

Article

Enhancing Protection Coordination Robustness in DER-Rich Grids Through Deep Learning-Based Preventive Relay Setting Calibration

Jheng-Lun Jiang ¹, Tung-Sheng Zhan ^{2,*}  and Jun-Jie Chi ²

¹ Department of Electrical Engineering and Information Technology, National Atomic Research Institute, No. 1000, Wenhua Rd. Jiaan Village, Longtan District, Taoyuan City 32546, Taiwan; jhenglun@nari.org.tw

² Department of Electrical Engineering, National Kaohsiung University of Science and Technology, No. 415, Jiangong Rd., Sanmin Dist., Kaohsiung City 80778, Taiwan; f113154112@nkust.edu.tw

* Correspondence: tszhan1109@nkust.edu.tw; Tel.: +886-73814526 (ext. 15509); Fax: +886-73921073

Highlights

Please indicate how your work links to systems science via your contributions to systems practice, theory, and/or methodology.

- This work contributes to systems practice by formulating protection coordination in DER-rich distribution networks as an integrated offline–online preventive relay-setting calibration problem, linking power system operation, protection coordination, and data-driven decision support.
- From a methodological perspective, the study develops a system-level framework that combines offline heuristic optimization-based label generation with CNN-based surrogate inference, enabling rapid recommendation of complete TMS and PCS setting vectors from structured relay–fault-location fault-current signatures.

What are the main findings and/or the implications of the main findings?

- The results show that the proposed CNN-based framework can restore primary–backup relay coordination and maintain acceptable CTI margins under varying DER-generation and load-demand conditions in both the 16-bus and IEEE 37-bus test systems.
- The findings indicate that data-driven surrogate models can support faster preventive protection coordination in active distribution systems, while offline optimization, setting-boundary checks, CTI verification, relay-sequence confirmation, and operator or engineering approval remain necessary for safe implementation.

Abstract

The increasing penetration of distributed energy resources (DERs) introduces significant operational uncertainties, challenging the reliability and robustness of conventional protection coordination in distribution networks. Traditional optimization methods can obtain high-quality relay settings, but their iterative computational burden limits their direct use for real-time adaptation. To address this issue, this paper proposes a deep learning-based preventive relay setting calibration framework for enhancing protection coordination robustness in DER-rich distribution networks. The proposed method adopts an offline–online architecture. In the offline stage, a refined heuristic algorithm is integrated with ETAP-based fault analysis to generate a comprehensive dataset of high-quality optimized time-multiplier settings (TMSs) and pickup current settings (PCSs) under a wide range of DER-generation and load-demand scenarios. Subsequently, a convolutional neural network (CNN) is trained to learn a nonlinear mapping from multidimensional fault-current



Academic Editor: Yutaka Matsubara

Received: 29 June 2026

Revised: 8 August 2026

Accepted: 10 August 2026

Published: 14 August 2026

Copyright: © 2026 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.

signatures to the corresponding optimized relay-setting vectors. In the online stage, the trained CNN serves as a predictive surrogate model, rapidly recommending coordinated relay settings for the current operating condition. The framework is validated using a 16-bus distribution system and the IEEE 37-bus test feeder. The results show that the proposed method can restore correct primary–backup relay operating sequences and maintain coordination time intervals (CTIs) within the required 0.2–0.4 s range under the studied DER-rich operating scenarios. This CNN-based preventive calibration approach provides a rapid, adaptive decision-support tool to improve protection coordination robustness against DER-induced operating uncertainties.

Keywords: protection coordination; time multiplier setting; pickup current setting; convolutional neural network; deep learning

1. Introduction

The global energy landscape is undergoing a profound transformation, driven by the increasing integration of Distributed Energy Resources (DERs) to meet decarbonization goals and enhance energy autonomy [1]. This paradigm shift, however, fundamentally alters the passive, unidirectional nature of traditional distribution networks, turning them into active, complex systems. The discipline of power system relaying has historically centered on centralized generation, allowing for the meticulous calibration of protective schemes within a predictable, hierarchical framework. Critically, the inherent intermittency and bidirectional power flow capabilities of DERs introduce significant operational uncertainties that challenge these foundational principles [2]. Consequently, legacy overcurrent protection schemes, designed for predictable fault conditions, are often ill-equipped to handle the dynamic environment of modern, DER-rich grids, leading to compromised reliability and security [3,4].

The integration of DERs introduces several specific challenges that render traditional relay coordination strategies ineffective. Firstly, bidirectional power flow, often occurring when DERs are integrated mid-feeder, reverses the current direction during faults, which can blind conventional protection devices designed for unidirectional current [2,3]. Secondly, fault current variability becomes a major issue; the contribution from DERs can either increase fault magnitudes beyond switchgear ratings or, conversely, reduce the current seen by upstream relays below their pickup thresholds, leading to protection failure [4]. This variability, compounded by the inherent intermittency of renewable sources like solar and wind, makes it exceedingly difficult to maintain a reliable set of relay parameters. The culmination of these issues is a systemic loss of protection coordination, in which the selective, swift isolation of faults can no longer be guaranteed, necessitating the development of new, adaptive coordination strategies to maintain grid stability and safety [2–4].

To address these challenges, the discipline is increasingly focused on developing innovative adaptive relay coordination schemes that dynamically adjust settings in response to changing grid conditions [5]. Reference [6] introduces the Protection Coordination Index (PCI) for DER-integrated meshed networks, which is solved via a two-stage nonlinear programming method. A prominent research avenue involves applying sophisticated heuristic optimization algorithms (HOAs) to obtain high-quality relay settings. A wide array of methods, ranging from classic Genetic Algorithms (GAs) and Particle Swarm Optimization (PSO) to more advanced techniques such as the Grey Wolf Optimizer (GWO) and the Firefly Algorithm (FA), have been extensively analyzed and proven effective in solving the overcurrent relay coordination problem [7]. A modified particle swarm optimization

(MPSO) algorithm and a refined immune algorithm (RIA) represent a state-of-the-art example in this domain, demonstrating the ability to find high-quality, optimized solutions [8,9]. However, a fundamental limitation shared by all heuristic methods is their iterative nature, which imposes a significant computational burden. This high computational cost, while acceptable for offline planning, limits their direct use in time-sensitive online applications that require rapid responses to dynamic grid conditions.

To overcome the computational limitations of conventional optimization methods, a complementary research direction based on artificial intelligence (AI) has emerged, aiming to develop fast predictive models from data generated offline. Early studies primarily employed artificial neural networks (ANNs), including multilayer perceptrons trained using the Levenberg–Marquardt backpropagation algorithm, as high-speed surrogate models. These ANN-based methods have been applied to predict relay operating times [10], model the operating characteristics of directional overcurrent relays (DOCRs) [11], and determine adaptive relay settings for both radial and looped distribution systems [12]. ANN techniques have also been investigated for other protection-related tasks, including centralized fault detection and location [13] and real-time anti-islanding protection, with some studies progressing to hardware-in-the-loop validation [14].

More recently, advances in deep learning have enabled the development of data-driven protection schemes capable of extracting more complex fault and system-state features. Convolutional neural networks (CNNs) have demonstrated considerable potential for fault diagnosis, classification, and recognition, particularly in power systems with high penetrations of inverter-based resources, where fault signatures may differ substantially from those of conventional synchronous-generation systems [15,16]. In addition to CNN-based approaches, other deep learning architectures have been introduced for protection applications. Reference [17] proposed a hybrid deep reinforcement learning-based protection scheme that combines centralized single-agent protection with decentralized multi-agent backup protection, allowing protection actions to be learned directly from simulated system interactions. Reference [18] developed a hybrid PSO–DNN framework to improve directional overcurrent relay coordination in AC microgrids under high-impedance faults, in which PSO was used to optimize relay settings and DNN hyperparameters. At the same time, the trained DNN provided adaptive relay settings under varying fault conditions. Reference [19] proposed an LSTM-based method for fault-direction estimation and communication-assisted protection coordination in networked distribution systems. These studies demonstrate that deep learning can support fault identification, direct protection-action selection, and adaptive relay-setting determination under increasingly complex system conditions.

