

Article

Multilayer Perceptron, Radial Basis Function, and Generalized Regression Networks Applied to the Estimation of Total Power Losses in Electrical Systems

Giovana Gonçalves da Silva ¹ , Ronald Felipe Marca Roque ², Moisés Arreguín Sámano ³, Neylan Leal Dias ¹, Ana Claudia de Jesus Golzio ⁴  and Alfredo Bonini Neto ^{4,*} 

¹ School of Engineering, São Paulo State University (UNESP), Bauru 17033-360, SP, Brazil; giovana.goncalves@unesp.br (G.G.d.S.); neylan.dias@unesp.br (N.L.D.)

² School of Biomedical Engineering, Valparaíso University (UV), Valparaíso 2362552, Chile; ronald.marca@postgrado.uv.cl

³ Faculty of Agricultural Sciences, Bolívar State University (UEB), Guaranda 020150, Ecuador; marreguin@ueb.edu.ec

⁴ School of Sciences and Engineering, São Paulo State University (UNESP), Tupã 17602-496, SP, Brazil; a.golzio@unesp.br

* Correspondence: alfredo.bonini@unesp.br

Abstract

This paper presents an Artificial Neural Network (ANN) approach for estimating total real and reactive power losses in electrical power systems. Three network architectures were explored: the Multilayer Perceptron (MLP), the Radial Basis Function (RBF) network, and the Generalized Regression Neural Network (GRNN). The main advantage of the proposed methodology lies in its ability to rapidly compute power loss values throughout the system. ANN models are especially effective due to their capacity to capture the nonlinear characteristics of power systems, thus eliminating the need for iterative procedures. The applicability and effectiveness of the approach were evaluated using the IEEE 14-bus test system and compared with the continuation power flow method, which estimates losses using conventional numerical techniques. The results indicate that the ANN-based models performed well, achieving mean squared error (MSE) values below the predefined threshold during both training and validation (0.001). Notably, the networks accurately estimated the total power losses within the expected range, with residuals on the order of 10^{-4} . Among the models tested, the RBF network showed slightly superior performance in terms of error metrics, requiring fewer centers to meet the established criteria compared to the MLP and GRNN models (11 centers). However, the GRNN achieved the shortest processing time; even so, all three networks produced satisfactory and consistent results, particularly in identifying the critical points of electrical power systems, which is of fundamental importance for ensuring system stability and operational reliability.

Keywords: artificial intelligence; power losses; forecasting; electric power systems; artificial neural networks



Academic Editor: Marek Sikora

Received: 4 November 2025

Revised: 9 December 2025

Accepted: 23 December 2025

Published: 26 December 2025

Copyright: © 2025 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

In the supply of electrical energy to consumers, a significant portion is dissipated in transmission and distribution systems, resulting in both technical and non-technical losses. Technical losses, caused by inefficiencies in equipment and infrastructure, generate considerable economic and environmental impacts [1]. For this reason, mitigating such

losses must be addressed comprehensively, regardless of the institutional structure of the power sector or the ownership of utilities. The level of losses in these systems ranges between 3% and 13%, highlighting their relevance to overall efficiency [1].

Among the main tools used in power system analysis, load flow calculation plays a central role, supporting activities such as expansion planning, network reconfiguration, and storage capacity sizing [2–4]. The classical formulation based on the Newton method is widely employed in both transmission and distribution systems due to its effectiveness [5]. However, the requirement to construct and factorize the Jacobian matrix at each iteration makes the process computationally demanding, especially in large-scale systems [6,7].

Over the past decades, several studies have explored the use of artificial neural networks (ANNs) as an efficient alternative for power system analysis [8–13]. In [8,9], Multilayer Perceptron (MLP) networks were applied to predict total real and reactive power losses under pre- and post-contingency conditions, achieving high accuracy (99% of losses estimated within the expected range), with mean squared error below the specified threshold. Additionally, a graphical user interface was developed to facilitate real-time visualization and adjustments [8].

In [10], two ANN architectures—MLP and Radial Basis Function (RBF) networks—were evaluated for estimating bus voltage magnitudes. The study considered variables such as loading factor, real and reactive power at the slack bus, and the number of branches under contingency. Results showed that the RBF network achieved superior performance, with an error on the order of 10^{-4} .

Research in other domains has also demonstrated the potential of artificial intelligence. In [11], AI was used to identify predictive biomarkers for diffuse large B-cell lymphoma prognosis, employing MLP and RBF networks. In [12], an MLP network was applied to automatically predict the geometry of injection-molded products, achieving an accuracy higher than 92%. In [13], different ANN algorithms were employed for voltage stability analysis, further confirming the viability of the approach.

There are few studies in the literature involving GRNN applications, particularly in electrical power systems.

In [14], the applications of General Regression Neural Networks (GRNN) in dynamic systems were presented, highlighting their universal approximation and data learning capabilities, demonstrating their potential for regression, classification, and forecasting problems. In [15], the GRNN was proposed as a four-layer feedforward network that performs regression through a probability density function. The pattern centers are defined using clustering techniques such as K-means, and the predicted value is obtained from the ratio between the computed sums in the output layer. Five statistical measures were used to compare the models, showing that the Backpropagation approach was more accurate, whereas the GRNN exhibited higher precision in the results. In [16], a deep learning system was proposed for real-time voltage stability assessment in electrical power systems. Five neural network architectures were compared using the New Voltage Stability Pointer (NVSP) indicator, with the Feedforward Neural Network (FFNN) and Cascade-Forward Neural Network (CFNN) demonstrating superior performance in accuracy and efficiency on the IEEE 30-bus and Nigerian power systems.

Building on these advances, ANNs have emerged as promising tools for modeling and optimization in power systems, reducing dependence on more complex and computationally expensive conventional methods.

In this context, and aiming to enhance the generalization capability of the models, this work proposes the use of ANNs to estimate total real and reactive power losses in electrical power systems. In addition to the variables commonly reported in the literature—such as loading factor (λ) and real and reactive power at the slack bus (P_g^{slack} e Q_g^{slack})—the

proposed model also includes current injections from all system buses as input variables. Three architectures are investigated—Multilayer Perceptron (MLP), Radial Basis Function (RBF), and Generalized Regression Neural Network (GRNN)—and their performances are compared both with each other and with the results obtained from continuation power flow, which is used as the reference.

2. Materials and Methods

The system analyzed in this study corresponds to the IEEE 14-bus configuration, presented in Figure 1. The 193 samples used for training and validation of the neural networks were obtained according to the method described in [6]. Each sample consists of 19 data points, of which 17 are used as inputs to the ANN—the loading factor (λ), the real and reactive power generated at the reference bus (P_g^{slack} and Q_g^{slack}), and the currents at all system buses—and 2 are used as outputs, representing the total real (Pa) and reactive (Pr) power losses of the system.

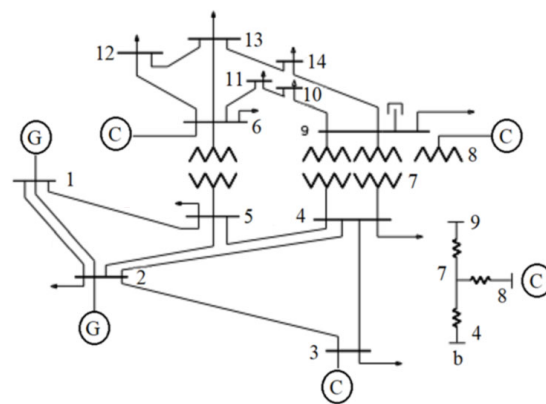


Figure 1. IEEE 14-bus test system utilized in this work.

The use of currents injected into the system buses as input variables for the artificial neural network (ANN) represents a significant advancement over traditional approaches, as it substantially enhances the model's generalization capability and its ability to more accurately capture the dynamic characteristics of the power system. While previous studies, such as those by [8,9], focused primarily on conventional variables—such as the loading factor and the real and reactive power generated at the reference bus—the incorporation of current data provides a more comprehensive and detailed representation of the system's operational state. This addition enables the ANN to recognize more complex interaction patterns among network components, reflecting not only static conditions but also variations associated with different operating regimes. As a result, the developed model becomes less dependent on scenario-specific adjustments and demonstrates greater robustness and generalization capacity, significantly improving the accuracy of total real and reactive power loss prediction under a wide range of operating conditions.

