slope recovery for deep and physics-informed neural networks,” *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 476, 2020.

[13] M. M. Lau and K. Hann Lim, “Review of Adaptive Activation Function in Deep Neural Network,” in *2018 IEEE-EMBS Conference on Biomedical Engineering and Sciences (IECBES)*, 2018.

[14] J. Arrillaga and B. Smith, *AC-DC power system analysis*. London: Institution of Electrical Engineers, 1998.

[15] Hadi Saadat, *Power system analysis*. McGraw Hill, 1999.

[16] S. Qian, H. Liu, C. Liu, S. Wu, and H. S. Wong, “Adaptive activation functions in convolutional neural networks,” *Neurocomputing*, vol. 272, 2018.

[17] “Power Grid Model.” [Online]. Available: https://github.com/PowerGridModel