Despite these advances, several research gaps remain. First, many existing AI-based protection methods focus primarily on fault classification, fault-direction estimation, trip/no-trip decision-making, or selection among predefined relay setting groups. These approaches do not directly predict the complete set of coordination parameters for all primary and backup relays. Second, although some recent studies have considered adaptive relay settings, their applications are often restricted to specific fault types, network configurations, or operating conditions, such as high-impedance faults in microgrids. Third, many existing models are validated using a limited number of discrete operating states, which may not adequately capture the continuous variation in DER generation and load demand observed in practical distribution networks. Finally, relatively few studies evaluate both the numerical accuracy of the predicted relay settings and their physical coordination performance in terms of primary–backup operating sequence and CTI constraints.

To address these gaps, this paper proposes a CNN-based framework for calibrating preventive relay settings to enhance the robustness of protection coordination in DER-rich

distribution networks. The proposed method adopts an offline–online architecture that combines the solution quality of heuristic optimization with the rapid inference capability of deep learning. In the offline stage, the refined immune algorithm generates a large, diverse dataset of high-quality, optimized relay settings, including both TMSs and PCSs, across a wide range of DER-generation and load-demand scenarios. The resulting fault-current patterns and optimized relay settings are then used to train a lightweight CNN. In the online stage, the trained CNN maps the system’s multidimensional fault-current signature directly to the complete TMS and PCS vectors of all coordinated relays. The predicted settings are subsequently verified using primary–backup relay operating sequences and CTI constraints, thereby enabling proactive relay setting adaptation before a fault occurs.

To further clarify the positioning of the proposed framework relative to existing AI-based protection studies, Table 1 summarizes representative ANN-, DNN-, LSTM-, DRL-, and CNN-based methods with respect to learning objective, input features, output parameters; relay-setting prediction scope; CTI verification; and online decision capability. Existing studies have demonstrated the effectiveness of AI for relay operating-time estimation, fault detection, fault-direction identification, trip/no-trip decision-making, and adaptive setting selection. However, relatively few studies directly predict the complete TMS and PCS vectors for all coordinated relays from multidimensional fault-current signatures while also verifying the primary–backup operating sequence and CTI constraints after prediction.

The main contributions of this study are summarized as follows:

- A preventive relay setting calibration framework: An offline–online framework is proposed to support proactive relay setting adaptation under changing DER and load conditions. Instead of repeatedly executing time-consuming heuristic optimization during online operation, the proposed framework uses offline-optimized setting data to train a fast predictive model for preventive calibration.
- A CNN-based surrogate model for complete TMS/PCS prediction: A lightweight CNN-based regression model is developed to simultaneously predict the optimized TMS and PCS values of all coordinated relays. Compared with AI-based methods that focus mainly on fault classification, trip decisions, fault-direction estimation, or partial setting selection, the proposed model directly generates complete relay-setting vectors for protection coordination.
- Validation using two distribution systems and physical coordination criteria: The proposed framework is evaluated using a 16-bus distribution system and the IEEE 37-bus test feeder under diverse DER-generation and load-demand scenarios. Its performance is assessed not only by prediction accuracy, but also by protection coordination criteria, including primary–backup relay operating sequences and CTI constraints.

A preliminary version of this work was presented in the conference paper [20]. Compared with the preliminary conference version, the present journal article provides substantial extensions in both methodology and validation. First, the scope of the study is expanded from a CNN-based protection coordination decision-support system to a deep learning-based preventive relay setting calibration framework with a more clearly defined offline–online architecture. Second, the dataset generation procedure is substantially expanded and clarified by considering wider DER-generation and load-demand variations, feasibility screening, and scenario-level data partitioning. Third, the validation is extended from the original 16-bus test system to include both the 16-bus distribution system and the IEEE 37-bus test feeder. Fourth, additional analyses are provided, including sensitivity tests, CNN–MLP baseline comparison, and computation-time analysis distinguishing offline optimization-based label generation from online CNN inference. These extensions provide a more comprehensive assessment of the proposed framework in terms of prediction accuracy, protection coordination performance, robustness, and practical decision-support applicability.

Table 1. Comparison of representative AI-based protection coordination studies.

Method/Reference	Main Task	Input Features	Output Parameters or Decision	Relay-Setting Prediction Scope	CTI/Relay-Sequence Verification
ANN-based relay models [10–12]	Relay operating-time prediction, DOCR characteristic modeling, or adaptive setting estimation	Current, relay, or system operating variables	Relay operating time or selected setting parameters	Partial; usually limited to specific relay parameters or operating cases	Limited or case-dependent
ANN-based protection applications [13,14]	Fault detection, fault location, or anti-islanding protection	Voltage/current measurements and system signals	Fault class, location, or islanding decision	Not focused on complete TMS/PCS prediction	Not the main focus
CNN-based fault diagnosis [15,16]	Fault diagnosis, classification, and recognition	Fault signatures, waveform features, or image-like feature maps	Fault type, location, or class label	Not focused on relay-setting prediction	Not the main focus
DRL-based hybrid protection [17]	Direct protection-action learning	Local measurements such as voltage, current, reactance, and breaker status	Trip/no-trip action	No direct TMS/PCS prediction	Indirect through learned protection actions
PSO–DNN relay coordination [18]	Adaptive DOCR coordination under high-impedance faults	Fault and system-condition data	Adaptive relay settings	Focused on AC microgrid and HIF conditions	Considered for DOCR coordination
LSTM-based protection coordination [19]	Fault-direction estimation and communication-assisted protection coordination	Fault-current waveform time-series data	Fault direction and faulty section information	No direct TMS/PCS prediction	Supports protection coordination indirectly
Proposed CNN framework	Preventive relay setting calibration	Multidimensional fault-current signature under DER/load scenarios	Complete TMS and PCS vectors for all coordinated relays	Full setting-vector prediction for all coordinated relays	Verified using primary–backup operating sequence and CTI constraints

2. System Architecture of the Proposed CNN-Based Decision Support System

The proposed framework for enhancing protection coordination robustness is built upon a synergistic, two-stage architecture: an offline stage for intensive data generation and model training, and an online stage for real-time decision support and preventive calibration. This architecture is designed to decouple the computationally expensive optimization process from the time-critical operational phase, thereby combining the solution quality of heuristic algorithms with the rapid inference capability of deep learning. Figure 1 illustrates the overall workflow of this system.