In electrical power systems, obtaining real and reactive power losses typically requires solving the iterative power flow problem to determine all bus voltages and voltage angles—an approach that becomes computationally demanding, especially in scenarios such as contingency analysis or continuation power flow near the critical point. The proposed neural network model overcomes this limitation by directly estimating total power losses without performing a new iterative power flow. This capability enables significantly faster assessments, making the method suitable for applications that require rapid evaluation of multiple operating conditions.

The artificial neural networks (ANNs) employed in this study encompass three distinct architectures: a multilayer perceptron (MLP) trained using the backpropagation algorithm, a radial basis function network (RBF), and a generalized regression neural network (GRNN). All architectures were structured into three layers. The input layer consists of 17 neurons corresponding to the system's input variables: the loading factor (λ), the real and reactive powers generated at the slack bus (P_g^{slack} and Q_g^{slack}), and the currents injected into the 14 buses of the electrical system, following the methodology established in [16], which employs line and bus data extraction as input parameters. The hidden layer comprises 20 neurons in the case of the MLP, whereas in the RBF network, the number of hidden neurons is defined by s , which represents the number of centers of the network and, consequently, the number of radial basis functions ($1 \leq s \leq p$, where p denotes the number of available samples). The 20 neurons in the hidden layer of the MLP were determined using the algorithm proposed in [17]. After repeating the procedure n times, the best-performing configuration was retained, corresponding to the most effective combination of hidden layers and neurons that yielded the highest validation accuracy.

The GRNN architecture, in turn, follows the principle of probability density function estimation and employs a hidden layer with one neuron per training sample, similar to the RBF network [14,15,18]. This structure enables fast response and strong generalization capability, even when using relatively small datasets.

The output layer, common to all three architectures, is composed of two neurons representing the target variables of the model: the total real and reactive power losses of the system. The inclusion of the GRNN in this study aims to complement the performance analysis of the MLP and RBF networks, providing a robust alternative for nonlinear regression problems and enabling the comparison of the efficiency and accuracy of different ANN paradigms in power loss estimation.

All processes related to data preparation, model training, and validation, as well as the computation of the presented results, were carried out using Matlab® R2024a [18] on a computer equipped with an Intel® Core™ i7-8750H CPU @ 2.20 GHz and 16 GB of RAM. Figure 2 illustrates the general structure of the neural networks employed in this research.

Figure 2a illustrates the structure of a Multilayer Perceptron (MLP) network used for estimating real and reactive power losses. The input vector x , composed of variables such as loading factor (λ), slack bus real power (P_g^{slack}), reactive power (Q_g^{slack}), and nodal currents, is processed through a nonlinear mapping defined by the network.

The first operation involves a weighted sum between the input data and the first weight matrix W_{mm} , added to the bias vector b_1 . This combination passes through a nonlinear activation function, specifically the hyperbolic tangent function (Equation (1)), expressed as:

$$a = \frac{(1 - e^{-tu})}{(1 + e^{-tu})} \quad (1)$$

where u represents $(W_{mm} \cdot x + b_1)$.

This function introduces nonlinearity into the hidden layer, allowing the network to learn complex relationships between the inputs and outputs. The resulting activation vector is then forwarded to the output layer, where a new weighted combination with W_{mi} and bias b_2 is performed $n_2 = W_{mi} \cdot a + b_2$.

In this stage, a linear activation function is used, generating the final network output $Y_{ob} = n_2$. The output Y_{ob} represents the estimated values of real power (Pa) and reactive power (Pr) losses.

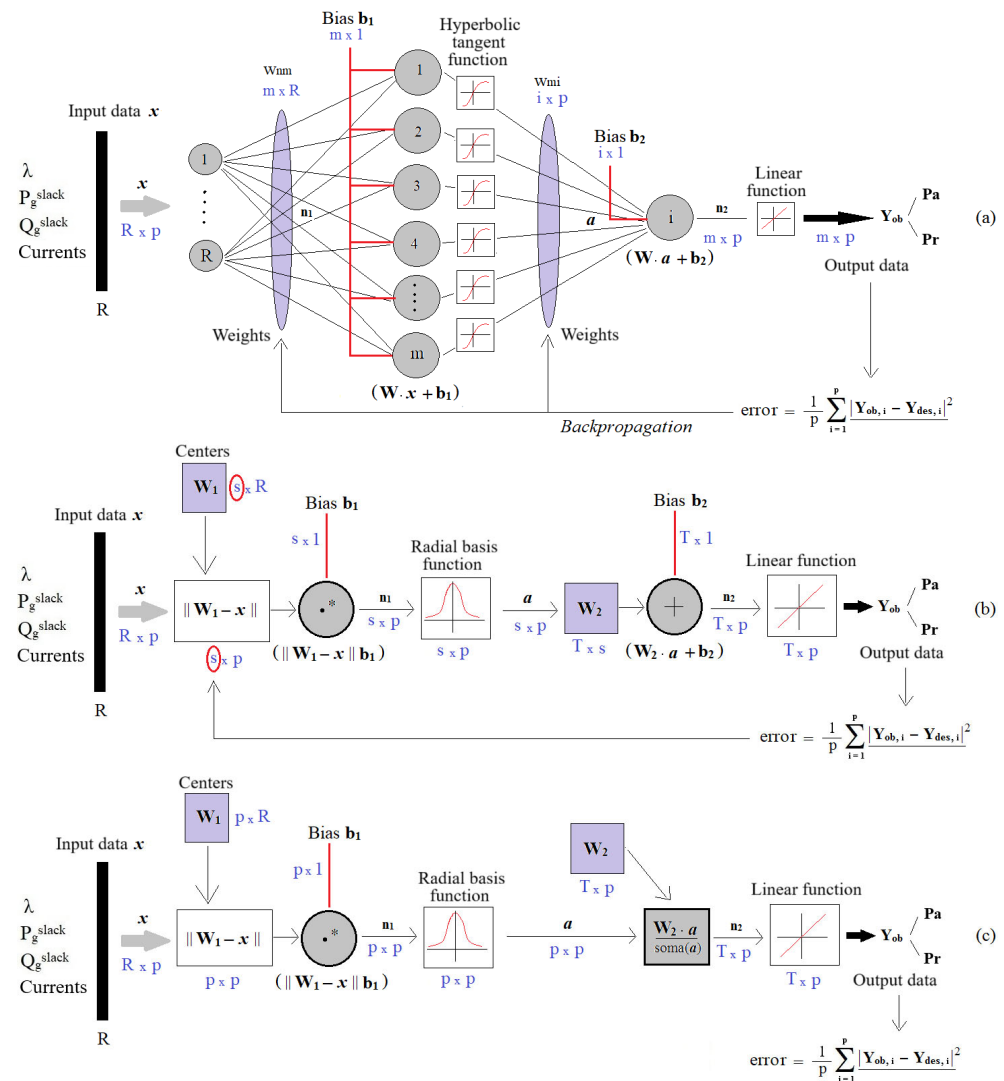


Figure 2. ANNs used in this work: (a) MLP, (b) RBF, and (c) GRNN.

Figure 2b represents the architecture of the Radial Basis Function (RBF) network, whose hidden layer performs a nonlinear transformation of the input data $x \in R^{R \times p}$ through Gaussian functions centered at vectors $W_1 \in R^{s \times R}$, referred to as centers. Each neuron computes the Euclidean distance between the input vector and its corresponding center $\|W_1 - x\|$, adding a bias term $b_1 \in R^{s \times 1}$. The resulting activation is obtained through the radial basis function, given by Equation (2):

$$a_i = e^{\left(-\frac{\|x - c_i\|^2}{2\sigma^2}\right)} \quad (2)$$

where c_i represents the center of the i -th neuron and σ is the spread parameter (Gaussian RBF width). The outputs of the hidden layer (a) are then linearly combined through the weights $W_2 \in R^{T \times s}$ and the bias $b_2 \in R^{T \times 1}$, resulting in the final output Y_{ob} , expressed as $Y_{ob} = W_2 a + b_2$. The weights W_2 are adjusted by least squares, minimizing the mean squared error (MSE) between the obtained and desired outputs, as shown in Figure 2. The Mean Squared Error (MSE) between two vectors Y_{des} (desired values) and Y_{ob} (observed values) is given by Equation (3):

$$\text{MSE} = \frac{1}{p} \sum_{i=1}^p |Y_{ob,i} - Y_{des,i}|^2 \quad (3)$$

where

p = number of samples;

$Y_{\text{des},i}$ = desired (target) value of the i -th sample;

$Y_{\text{ob},i}$ = observed (predicted) value of the i -th sample.

Figure 2c, in turn, illustrates the Generalized Regression Neural Network (GRNN), which preserves the same functional form of the radial activation but with a fundamental difference: the centers $c_i = W_1$ coincide exactly with the training samples ($p \times p$), so there is no iterative adjustment of these parameters. Each neuron in the hidden layer computes a Gaussian function (radial basis function), analogous to Equation (2):

The output layer then performs a weighted average of the training outputs associated with each input vector, where the term $W_2 \cdot a$ is normalized by the sum of activations ($\text{sum}(a) = \sum a$), as indicated in Figure 2c. Each element is the dot product of a row of W_2 with the input vector a , all normalized by the sum of the elements of a . Thus, the final output is given by Equation (4):

$$n_2 = Y_{\text{ob}} = \frac{W_2 \cdot a}{\sum a} \quad (4)$$

In this case, learning does not involve weight adjustment through backpropagation but rather a direct statistical interpolation, controlled solely by the smoothing parameter σ . In summary, the RBF network (Figure 2b) requires supervised training with explicit adjustment of output weights, whereas the GRNN (Figure 2c) performs an instantaneous estimation based on all data points, being a non-parametric and instant-training method. However, GRNN tends to be more computationally demanding for large datasets due to its dependence on all training samples as reference centers.

3. Results and Discussion

The same dataset was submitted to the three modeling approaches presented: MLP, RBF, and GRNN. Of the 193 available samples, 80% (154 samples) were randomly selected for training, while the remaining 20% (39 samples) were used for validation. The performance of the neural network during training and validation using the MLP model is illustrated in Figure 3a. It can be observed that the best performance, measured by the mean squared error (MSE), was achieved after four iterations (epochs). During training, the MSE reached 3.38×10^{-4} , whereas for validation, the error was slightly lower, 3.04×10^{-4} , both occurring at the fourth iteration, as detailed in Table 1.

The training process of the neural network involves presenting the model with a dataset (in this case, 154 samples) so it can learn to map inputs to the desired outputs, adjusting internal weights and parameters to minimize error. Validation, performed with a separate set of 39 samples, assesses the network's generalization capability, i.e., how well the model predicts results for unseen data.

The total training process took approximately 3 s for the four iterations. Figure 3b presents the error histogram, which shows the distribution of the differences between the desired output and the MLP-predicted output ($|Y_{\text{des}} - Y_{\text{ob}}|$) with respect to the zero-error line. The results indicate that the errors remained predominantly close to zero, demonstrating the strong ability of the network to accurately approximate the target values (obtained from the continuation power flow proposed in [6]). This suggests that the trained model exhibits good generalization capability for the analyzed data.

Nevertheless, as shown in Figure 3b, most of the errors—both negative and positive—remain concentrated around zero. On average, the results are very similar to those obtained with the RBF network: the MLP exhibits slightly smaller negative errors and slightly

larger positive errors; however, the overall performance remains comparable, with a slight advantage in favor of the RBF model.

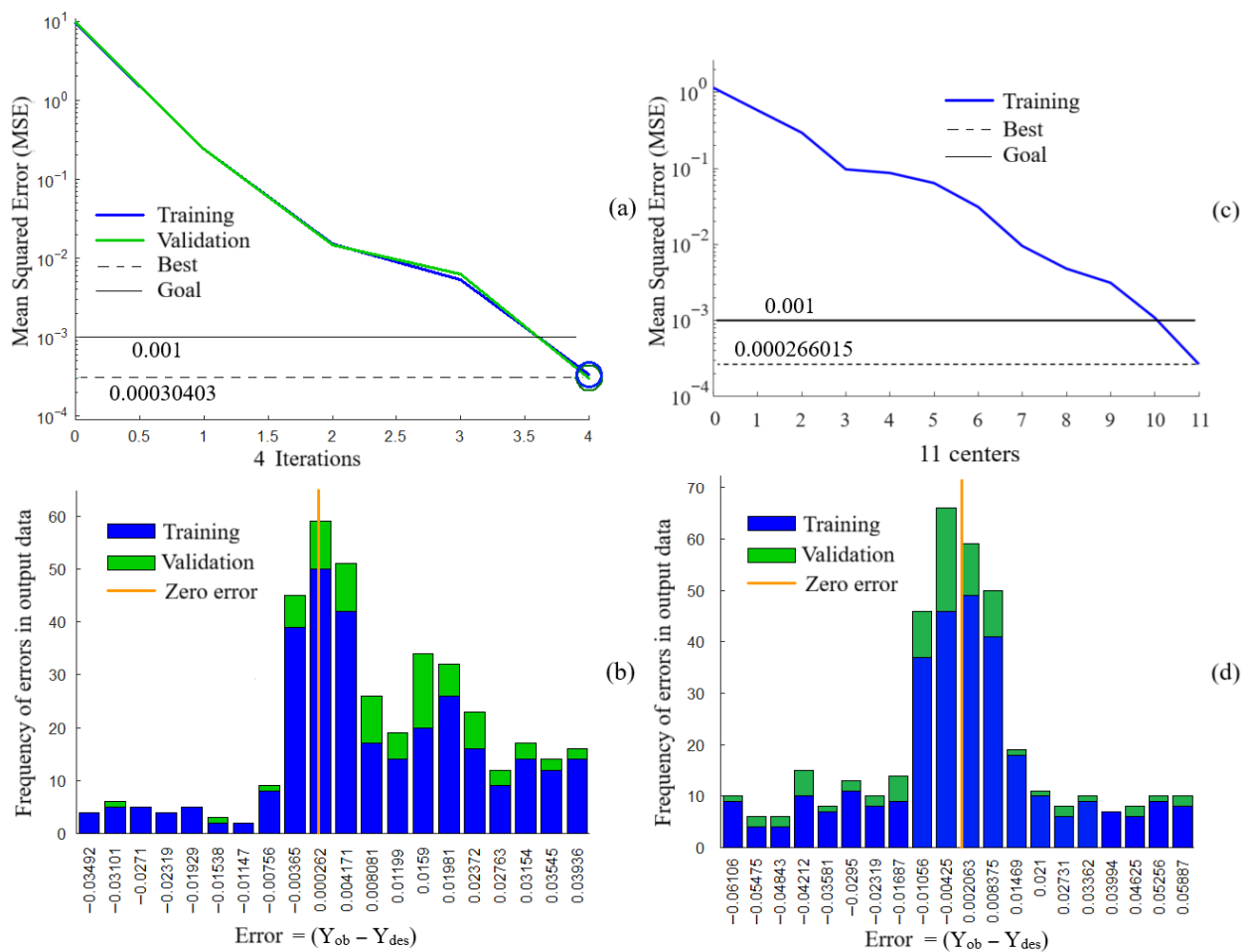


Figure 3. Performance of ANNs (MLP and RBF): (a) MSE of MLP network, (b) error histogram ($Y_{des} - Y_{ob}$) of MLP network, (c) MSE of RBF network, and (d) error histogram ($Y_{des} - Y_{ob}$) of RBF network.

Table 1. Specified and achieved values in the training and validation of the MLP ANN compared to the desired output.

	Specified Values	Achieved Values
Iterations	100	4
Time (s)	60	3
MSE—Training	0.001	* 0.0003834
MSE—Validation	0.001	* 0.0003040

* Achieved criterion.

The operation of the MLP neural network follows a feedforward architecture, in which the input data (λ , P_g^{slack} , Q_g^{slack} , and currents) are processed through multiple layers. In the hidden layer, the hyperbolic tangent activation function is applied to the weighted inputs (W_{nm}) plus the bias term (b_1), producing a nonlinear output that is subsequently transmitted to the output layer. This structure enables the modeling of complex relationships among the variables to estimate the desired parameters (Pa and Pr).