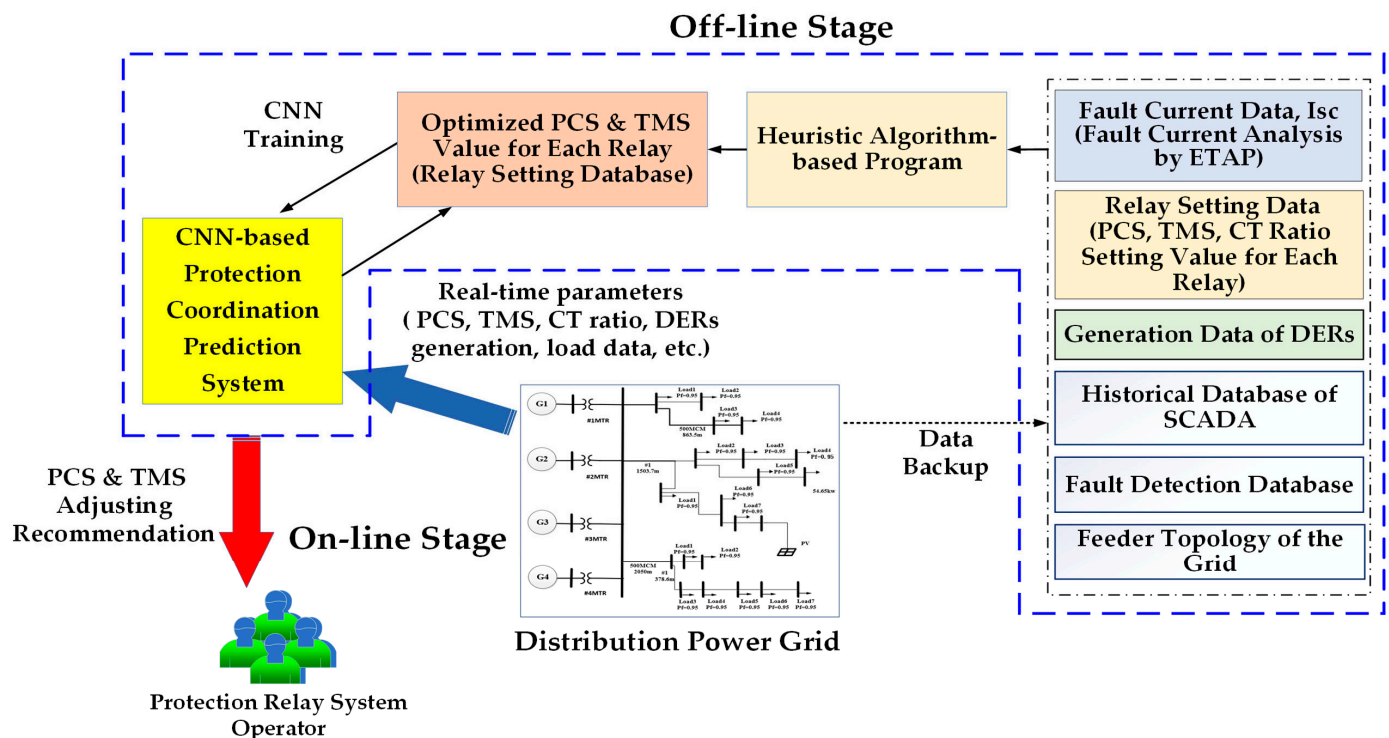


Figure 1. Framework of the proposed CNN-based protection coordination decision system.

2.1. Offline Stage: Data Generation and Model Training

The primary objective of the offline stage is to generate a high-quality, diverse dataset and use it to train a robust CNN model. This process involves two key steps:

(1) Best-Found Solution Dataset Generation:

A comprehensive set of operating scenarios was generated by varying DER penetration levels and load demand conditions. For each scenario, the corresponding power system model was simulated in ETAP to obtain fault currents at all relevant fault locations and relays. These fault-current values were then used as inputs to the refined immune algorithm (RIA), which solves the relay coordination optimization problem subject to CTI and parameter boundary constraints. Because RIA is a heuristic optimization method, the resulting relay settings are referred to in this study as high-quality, rather than globally optimal, solutions. This process was repeated for all predefined scenarios, resulting in a dataset in which each sample consists of a system operating condition, its corresponding fault-current signature, and the optimized TMS/PCS vector.

(2) CNN Model Training:

The generated dataset was then used to train a convolutional neural network. The fault-current values were organized into an $R \times M$ fault-current matrix as input to the

CNN, where R denotes the number of coordinated relays, and M denotes the number of predefined fault locations. The corresponding optimized TMS and PCS values formed the output reference vector. The CNN architecture was designed to learn a nonlinear mapping from the input fault-current signatures to the optimized relay-setting targets. The dataset was partitioned into training, validation, and testing sets to evaluate the model's generalization capability and reduce overfitting.

2.2. Online Stage: Preventive Calibration and Decision Support

In the online stage, the fully trained and validated CNN model is deployed as a real-time decision support tool. Its function is to provide preventive calibration for the protection system. The workflow is as follows:

(1) Online Data Acquisition:

The system continuously monitors key grid parameters, such as DER output levels and load demands, through a SCADA system.

(2) Fast Surrogate Inference:

When a significant change in the system's operating state is detected (e.g., a large cloud passing over a solar farm or a major industrial load coming online), the relevant system data is fed into the trained CNN model.

(3) Proactive Setting Recommendation:

The trained CNN model rapidly outputs recommended TMS and PCS values corresponding to the current grid condition.

(4) Decision Support and Setting Verification:

The recommended settings are not intended to be written directly to protection devices without verification. Before implementation, the predicted TMS and PCS values should pass setting-boundary checks, primary-backup relay sequence checks, and CTI verification. The verified settings can then be presented to the system operator for confirmation or transferred to intelligent electronic devices through a secure engineering procedure. Therefore, the proposed framework is positioned as a preventive decision-support tool for relay setting adaptation rather than an unrestricted automatic setting-update mechanism.

3. Methodologies

3.1. Overcurrent Relay Coordination Problem Formulation

In this paper, the relay protection coordination problem is formulated as an optimization issue, a methodological approach also adopted by the authors in [8,9]. The objective is to minimize the total cumulative operating time (COT) of all relays across all pre-scheduled tracking routes in the system. Since the operating time of each relay depends on its PCS and TMS, these two parameters are treated as decision variables in the optimization process. The proposed method aims to achieve effective protection coordination by appropriately adjusting PCS and TMS while ensuring rapid and selective fault clearance. This optimization problem can be mathematically formulated as follows:

Objective function:

$$\text{Minimize } COT = \sum_{k=1}^{TP} \sum_{j=1}^{FN} \sum_{i=1}^{RN} \frac{\beta_i}{\left(\frac{I_{f,i,j}^k}{PCS_i \cdot CT_i} \right)^{\alpha_i} - 1} \times TMS_i \quad (1)$$

Subject to

$$t_{i_b,j}^k - t_{i_p,j}^k \geq CTI_{x,j}^k \quad (2)$$

$$CTI_{x,j}^k \text{--}min \leq CTI_{x,j}^k \leq CTI_{x,j}^k \text{--}max \quad (3)$$

$$PCS_i \text{--}min \leq PCS_i \leq PCS_i \text{--}max \quad (4)$$

$$TMS_i \text{--}min \leq TMS_i \leq TMS_i \text{--}max \quad (5)$$

where

COT: Object function of calculating the COT for all specified tracking routes.

i : The relay identifier.

j : The fault location identifier.

k : The coordination tracking route identifier (relay connectivity).

FN : The total number of fault locations is considered.

TP : The total number of specific tracking routes.

RN : The total relay number of the set of tracking routes.

TMS_i : TMS parameter of relay i .

PCS_i : PCS parameter of relay i .

$If_{i,j}^k$: Fault current through relay i when the j th fault occurs in the k th coordination tracking route.

α_i, β_i : Parameters of the standard normal time-current characteristic function [21] for relay i .

CT_i : CT ratio corresponds to relay i .

$CTI_{x,j}^k$: CTI of the x th pair of primary/backup relay when the j th fault occurs in the k th coordination tracking route.

$t_{i_p,j}^k$: OT of primary relay i_p when the j th fault occurs in the k th coordination tracking route.

$t_{i_b,j}^k$: OT of backup relay i_b when the j th fault occurs in the k th coordination tracking route.

$CTI \text{--}min$: The lower CTI bound is set to 0.2 s.

$CTI \text{--}max$: The upper CTI bound is set to 0.4 s.

$TMS \text{--}min$: The lower bound of TMS is set to 0.05.

$TMS \text{--}max$: The upper bound of TMS is set to 1.0.

$PCS \text{--}min$: The lower bound of PCS is set to 0.05.

$PCS \text{--}max$: The upper bound of PCS is set to 5.0.

Effective protection hinges on precisely setting the two key decision variables, TMS and PCS. These parameters collectively define the Time-Current Characteristic (TCC) of the relay, which governs the inverse relationship between fault current magnitude and operating time, as mathematically expressed in Equation (1) and standardized in IEC 60255-3 [21]. For the offline data generation in this work, we adopt the specific optimization model detailed in our previous studies [8,9]. This model conceptualizes the coordination problem along predefined tracking routes, which represent primary/backup relay sequences based on the network topology. Within this framework, the primary objective is to minimize the COT for all relays along each route, subject to the CTI and parameter boundary constraints. While the complete theoretical foundation and solution methodology for the heuristic optimization are elaborated in [8,9], the primary focus of the present study is the subsequent stage: leveraging the dataset generated by this robust offline process to train a CNN for preventive calibration of protection parameters.

3.2. CNN Model Architecture and Training

The optimization formulation described in Section 3.1 is used in this study to generate supervised learning targets during the offline stage. The main objective of this section is not to repeat the detailed procedures of MPSO or RIA, which have been reported in previous studies [8,9], but to describe how the optimized relay-setting dataset is transformed into a CNN-based predictive model for preventive relay setting calibration. In this framework, the offline heuristic optimization process provides high-quality optimized TMS and PCS

values. At the same time, the CNN learns the nonlinear relationship between system fault-current signatures and the corresponding relay-setting vectors. In the online stage, the trained CNN model serves as a fast surrogate to provide relay-setting recommendations without repeatedly running time-consuming optimization.