The backpropagation algorithm constitutes the core of the MLP learning process, enabling the iterative adjustment of the synaptic weights (W) of both layers based on the

difference between the predicted and target values, as shown in [19]. By computing the gradient of the mean squared error (MSE), the error is backpropagated through the network layers, allowing the gradual update of the weights to minimize the observed discrepancy—evidenced, for instance, in the error histogram. This supervised optimization process, completed in only four epochs and yielding an MSE on the order of 10^{-4} , highlights not only the computational efficiency of the MLP but also its fast and stable convergence, which are essential characteristics for applications requiring high precision and reliability in the modeling of complex nonlinear systems, as demonstrated in [20].

Figure 3c shows the evolution of the mean squared error (MSE) of the radial basis function (RBF) network over 11 iterations (centers). It can be observed that the error progressively decreases during training, indicating that the network efficiently adjusts the number of centers (or radial basis functions) to minimize the error. The lowest MSE achieved was approximately 0.000266015, a considerably small value, below the established threshold of 0.001. To reach this performance, the network adjusted the number of radial basis functions to $s = 11$, ensuring a good balance between model complexity and accuracy. In addition, the dashed line represents the best performance achieved, reinforcing that the model successfully converged to a satisfactory error level in just a few iterations.

Figure 3d displays the error histogram, where the error values (the difference between the desired output and the RBF-predicted output) are concentrated near zero. This indicates that the neural network accurately predicted the output values for both the training set (blue bars) and the validation set (green bars). The presence of a central peak reinforces the low variability of the errors, suggesting that the RBF network achieved a good fit to the data.

The structure and operation of the radial basis function (RBF) neural network used in this study are illustrated in Figure 4b. The model consists of three main layers: an input layer that receives the variables λ , P_g^{slack} , Q_g^{slack} , and currents; a hidden layer with $s = 11$ radial basis functions acting as nonlinear transformation units; and a linear output layer responsible for estimating the desired parameters (Pa and Pr). Each neuron in the hidden layer computes the Euclidean distance between the input vector and its corresponding center (W_1), modulated by the bias term (b_1). The resulting activation is processed through a Gaussian-type radial function, which defines the degree of similarity between the input and each center. The output layer then linearly combines these activations using weights (W_2) and bias (b_2) to generate the network outputs.

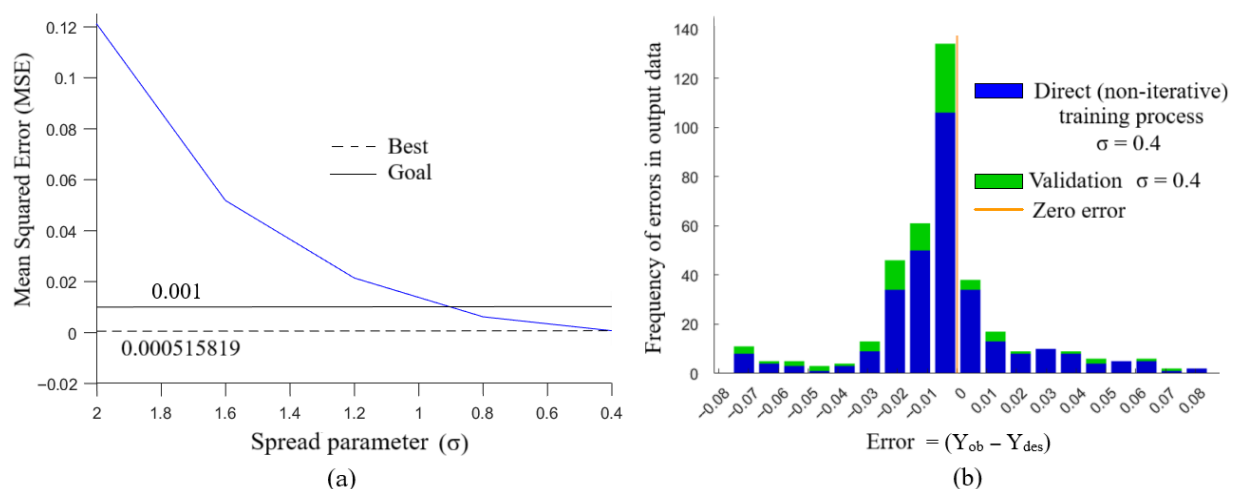


Figure 4. Performance of ANNs (GRNN): (a) MSE performance; (b) error histogram ($Y_{des} - Y_{ob}$).

The mean squared error (MSE) of the RBF network progressively decreased over 11 iterations, reaching a minimum value of approximately 2.66×10^{-4} , well below the target threshold of 10^{-3} .

This behavior indicates that the network effectively adjusted the number of radial centers to optimize performance while maintaining a balance between model complexity and accuracy. The convergence observed after only a few iterations reflects the inherent efficiency of RBF networks in capturing nonlinear relationships, consistent with the results reported by [21,22], which highlighted the fast convergence and low generalization error of RBF-based models in complex system modeling.

The corresponding error histogram further corroborates the quality of the network's predictions. The error distribution remained strongly concentrated around zero for both the training and validation datasets, with minimal dispersion. This narrow distribution demonstrates that the RBF network achieved a high level of precision in approximating the target values, confirming its robustness and generalization capability. Similar findings have been reported in the literature, where RBF architectures are recognized for their ability to approximate nonlinear mappings with fewer training iterations compared to multilayer perceptrons (MLPs).

The RBF network exhibited a stable and efficient learning process, achieving a low MSE and a compact error distribution in just a few iterations. Overall, the results indicate that the network demonstrated good learning and generalization performance, minimizing error and providing accurate predictions with only 11 radial basis functions. These findings reinforce its suitability for modeling nonlinear behaviors in power system estimation problems, aligning with trends described in recent studies on hybrid and data-driven modeling approaches. The corresponding results are summarized in Table 2. The results obtained using the RBF neural network were computed with the default spread parameter value ($\sigma = 1$).

Table 2. Specified and achieved values in the training and validation of the RBF ANN compared to the desired output.

	Specified Values	Achieved Values
Iterations (centers)	100	11
Time (s)	60	2
MSE—Training	0.001	* 0.0002660
MSE—Validation	0.001	* 0.0003014
Radial basis functions (RBF)	$p = 193$	11

* Achieved criterion, p = number of samples.

The architecture of the Generalized Regression Neural Network (GRNN) used to estimate the total real and reactive power losses (P_a and P_r) in the power system follows a structure similar to that of a Radial Basis Function (RBF) network but with a fundamental distinction: the number of centers in the radial basis layer is fixed and equal to the number of samples used for direct training, in this case, $p = 193$. Each input vector is compared with all centers, and the resulting Euclidean distances are processed by the radial basis function, typically Gaussian. The spread (σ) parameter plays a crucial role in determining the width of the Gaussian function and, consequently, the smoothness and generalization ability of the model. An inadequate σ may lead to overfitting or underfitting, emphasizing the importance of careful tuning. The activation outputs are then linearly combined in the summation layer, where the weights (W_2) are adjusted to minimize the mean squared error (MSE) between the observed (Y_{ob}) and desired (Y_{des}) outputs. As reported in [23], this type of network provides fast direct training and robust approximation performance for nonlin-

ear regression problems, demonstrating high adaptability to data-driven estimation tasks. The results obtained in this study corroborate these findings, confirming the suitability of the GRNN for modeling complex nonlinear relationships in power system estimation.

Despite its efficiency and fast learning capability, the GRNN has been relatively underexplored in the literature, especially concerning its application to power system estimation problems. Most studies in this field have focused on Multilayer Perceptrons (MLP) or Radial Basis Function Networks (RBF).

For the GRNN, the value of the spread parameter (σ) was adjusted by varying it in increments of $\Delta\sigma = 0.4$, starting from an initial value of $\sigma_0 = 2.0$, according to the relation $\sigma_{i+1} = \sigma_i - \Delta\sigma$. For each σ value, the corresponding mean squared error (MSE) was recorded, resulting in the graph shown in Figure 4a, which depicts the variation in MSE as a function of σ . The best performance was observed for $\sigma = 0.4$, which was therefore adopted for generating the final GRNN results.

It is important to note that excessively small values of the spread parameter (σ) can lead to poor generalization capability of the network. In such cases, the Gaussian functions become too narrow, causing the GRNN to fit the direct training data too closely—a phenomenon known as overfitting. This results in high sensitivity to noise and large variations in the predicted outputs for unseen data. Conversely, excessively large σ values may cause the Gaussian functions to overlap excessively, producing an overly smoothed response and loss of detail in the estimated behavior. Therefore, selecting an appropriate σ is crucial to ensure a balance between accuracy and generalization.

Figure 4b shows the error histogram of the GRNN with spread $\sigma = 0.4$, demonstrating that most errors are concentrated near zero. The symmetric error distribution indicates good generalization capability of the network, with few significant errors at the extremes. The direct (non-iterative) training process of the GRNN enabled fast convergence, with similar performance between training and validation datasets.

Table 3 presents the comparison between the specified values and those achieved during the direct training and validation of the GRNN. The training time was only 0.5 s, demonstrating the efficiency of the non-iterative training process of the GRNN. The MSE values for both training (0.0005158) and validation (0.0006932) remained below the specified target (0.001), indicating excellent accuracy. The spread parameter (σ) was adjusted to 0.4, providing better network generalization. Table 4 presents the MSE values for each spread (σ) value.

Table 3. Specified and achieved values in the direct training and validation of the GRNN ANN compared to the desired output.

	Specified Values	Achieved Values
Time (s)	60	0.5
MSE—Direct training	0.001	* 0.0005158
MSE—Validation	0.001	* 0.0006932
Spread (σ)	0.1	0.4

* Achieved criterion, σ = width of the Gaussian function (radial basis function).

The Generalized Regression Neural Network (GRNN) had the shortest training time among the three architectures analyzed, completing the process in just 0.5 s, which represents a significant reduction compared to RBF (2 s) and MLP (3 s). This difference is due to the fact that GRNN uses a direct and non-iterative training process, based solely on the definition of the spread parameter (σ) and the analytical calculation of the output weights, without the need for multiple weight updates through error backpropagation. This procedure substantially reduces computational cost and eliminates the iterative step

typical of MLP and RBF networks. Furthermore, GRNN demonstrated excellent accuracy, with MSE values below the specified target, confirming its efficiency and suitability for estimation problems in electric power systems.

Table 4. MSE with respect to the spread (σ) values.

Spread (σ)	MSE
2.0	0.126839
1.6	0.055701
1.2	0.022710
0.8	0.006419
0.4	* 0.000515

* Achieved criterion.

An increase in the number of centers in the Radial Basis Function (RBF) Network can raise the computational complexity and processing time, potentially surpassing that of the Multilayer Perceptron (MLP). This occurs because each additional center introduces more distance calculations and increases the computational demand for inverting the weight matrix in the output layer. Thus, although the RBF network is faster for a moderate number of centers, an excessive growth in centers may compromise its efficiency.

Table 5 reveals a clear trend: as the number of radial basis functions (s) increases, the mean squared error (MSE) progressively decreases. This behavior indicates that the RBF network is efficiently refining its model, improving its approximation capability. The process continues until the MSE reaches the pre-established value of 0.001, at which point the network achieves a balance between complexity and accuracy. This result demonstrates that selecting an appropriate number of radial basis functions is essential to ensure good network performance, avoiding both underfitting and overfitting of the data.

Table 5. MSE with respect to the number of radial basis functions.

Radial Basis Functions (s)	MSE
1	1.14705
2	0.295114
3	0.0972069
4	0.0870737
5	0.0641629
6	0.0310596
7	0.0095354
8	0.0047875
9	0.00312592
10	0.00108671
11	* 0.000266015

* achieved criterion.

Figure 5 presents the comparative performance of the three neural network architectures analyzed (MLP, RBF, and GRNN), comparing the outputs obtained by the networks with the reference output derived from the continuation power flow [6]. The dataset was divided into 80% for training and 20% for validation in the MLP network. The same samples were used for the training and validation of the RBF and GRNN, ensuring equivalent analysis conditions and enabling an accurate and consistent comparison among the models.

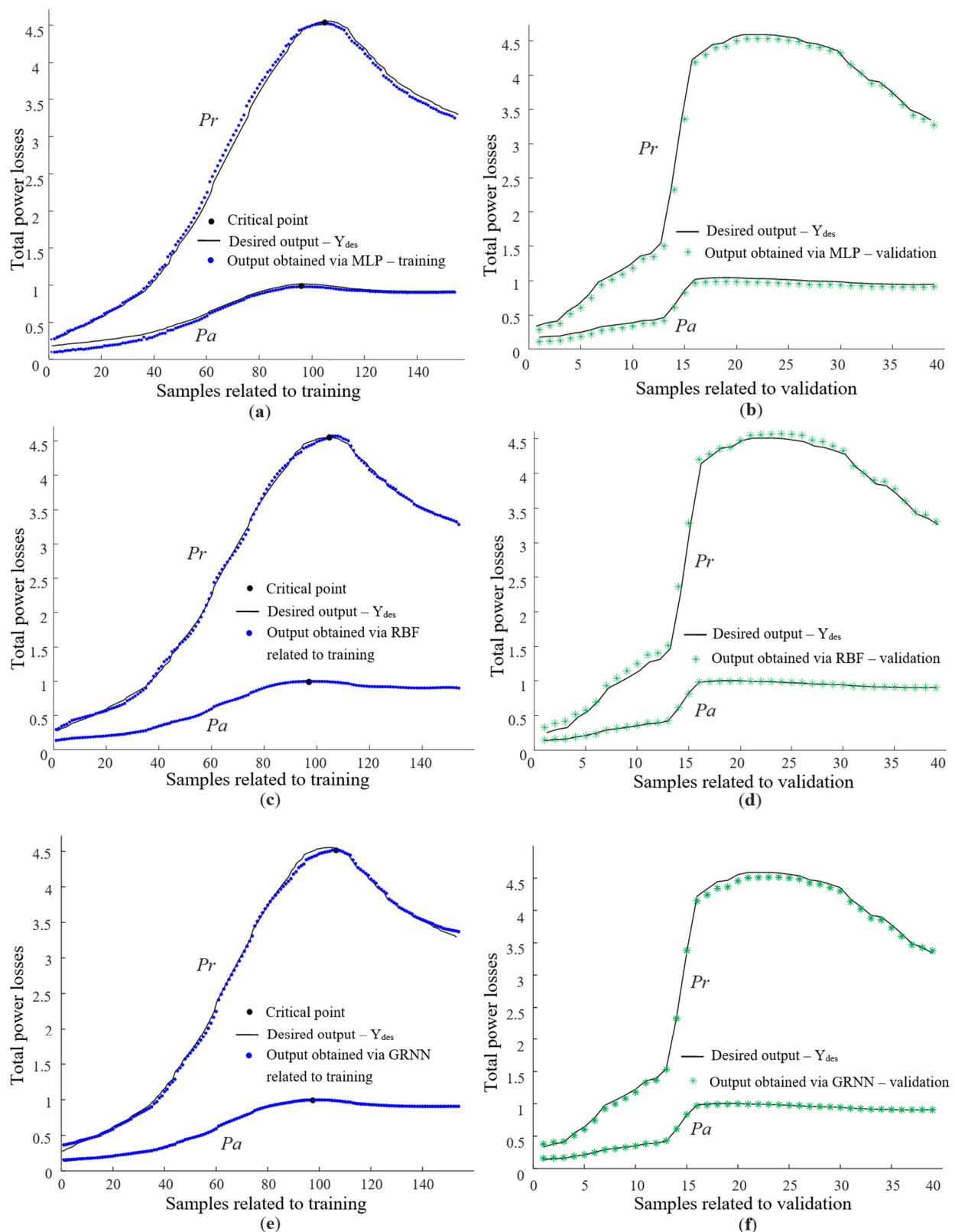


Figure 5. Comparison between the output obtained by the neural network and the desired output for training (154 samples) and validation (39 samples): (a) MLP training; (b) MLP validation; (c) RBF with $s = 11$, training samples; (d) RBF validation; (e) GRNN with $\sigma = 0.4$, training samples; (f) GRNN validation.

Figure 5a presents the P_a and P_r curves as a function of the training samples (154 samples), while Figure 5b illustrates the results for the samples not included in this

stage, i.e., those corresponding to the validation phase. It can be observed that the MLP neural network achieved satisfactory performance, closely matching the desired values, with errors of 0.0003834 and 0.0003040 in the training and validation stages, respectively.