For each operating scenario, the model receives an $R \times M$ fault-current matrix as input, where R represents the number of coordinated relays, and M represents the number of predefined fault locations. The DER output and load-demand conditions are reflected in the resulting fault-current matrix and are treated as scenario-defining variables rather than as an additional input dimension. As illustrated in Figure 2, the CNN processes the structured fault-current matrix through convolutional layers to extract local correlation patterns among fault locations and relay measurements. Pooling layers are then used to reduce feature dimensionality while retaining representative features. The extracted features are flattened and passed through fully connected layers to establish the nonlinear mapping between the fault-current signature and the relay-setting parameters. Finally, the output layer generates the recommended TMS and PCS values for all coordinated relays.

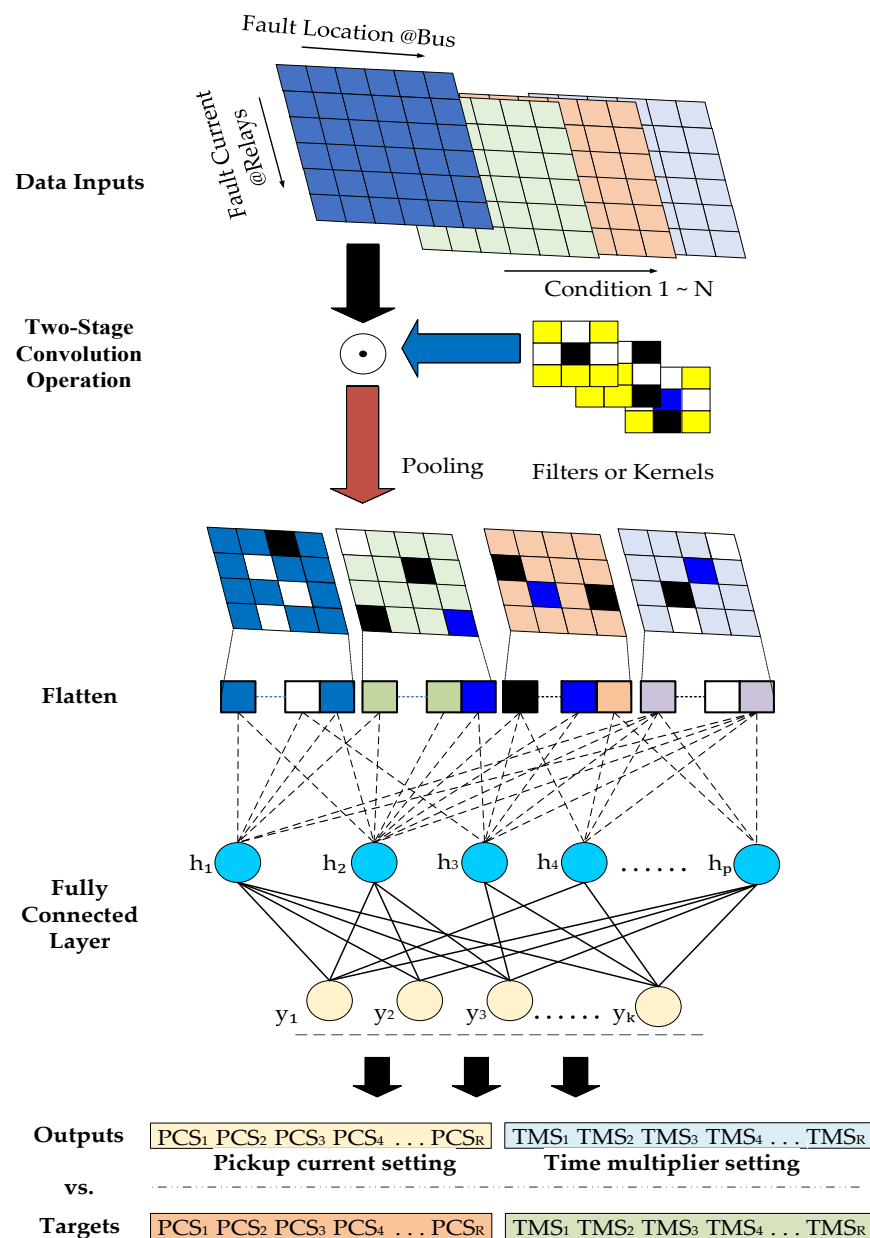


Figure 2. CNN training procedure.

A CNN architecture was selected because the fault currents observed by different relays under different fault locations are not independent scalar quantities. Instead, they form a structured pattern associated with feeder topology, DER penetration, load variation, and primary–backup relay relationships. The convolutional kernels are well suited for extracting local, spatially correlated patterns from the fault-current matrix. The following subsections describe the input data structuring, CNN architecture, and training protocol used in the proposed preventive calibration framework.

3.2.1. Input Data Structuring

For each operating scenario, the fault-current data are organized into a two-dimensional matrix. Let M denote the number of predefined fault locations, and R denote the number of coordinated relays. The CNN input is therefore represented as an $R \times M$ fault-current matrix, where each element corresponds to the fault current measured by a specific relay at a specific fault location. This matrix is referred to as the fault-current signature of the corresponding operating scenario.

The DER output levels and load-demand levels are not treated as an additional matrix dimension in this study. Instead, they are used to define each operating scenario and to generate the corresponding fault-current signature through ETAP-based fault analysis. In other words, each DER/load condition corresponds to one independent sample, and the input of that sample remains an $R \times M$ fault-current matrix. The optimized TMS and PCS values obtained from the offline heuristic optimization process are used as the target output vector for that sample.

This representation is consistent with the CNN training procedure shown in Figure 2, where the fault-current matrix is constructed according to fault locations and relays. By arranging the fault-current responses in a structured matrix form, the CNN can extract local correlation patterns among fault locations and relay measurements. The output target of each sample is a $(2R)$ -dimensional vector containing the optimized TMS and PCS values for all coordinated relays.

3.2.2. CNN Architecture

The proposed CNN model architecture, as illustrated in Figure 2, consists of two convolutional blocks followed by two fully connected layers. This design balances feature extraction capability with computational efficiency. The detailed parameters of the architecture are as follows:

1. Input Layer

The input layer accepts the $R \times M$ fault-current matrix of each operating scenario, where M is the number of predefined fault locations and R is the number of coordinated relays. Each column corresponds to single fault location, and each row corresponds to a relay measurement. The DER generation and load-demand conditions are used to form different operating scenarios and are reflected in the resulting fault-current matrices; they are not represented as an additional tensor dimension. Therefore, the scenario index is treated as the sample index in the dataset.

2. Convolution Layer 1 and 2

Feature extraction is achieved using convolutional kernels. The first convolution layer employs 32 kernels of size 3×3 to capture low-level local features, and layer 2 uses 64 kernels with the same kernel dimension to learn more abstract, higher-level feature combinations, again followed by a Rectified Linear Unit (ReLU) activation function. The ReLU

is used as the activation function to introduce non-linearity. The convolution operation is formulated as:

$$f_{i,j}^k = \text{ReLU} \left(\sum_{p=1}^H \sum_{q=1}^N W_{pq}^k \cdot X_{(i+p,j+q)} + b^k \right) \quad (6)$$

where W_{pq}^k and b^k represent the kernel weights and bias of the k -th filter, and it is one of the most widely used activation functions in deep learning. Mathematically, it is defined as:

$$\text{ReLU}(x) = \begin{cases} 0, & x \leq 0 \\ x, & x > 0 \end{cases} \quad (7)$$

3. Pooling Layer 1 and 2

Dimensionality reduction in these two layers was performed via pooling operations, defined as:

$$\tilde{f}_{i,j}^k = \text{Max.} \left(f_{i+p,j+q}^k \mid p, q \in \Omega \right) \quad (8)$$

where Ω denotes the pooling region. A 2×2 pooling window was applied to down-sample the feature map, retaining the most salient features while reducing computational complexity.

4. Flatten Layer

The two-dimensional pooled feature maps are converted into a one-dimensional vector. This transformation allows the feature maps to be connected to fully connected layers.

$$z = \text{Flatten}(\tilde{f}) \quad (9)$$

where z is a one-dimensional vector.

5. Fully Connected Layer

The flattened vector is propagated through dense layers, represented as:

$$h^l = \text{ReLU} \left(W^l \cdot h^{l-1} + b^l \right), \quad l = 1, 2, \dots, L \quad (10)$$

where $h^0 = z$, and W^l and b^l are the weights and bias of the l -th connection layer. These layers learn higher-level nonlinear feature representations. The fully connected layer consists of 128 neurons with ReLU activation, which is responsible for integrating the extracted features to facilitate high-level reasoning.

6. Output Layer

The final layer has $2 \times R$ neurons, corresponding to the predicted TMS and PCS values for all R relays. As this is a regression task, no activation function is applied to this layer. The final output produces the recommended protection coordination parameters, including TMSs and PCSs for each relay, as shown on the left side of Figure 1.

$$Y = [TMS_1, TMS_2, \dots, TMS_{RN}, PCS_1, PCS_2, \dots, PCS_{RN}] \quad (11)$$

where RN represents the total number of relays.

3.2.3. Model Training and Hyperparameters

This CNN model is trained using a supervised learning approach. As shown in Figure 2, the training process aims to minimize the discrepancy between the CNN-predicted settings and the offline-optimized reference settings generated by RIA or MPSO. The mean

squared error (MSE) was selected as the loss function to quantify this discrepancy and guide the network weight updates during training. The MSE between the predicted relay settings and the offline-optimized reference targets is defined as:

$$L = \frac{1}{T} \sum_{i=1}^T \|Y_i - \hat{Y}_i\|^2 \quad (12)$$

The model training was implemented using the Adam optimizer, a highly effective and widely used algorithm for deep learning applications. The sample dataset was partitioned into a training set (80% samples), a validation set (10% samples), and a final test set (10% samples). The model was trained for a maximum of 200 epochs with a batch size of 32. To prevent overfitting and ensure generalization, an Early Stopping mechanism was implemented, which monitors the validation loss and terminates training if no improvement is observed for 10 consecutive epochs, saving the model with the best performance.

3.2.4. Implementation Workflow of the Offline Stage

To synthesize the methodologies described in Sections 3.1 and 3.2, Figure 3 provides a detailed, step-by-step flowchart of the entire offline implementation. This workflow guides the process from initial scenario setup to the final generation of the trained CNN model. The sequential steps of this workflow are as follows:

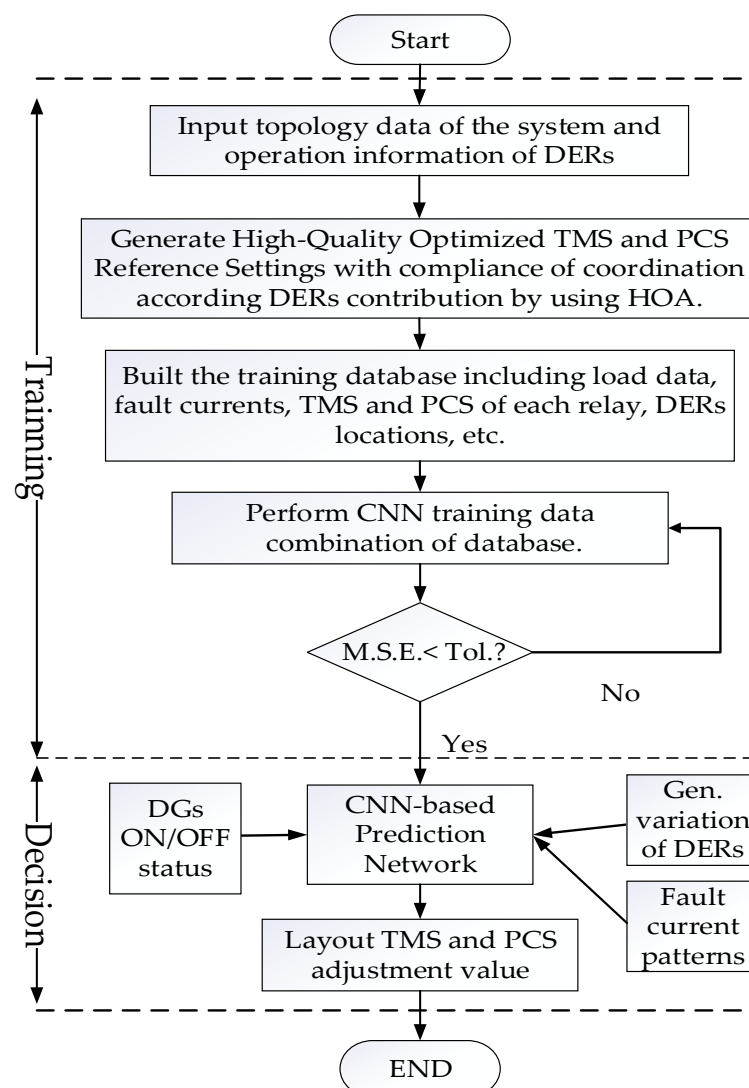


Figure 3. Offline implementation workflow for dataset generation and CNN training.

Step 1: System Initialization

The process begins by initializing core system parameters and generating the complete set of total unique operating scenarios, which represent a wide spectrum of DER penetration and load levels.

Step 2: Data Generation Loop

A loop is initiated to process each scenario. For each iteration:

Sub-step 2a: The operational data for the current scenario is loaded into ETAP to perform a comprehensive fault analysis.

Sub-step 2b: The fault-current data resulting from the ETAP simulation is extracted and loaded.

Sub-step 2c: The fault-current data are fed into the HOA, such as RIA or MPSO, to generate the optimized TMS and PCS reference vector used as the supervised learning target.

Sub-step 2d: Both the input features (fault currents) and the target labels (TMSs/PCSs) for the scenario are saved to the dataset.

Sub-step 2e: The process checks if all scenarios are complete. If not, the loop counter is incremented, and the process repeats for the next scenario.

Step 3: Training Data Preparation

Once the data generation loop is complete, the whole sample dataset is loaded and prepared. This step involves structuring the data and splitting it into the training, validation, and test sets (as defined in Section 3.2.3).

Step 4: Model Compilation

The CNN model, with the architecture specified in Section 3.2.2, is formally built and compiled with the Adam optimizer and MSE loss function.

Step 5: Model Training Loop

The training process begins.

Sub-step 5a: The CNN model is trained on the training data (e.g., for one epoch).

Sub-step 5b: The system then checks if the stopping criteria (e.g., convergence of validation loss or activation of Early Stopping, as detailed in Section 3.2.3) have been met.

Sub-step 5c: If the criteria are not met, the training loop continues. If they are met, the loop terminates.

Step 6: Finalization

The best-performing trained CNN model is saved for subsequent use in the online preventive calibration and decision-support stage described in Section 2.2.

4. Simulation Results

To validate the performance, robustness, and scalability of the proposed CNN-based preventive calibration framework, two distinct distribution test systems were modeled. This chapter details the experimental setup for both systems, followed by a comprehensive analysis of the simulation results.

4.1. Case Studies and Experimental Setup

4.1.1. Test System 1: 16-Bus System

The first case study utilizes a 16-bus, 22.8 kV actual distribution network, supplied by a 50 MVA, 161/22.8 kV main transformer. The system topology, which includes six Distributed Generators (DGs), is illustrated in Figure 4. The rated capacities of these DGs and specific locations are detailed in Table 2. To simulate diverse operational conditions and generate the training dataset, four distinct DER penetration rates (PRs) were defined, representing power outputs of 10%, 30%, 50%, and 70% of the DGs' rated capacities. Table 3 shows the load bus location and power demand specification of the 16-bus system.