Similar results were obtained with the RBF network, as shown in Figure 5c,d. The MSE values obtained for the training and validation samples were 0.0002660 and 0.0003014, respectively. It can be observed that the network accurately estimated the total real (Pa) and reactive (Pr) power losses for all samples, achieving values very close to the desired ones.

Figure 5e,f present the results obtained with the GRNN, which also demonstrated good estimation capability, with an MSE of approximately 0.0005 for both the training and validation samples.

It is important to note that, although the networks have distinct architectures and learning mechanisms, the results obtained do not show significant differences that would justify the preference for one model over another. This indicates that all analyzed architectures—MLP, RBF, and GRNN—were able to adequately represent the system behavior, achieving comparable performance in terms of error and generalization capability. Consequently, such reliable estimation of system variables is essential when evaluating critical operating conditions. In the context of electrical power systems, identifying the critical point is crucial to determine the maximum load the system can handle before instabilities, such as voltage drops or overloads, occur. Knowing this point is essential for ensuring the system's safety, reliability, and efficient operational planning [6].

Figure 6 presents a comparison between the output obtained by the MLP network (Y_{ob}) and the desired output (Y_{des}) for the total real (Pa) and reactive (Pr) power losses as a function of the loading factor (λ). Figure 6a shows the λ - Pa curve. The black line represents the desired output (Y_{des}) obtained from the continuation power flow, while the blue and green points indicate the outputs obtained by the MLP during the training and validation phases, respectively. It can be observed that the neural network estimated the output with good accuracy, with values generally close to the desired ones throughout the entire curve. However, subtle differences can be noted under light load conditions and in the region of maximum losses, where the output obtained by the MLP shows small deviations compared to the desired output. Despite these differences, the model still demonstrated satisfactory performance, especially in the region near the critical point, where the system reaches its maximum loading condition.

Figure 6b shows the λ - Pr curve. In this case as well, the neural network was able to reproduce the curve with good accuracy, highlighting its generalization capability, including in the critical point region. In both figures, it can be observed that the MLP effectively estimated the total real and reactive power losses, providing results consistent with those obtained from the continuation power flow, which confirms the suitability of the proposed model.

Figure 6c,d presents the performance of the Radial Basis Function (RBF) neural network, comparing the outputs obtained with the desired outputs derived from the continuation power flow. To achieve the pre-established error, the network was configured with 11 radial basis functions ($s = 11$), ensuring accurate estimation of the total real (Pa) and reactive (Pr) power losses. The Figures show the λ - Pa and λ - Pr curves, highlighting the network's ability to capture the system behavior across the entire operating range. A slight improvement can be observed in the accuracy of the curves obtained by the RBF network compared to the MLP network. It is important to note that the green points represent samples that were not included in the training phase, yet they remained very close to the desired output, demonstrating the network's strong generalization capability.

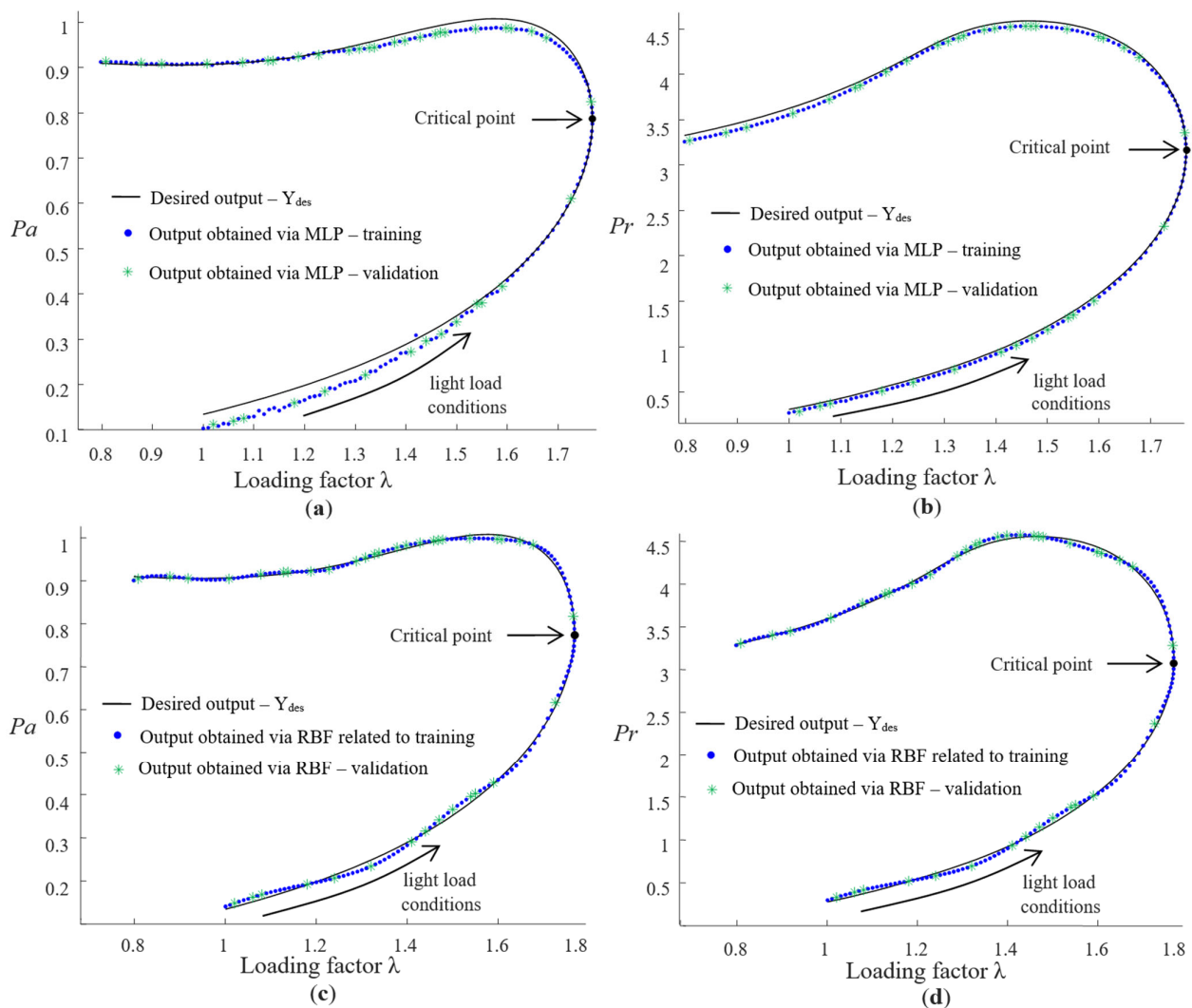


Figure 6. λ - P_a and λ - P_r curves obtained using the neural network outputs (Y_{ob}) compared with the desired outputs (Y_{des}) for all samples (193). (a) λ - P_a curve obtained via MLP, (b) λ - P_r curve obtained via MLP, (c) λ - P_a curve obtained via RBF ($s = 11$), and (d) λ - P_r curve obtained via RBF ($s = 11$).

Figure 7 illustrates the performance of the GRNN in estimating the total real and reactive power losses, considering different values of the radial basis Gaussian function width parameter (σ). Specifically, Figure 7a presents the results for real power losses, while Figure 7b shows the reactive power losses as a function of the loading factor λ , highlighting the effect of adjusting the σ parameter on the accuracy of the network's predictions. The best performance was achieved with a spread value of 0.4 (red line).

Although the values obtained by the three networks are very close to each other—indicating that any of them could be used to estimate the total power losses—Figure 8a shows that the RBF network achieved the best overall performance. This network was able to estimate the desired values with high accuracy (error of 0.0002660), including in the region near the critical point, demonstrating a closer approximation than that obtained by the MLP and GRNN. Moreover, the RBF maintained more consistent performance, with outputs even closer to the desired values, especially in regions with greater variation in losses. Despite small discrepancies observed under light-load conditions and near the maximum-loss region, the RBF network presented the lowest mean error (0.0003014), confirming its superior ability to estimate total real and reactive power losses compared to the MLP and GRNN, which exhibited errors of 0.0003040 and 0.0006932, respectively.

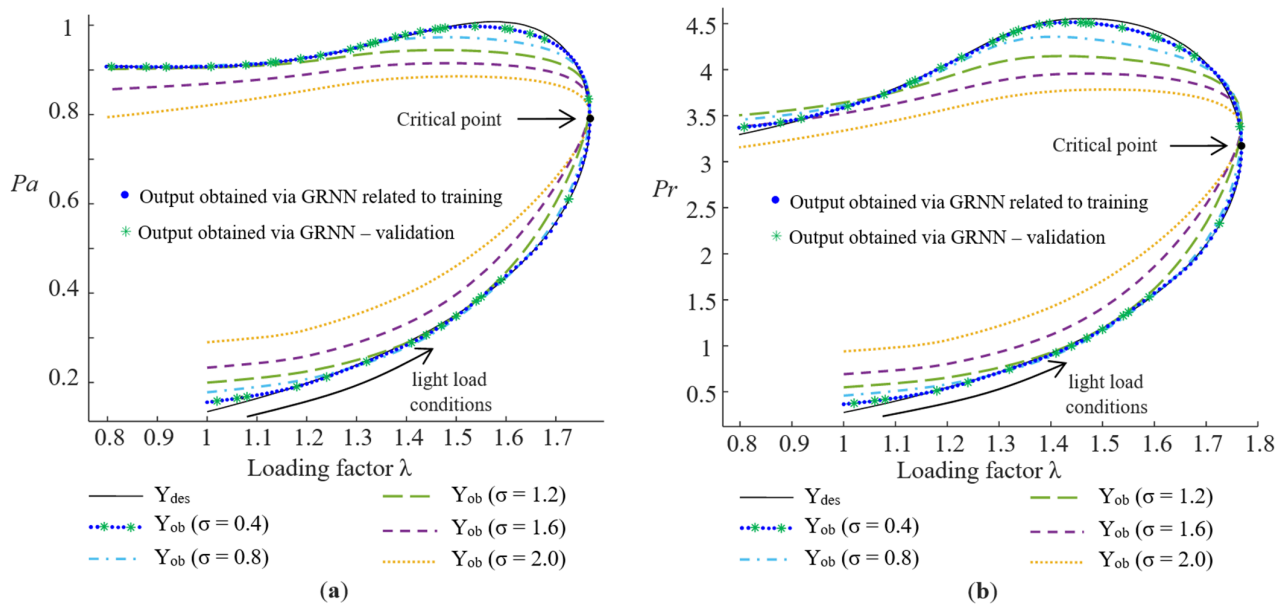


Figure 7. Performance of the GRNN in estimating total power losses for different values of the spread parameter (σ) as a function of the loading factor λ , λ - P_a and λ - P_r curves: (a) real; (b) reactive.

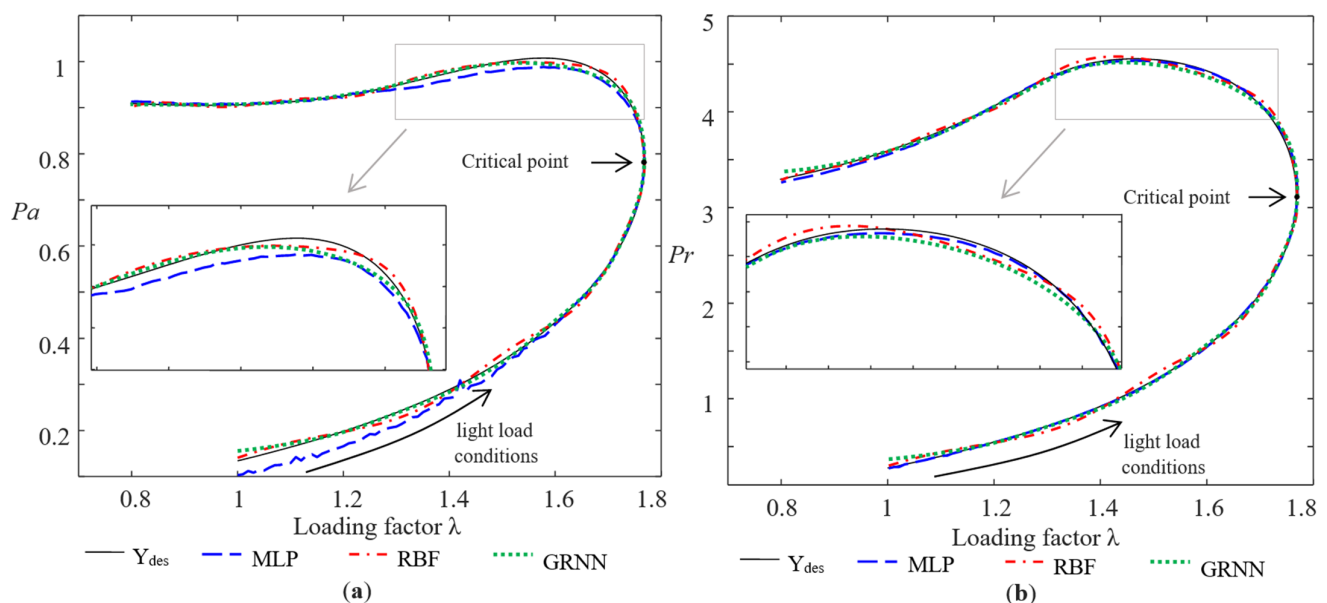


Figure 8. Comparison of predicted outputs (Y_{ob}) from MLP, RBF, and GRNN against the desired output (Y_{des}): (a) real power (P_a) versus loading factor λ (λ - P_a curve); (b) reactive power (P_r) versus loading factor λ (λ - P_r curve).

Figure 8b presents the λ - P_r curve. It can be observed that, even with only 11 centers (radial basis functions), the RBF network achieved an exceptionally accurate approximation of the desired values, demonstrating its robustness and efficiency in system modeling. These results reinforce the RBF network's strong capability to capture the nonlinear behavior of power losses, showing slightly superior performance compared to the MLP and GRNN. This confirms its potential as a precise and computationally efficient approach for estimation problems in electrical power systems.

The results presented in Table 6 show that the RBF network achieved the minimum errors for both real and reactive power at the critical point, confirming its superior estimation

capability compared to the MLP and GRNN. On the other hand, the GRNN required the lowest CPU time, highlighting its computational efficiency despite slightly higher errors.

Table 6. Critical points estimated by the MLP, RBF, and GRNN, together with their corresponding errors relative to the desired output and the CPU time (s) for training and validation samples.

λ		Y_{des}	Y_{ob}		
			MLP	RBF	GRNN
1.76799	Pa	0.7855	0.7740	0.7869	0.7914
	Pr	3.1321	3.0700	3.1602	3.1666
Error					
	Pa	--	0.0115	* 0.0014	0.0059
	Pr	--	0.0621	* 0.0281	0.0345
CPU time (s) for training and validation					
			3	2	** 0.5

* minimum error for the RBF network; ** lowest CPU time.

It is important to emphasize that artificial neural networks, despite their powerful predictive capabilities, are often considered “black box” models, making it challenging to interpret the internal decision-making process. However, in power system stability analysis, accurate determination of critical points is crucial, as these values define the boundary between stable and unstable operating conditions. The critical loading factor (λ) and corresponding power values directly indicate the maximum loading capacity before voltage collapse occurs, providing essential information for system operators to maintain secure operation.

Table 6 was specifically designed to demonstrate the networks’ performance at this critical operating point, where computational time becomes a critical factor. Power system operators require real-time or near-real-time estimations to make informed decisions and implement preventive control actions before the system reaches instability. A delay of even a few seconds in obtaining critical point estimates could be the difference between successful intervention and system collapse. Therefore, while the GRNN exhibited slightly higher errors compared to the RBF network, its significantly reduced CPU time (0.4 s versus 2 s) makes it a highly attractive option for online stability assessment applications, where speed is paramount without compromising acceptable accuracy levels.