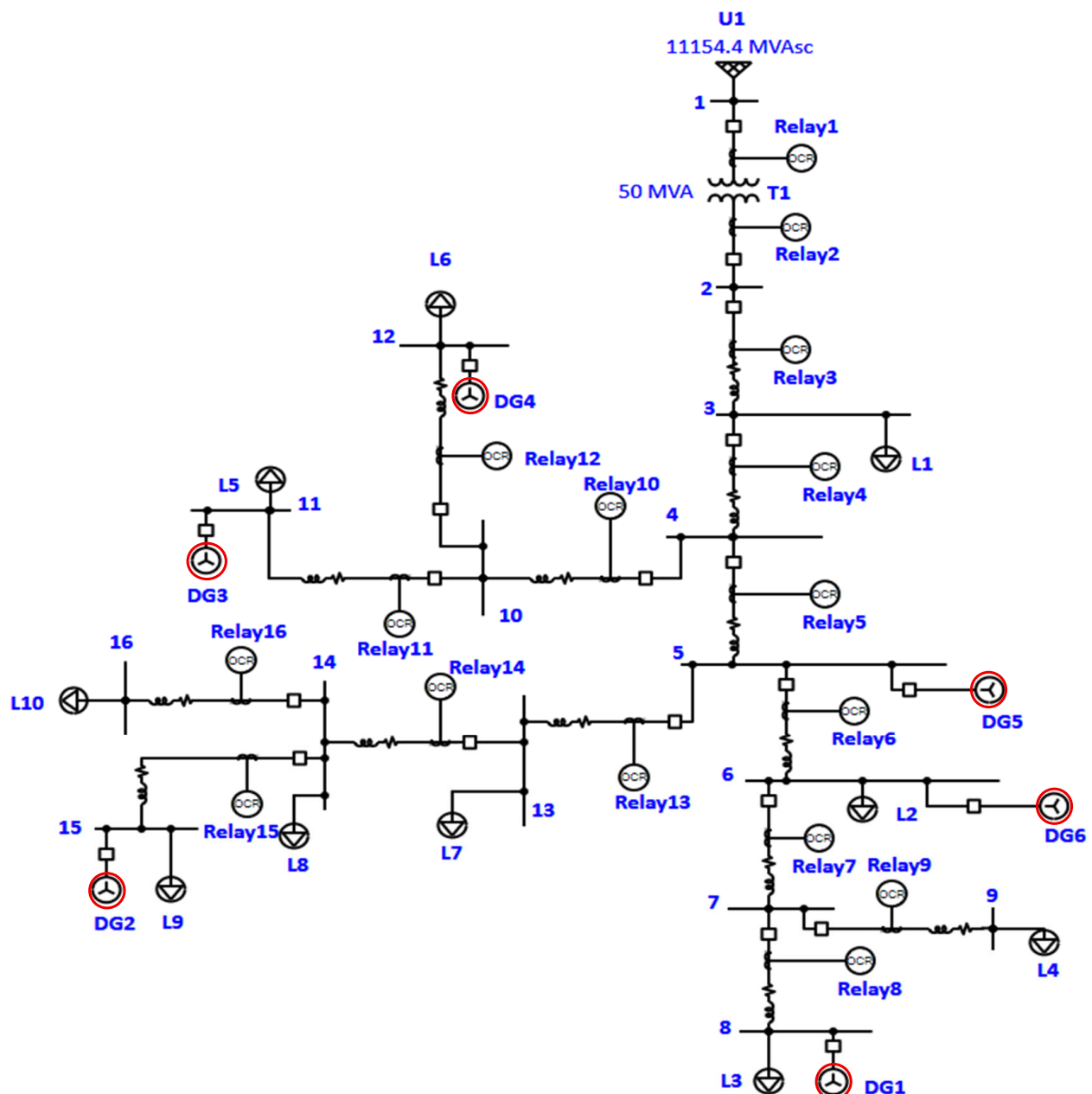


Figure 4. The topology of the 16-bus system.

Table 2. The installation and generation data of the DGs of the 16-bus system.

DG No.	DG Location Bus	Case 1	Case 2	Case 3	Case 4
		Gen. (kW)	Gen. (kW)	Gen. (kW)	Gen. (kW)
1	8	475	1425	2375	3325
2	15	463	1389	2315	3241
3	11	438	1314	2190	3066
4	12	475	1425	2375	3325
5	5	238	952	1190	1666
6	6	316	948	1580	2212
Total Gen. (kW)		2405 (PR 10%)	7215 (PR 30%)	12,025 (PR 50%)	16,835 (PR 70%)

Table 3. Load specification of the 16-bus system.

Load Location Bus	Load Type	Lumped Load	
		P_L (kW)	Q_L (kVAR)
3	PQ	2640	1391
6	PQ	2883	1130
8	Z	1190	518.7
9	Z	1706.6	300
11	PQ	1870	1159
12	PQ	2771	1717
13	Z	2295	1422
14	PQ	2592	1607
15	Z	3395	1446
16	PQ	2125	1317
Total		24,036	12,657

4.1.2. Test System 2: IEEE 37-Bus System

To evaluate the framework's scalability, the standard IEEE 37-bus test feeder was adopted as the second case study. The system was modified to include nine DGs, as shown in the single-line diagram in Figure 5. The operational data and locations for these DGs are presented in Table 4. The load bus location and power demand specification are shown in Table 5. For this system, scenarios were generated at three DER penetration levels: 50%, 65%, and 82%.

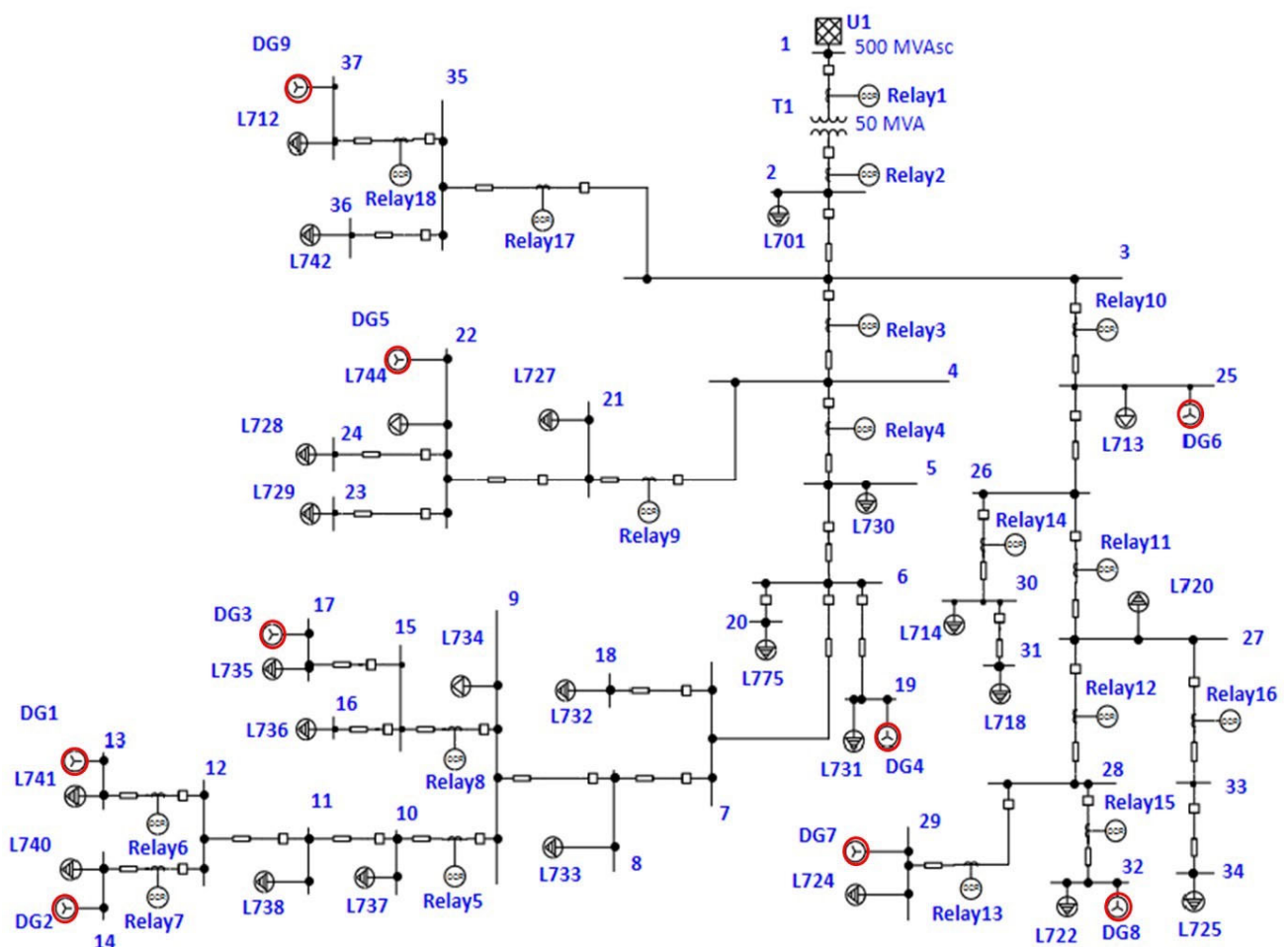
**Figure 5.** The topology of the IEEE 37-bus system.