4. Conclusions

This study highlighted the effectiveness of an Artificial Neural Network (ANN)-based approach for estimating real and reactive power losses in electrical power systems. The comparison among MLP, RBF, and GRNN architectures, conducted using the same training and validation samples for all three networks, showed that the RBF network outperformed the others, even with a reduced number of radial basis functions, providing more accurate estimates, particularly in the critical regions of the system. The proposed methodology proved advantageous due to its rapid results and its ability to handle the system’s nonlinearity, eliminating the need for iterative procedures. The numerical results indicate that the average error for the MLP was 3.44×10^{-4} , while the RBF achieved 2.84×10^{-4} , and the GRNN achieved 6.045×10^{-4} . This confirms the superiority of the RBF in approximating the desired output, demonstrating its higher effectiveness in predicting power losses. The MLP exhibited intermediate performance in terms of MSE, whereas the GRNN, despite presenting a slightly higher error than both RBF and MLP, achieved a shorter processing time (0.5 s compared to 2 s for RBF and 3 s for MLP). For

the GRNN, a spread value of 0.4 was used, which enabled the network to meet the established MSE criterion of 0.001. In contrast, the RBF network employed the default spread of 1.0, showing stable convergence and satisfactory accuracy within a few iterations (11 centers). These findings reinforce that Artificial Neural Networks provide an efficient, accurate, and practical alternative for estimating losses in electrical power systems, offering fast and reliable solutions for system planning and operational applications.

Author Contributions: G.G.d.S.: writing—original draft, writing—review and editing, investigation, methodology; R.F.M.R.: data curation, visualization; M.A.S.: visualization, validation; N.L.D.: formal analysis, visualization; A.C.d.J.G.: validation, methodology; A.B.N.: writing—original draft, methodology, writing—review and editing, supervision. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by CAPES under grant number 88887.956029/2024-00.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: The authors would like to thank the financial support from São Paulo State University—UNESP—Prope and CNPq process 406060/2025-8.

Conflicts of Interest: The authors declare that they have no competing interests.

Abbreviations

ANN	Artificial neural network
MSE	Mean square error
MLP	Multilayer perceptron network
RBF	Radial Basis Function
GRNN	Generalized Regression Neural Network
Y_{ob}	Obtained outputs
Y_{des}	Desired outputs
λ	Loading factor
P_a	Total real power losses
P_r	Total reactive power losses

References

1. Moldoveanu, C.; Ionita, I.; Brezoianu, V.; Zaharescu, S.; Grigorescu, S.; Hategan, I.; Marcolt, M. Romanian innovative system for power losses real time monitoring and analysis in electrical transmission and distribution systems. In Proceedings of the 9th International Conference on Modern Power Systems (MPS), Cluj-Napoca, Romania, 16–17 June 2021; pp. 1–6. [\[CrossRef\]](#)
2. Pesántez, G.; Guamán, W.; Córdova, J.; Torres, M.; Benalcázar, P. Reinforcement Learning for Efficient Power Systems Planning: A Review of Operational and Expansion Strategies. *Energies* **2024**, *17*, 2167. [\[CrossRef\]](#)
3. Tostado-Véliz, M.; Kamel, S.; Jurado, F. Development and Comparison of Efficient Newton-Like Methods for Voltage Stability Assessment. *Electr. Power Compon. Syst.* **2020**, *48*, 1798–1813. [\[CrossRef\]](#)
4. Fernandes, T.R.; Ricciardi, T.R.; Schincariol, R.; De Almeida, M.C. Contributions to the Sequence-Decoupling Compensation Power Flow Method for Distribution System Analysis. *IET Gener. Transm. Distrib.* **2019**, *13*, 583–594. [\[CrossRef\]](#)
5. Li, X.; Zhao, J.; Liang, H.; Xu, J. Application of an Improved Continuous Power Flow Method in Voltage Stability Analysis. In Proceedings of the 6th International Conference on Systems and Informatics (ICSAI), Shanghai, China, 2–4 November 2019; pp. 244–248. [\[CrossRef\]](#)
6. dos Santos, W.P.L.; Bonini Neto, A.; Gabriel Filho, L.R.A. Post-Contingency Loading Margin through Plane Change in the Continuation Power Flow. *Energies* **2023**, *16*, 7583. [\[CrossRef\]](#)
7. Kaseb, Z.; Möller, M.; Vergara, P.P.; Palensky, P. Power flow analysis using quantum and digital annealers: A discrete combinatorial optimization approach. *Sci. Rep.* **2024**, *14*, 23216. [\[CrossRef\]](#) [\[PubMed\]](#)
8. Bonini Neto, A.; De Queiroz, A.; Da Silva, G.G.; Gifalli, A.; De Souza, A.N.; Garbelini, E. Predictive Modeling of Total Real and Reactive Power Losses in Contingency Systems Using Function-Fitting Neural Networks with Graphical User Interface. *Technologies* **2025**, *13*, 15. [\[CrossRef\]](#)

9. da Silva, G.G.; De Queiroz, A.; Garbelini, E.; Dos Santos, W.P.L.; Minussi, C.R.; Bonini Neto, A. Estimation of Total Real and Reactive Power Losses in Electrical Power Systems via Artificial Neural Network. *Appl. Syst. Innov.* **2024**, *7*, 46. [CrossRef]
10. Bonini Neto, A.; Alves, D.A.; Minussi, C.R. Artificial Neural Networks: Multilayer Perceptron and Radial Basis to Obtain Post-Contingency Loading Margin in Electrical Power Systems. *Energies* **2022**, *15*, 7939. [CrossRef]
11. Carreras, J.; Kikuti, Y.Y.; Miyaoka, M.; Hiraiwa, S.; Tomita, S.; Ikoma, H.; Kondo, Y.; Ito, A.; Nakamura, N.; Hamoudi, R. A Combination of Multilayer Perceptron, Radial Basis Function Artificial Neural Networks and Machine Learning Image Segmentation for the Dimension Reduction and the Prognosis Assessment of Diffuse Large B-Cell Lymphoma. *AI* **2021**, *2*, 106–134. [CrossRef]
12. Ke, K.-C.; Huang, M.-S. Quality prediction for injection molding by using a multilayer perceptron neural network. *Polymers* **2020**, *12*, 1812. [CrossRef] [PubMed]
13. Aydin, F.; Gümüş, B. Study of Different ANN Algorithms for Voltage Stability Analysis. In Proceedings of the Innovations in Intelligent Systems and Applications Conference (ASYU), Istanbul, Turkey, 15–17 October 2020; pp. 1–5. [CrossRef]
14. Al-Mahasneh, A.J.; Anavatti, S.; Garratt, M.; Pratama, M. Applications of General Regression Neural Networks in Dynamic Systems. In *Digital Systems*; IntechOpen: London, UK, 2018. [CrossRef]
15. Alzo'ubi, A.K.; Ibrahim, F. Predicting the Pile Static Load Test Using Backpropagation Neural Network and Generalized Regression Neural Network—A Comparative Study. *Int. J. Geotech. Eng.* **2018**, *15*, 810–821. [CrossRef]
16. Badrudeen, T.U.; Nwulu, N.I.; Gbadamosi, S.L. Neural Network Based Approach for Steady-State Stability Assessment of Power Systems. *Sustainability* **2023**, *15*, 1667. [CrossRef]
17. Gifalli, A.; Bonini Neto, A.; de Souza, A.N.; de Mello, R.P.; Ikeshoji, M.A.; Garbelini, E.; Neto, F.T. Fault Detection and Normal Operating Condition in Power Transformers via Pattern Recognition Artificial Neural Network. *Appl. Syst. Innov.* **2024**, *7*, 41. [CrossRef]
18. MathWorks. Available online: <https://www.mathworks.com/> (accessed on 10 January 2025).
19. Otaru, A.J.; Alhulaybi, Z.A.; Dubdub, I. Machine Learning Backpropagation Prediction and Analysis of the Thermal Degradation of Poly (Vinyl Alcohol). *Polymers* **2024**, *16*, 437. [CrossRef] [PubMed]
20. Hammad, M.M. Comprehensive Survey of Complex-Valued Neural Networks: Insights into Backpropagation and Activation Functions. *arXiv* **2024**. [CrossRef]
21. Yang, Y.; Wang, P.; Gao, X. A Novel Radial Basis Function Neural Network with High Generalization Performance for Nonlinear Process Modelling. *Processes* **2022**, *10*, 140. [CrossRef]
22. Lin, C.L.; Wang, J.F.; Chen, C.Y.; Chen, C.W.; Yen, C.W. Improving the generalization performance of RBF neural networks using a linear regression technique. *Expert. Syst. Appl.* **2009**, *36*, 12049–12053. [CrossRef]
23. Al-Mahasneh, A.J.; Anavatti, S.G.; Garratt, M.A. Review of Applications of Generalized Regression Neural Networks in Identification and Control of Dynamic Systems. *arXiv* **2018**, arXiv:1805.11236. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.