Table 4. DG Generation of IEEE 37-Bus System.

DG No.	DG Location Bus	Case 1	Case 2	Case 3
		Gen. (kW)	Gen. (kW)	Gen. (kW)
1	13	300	1000	1000
2	14	600	800	1600
3	17	500	700	1150
4	19	700	800	800
5	22	650	750	800
6	25	500	750	950
7	29	650	500	550
8	32	600	700	750
9	37	700	1000	1200
Total Gen. (kW)		5200 (PR 50%)	7000 (PR 65%)	8800 (PR 82%)

Table 5. Load specification of the IEEE 37-bus system.

Load Location Bus	Load Type	Lumped Load	
		P _L (kW)	Q _L (kVAR)
2	PQ	630	315
5	PQ	425	200
8	PQ	425	200
9	PQ	210	105
10	PQ	700	350
11	PQ	630	310
13	PQ	210	105
14	PQ	425	200
16	Z	210	105
17	PQ	425	200
18	PQ	210	105
19	Z	170	80
20	PQ	178.8	110.8
21	PQ	840	420
22	PQ	210	105
23	PQ	210	105
24	PQ	598.8	371.1
25	PQ	425	200
27	PQ	425	200
29	Z	840	420
30	PQ	390	180
31	Z	170	80
32	PQ	161	80
34	PQ	210	105
36	Z	930	440
37	PQ	425	200
Total Gen.		10,683.6	5751.1

4.1.3. Implementation Details

All power system modeling and fault analyses for offline data generation were performed using ETAP 19.0.1. (Electrical Transient Analyzer Program) software. The optimization algorithm (RIA) and the entire CNN model, including development, training, and testing, were implemented in MATLAB R2025a. All computations were performed on a personal computer equipped with an Intel Core i7-10510U 2.3 GHz CPU, 32.0 GB of RAM, and an NVIDIA RTX 5060 with 16 GB of RAM. As described in Section 3, a comprehensive

dataset comprising over 10,000 unique samples across the two test systems (with variations in DER penetration and load levels) was generated to train the CNN.

The DER-generation and load-demand scenarios were generated by combining predefined DER penetration levels with load-variation conditions within the studied operating range. Each scenario was treated as an independent operating condition. For each scenario, ETAP fault analysis was performed to obtain the corresponding fault-current matrix, and the offline heuristic optimization process was then used to generate the optimized TMS and PCS reference targets. For constructing the CNN input matrix, the numerical values of R and M were specified for the corresponding test system. In the 16-bus system, R = 16 coordinated relays were considered, corresponding to the relay settings listed in Section 4.3.2, and M = 15 predefined fault locations were used, excluding the swing bus. Therefore, each input sample for the 16-bus system was represented as a 16×15 fault-current matrix. In the IEEE 37-bus system, R = 18 coordinated relays were considered, corresponding to the relay settings listed in Section 4.4.2, and M = 36 predefined fault locations were used, excluding the swing bus. Therefore, each input sample for the IEEE 37-bus system was represented as an 18×36 fault-current matrix. Before CNN training, the fault-current input matrices were normalized using min–max normalization. The minimum and maximum values used for scaling were derived solely from the training dataset and applied consistently to the validation and test datasets. The dataset was divided into training, validation, and testing subsets at the scenario level to reduce data leakage between model training and performance evaluation. Although the generated dataset covers the predefined DER-generation and load-demand combinations considered in this study, operating conditions outside this scenario space may require additional data generation and model retraining or fine-tuning.

In this study, the DER penetration levels were defined as discrete operating cases for scenario generation. For the 16-bus system, four DER penetration levels were considered: 10%, 30%, 50%, and 70% of the DG rated capacities. For the IEEE 37-bus system, three DER penetration levels were considered: 50%, 65%, and 82% of the DG rated capacities. These penetration levels were used to represent different DER operating conditions rather than minimum penetration thresholds. Additional load variations were combined with these DER levels to generate the training, validation, and testing scenarios.

4.2. Model Training and Validation Results

This section presents the CNN model training and validation results for both test systems. The primary objective is to demonstrate that the trained models achieve satisfactory prediction accuracy and stable predictive behavior under the tested conditions. In addition to the training and validation loss curves, an input-perturbation sensitivity test was conducted to evaluate the responses of the trained CNN models to small variations in the fault-current input matrix. Specifically, the original fault-current matrix for selected test cases was randomly perturbed by $\pm 3\%$ to emulate potential measurement variations in fault-current data. The perturbed input matrices were then fed into the trained CNN models, and the resulting TMS and PCS predictions were summarized using box-and-whisker plots.

4.2.1. 16-Bus System: Training and Sensitivity Analysis

For the 16-bus system, the dataset was generated based on the DG generation data listed in Table 2 and the load data listed in Table 3. Both DG generation and load demand were varied within $\pm 50\%$ of their nominal values. The variation range was divided into 10% intervals, and 500 random DG/load paired operating scenarios were generated for each interval. Consequently, 5000 candidate operating scenarios were initially generated for the 16-bus system. Each candidate scenario was then checked using a load-flow program to

verify whether the corresponding power system operating condition was feasible. Scenarios that did not lead to a solvable or feasible load-flow condition were excluded. After this screening process, 4680 feasible operating scenarios were retained and used for dataset construction, as summarized in Table 6.

Table 6. Composition of the 16-bus system dataset.

Test Condition	Number of Samples (Sets)
DER Penetration Variation	1600
Load Variation Scenarios	1600
Combined DER/load scenarios	1480
Total	4680

The retained feasible scenarios were then partitioned into training, validation, and testing subsets, as shown in Table 7.

Table 7. Data partitioning for the 16-bus system CNN model.

Data Type	Number of Records
Training	3800 (81.2%)
Validation	440 (9.4%)
Testing	440 (9.4%)
Total	4680

This means that all fault-current data and corresponding optimized relay-setting targets generated from the same DER/load operating scenario were assigned to only one subset, thereby reducing data leakage between the training and test sets. The CNN model was trained using the Adam optimizer to minimize mean squared error (MSE). Figure 6 illustrates the training and validation loss curves. The curves show rapid convergence to a low MSE. Crucially, the validation loss closely tracks the training loss without diverging, confirming that the model is well generalized and that no significant overfitting occurred.

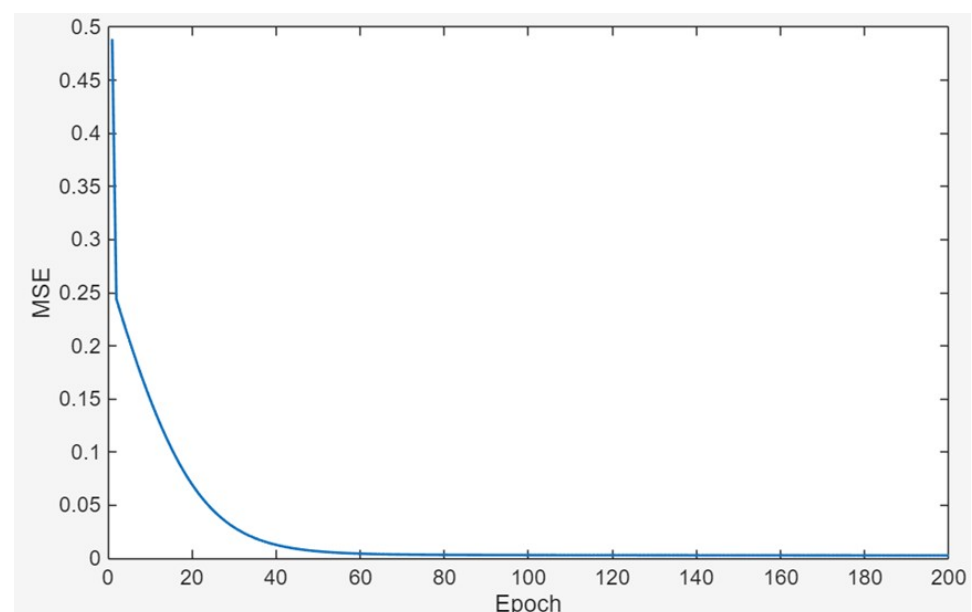


Figure 6. 16-Bus system MSE convergence curve.

To evaluate the trained CNN model's sensitivity to imperfect fault-current inputs, an input-perturbation test was conducted. For the 16-bus system, the original fault-current matrix of the selected testing case (Case 2, 50% PR) was randomly perturbed within $\pm 3\%$ to emulate possible measurement variation in fault-current data. A total of 30 perturbed input samples were generated and fed into the trained CNN model. The resulting TMS and PCS predictions were then summarized using box-and-whisker plots, as shown in Figures 7 and 8. The narrow interquartile ranges indicate that the CNN-predicted relay settings remain concentrated under small perturbations of the fault-current input matrix, suggesting that the trained CNN model has a certain degree of tolerance to small measurement variations.

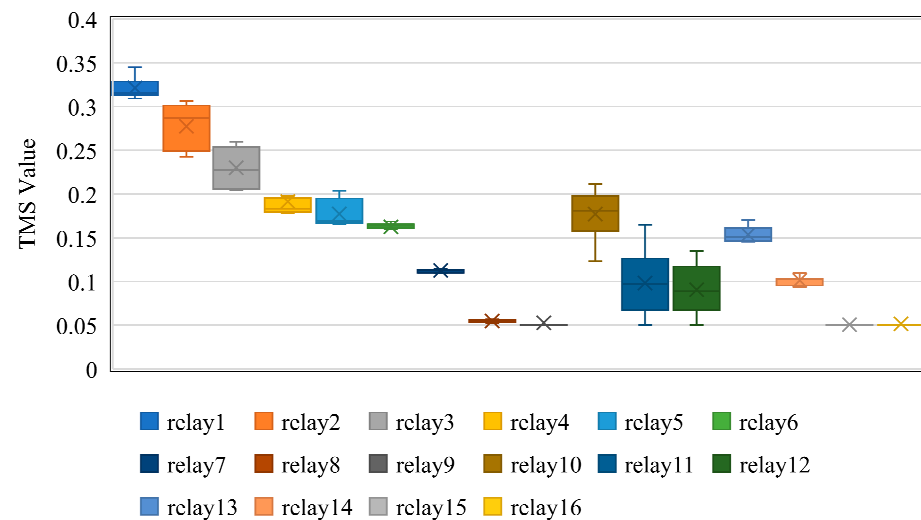


Figure 7. Box-and-whisker plot of CNN-predicted TMS values for the 16-bus system under $\pm 3\%$ fault-current input perturbations.

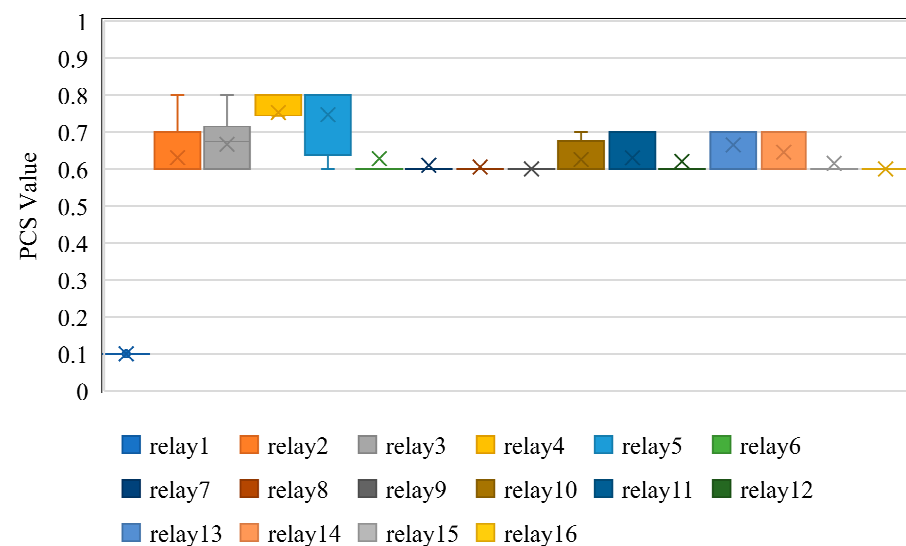


Figure 8. Box-and-whisker plot of CNN-predicted PCS values for the 16-bus system under $\pm 3\%$ fault-current input perturbations.

4.2.2. IEEE 37-Bus System: Training and Sensitivity Analysis

For the IEEE 37-bus system, the dataset was generated using the same procedure as that used for the 16-bus system. The DG generation data listed in Table 4 and the load data listed in Table 5 were used as the nominal operating data. Both DG generation and load demand were varied within $\pm 50\%$ of their nominal values. The variation range was divided

into 10% intervals, and random DG/load paired operating scenarios were generated for each interval. A total of 6500 candidate operating scenarios were initially generated. Each candidate scenario was then checked using a load-flow program to verify whether the corresponding system operating condition was feasible. After excluding scenarios that did not lead to a solvable or feasible load-flow condition, 6277 feasible operating scenarios were retained, as summarized in Table 8.

Table 8. Composition of the IEEE 37-bus system dataset.

Test Condition	Number of Samples (Sets)
DER Penetration Variation	2550
Load Variation Scenarios	2030
Combined DER/load scenarios	1697
Total	6277

The retained feasible scenarios were then partitioned into training, validation, and testing subsets, as shown in Table 9. Therefore, closely related samples generated from the same DG/load operating scenario were not allowed to appear simultaneously in both the training and test sets.

Table 9. Data partitioning for the CNN model of the IEEE 37-bus system.

Data Type	Number of Records
Training	5021 (80%)
Validation	628 (10%)
Testing	628 (10%)
Total	6277 (100%)

No fixed random seed was explicitly specified in the dataset-generation procedure. To support reproducibility, the variation range, interval-based random sampling procedure, feasibility screening process, and scenario-level data partitioning strategy are described in detail. To reduce data leakage, the dataset partitioning was performed at the operating-scenario level rather than at the individual fault-location or relay-measurement level. Therefore, all fault-current data and corresponding optimized relay-setting targets generated from the same DG/load operating scenario were assigned to only one subset, namely training, validation, or testing. This prevented closely related samples from the same operating condition from appearing simultaneously in both the training and test sets.

Figure 9 shows the MSE convergence curves for this system. Similarly to the 16-bus case, the model converges smoothly, proving its ability to learn the more complex, higher-dimensional protection coordination problem of this larger network.

During the development of this larger system, a hyperparameter study was conducted to find the optimal CNN architecture. Table 10 compares the performance of models with different numbers of convolutional layers and kernel sizes. While a single-layer CNN (3×3 kernel) achieved a low MSE (0.000702), the two-layer model with 3×3 kernels achieved the lowest MSE (0.000665) while also ensuring zero CTI violations and maintaining a fast operational time (COT). This architecture was therefore selected as the optimal design.

For the IEEE 37-bus system, the same input-perturbation procedure was applied. The original fault-current matrix of the selected testing case (82% PR) was randomly perturbed within $\pm 3\%$, and 50 perturbed input samples were generated. These samples were then fed into the trained CNN model to evaluate the sensitivity of the predicted relay settings under imperfect fault-current inputs. The resulting TMS and PCS distributions, presented

in Figures 10 and 11, show concentrated prediction distributions, indicating that the trained CNN model maintains stable prediction behavior despite small variations in fault-current measurements in the larger test system.

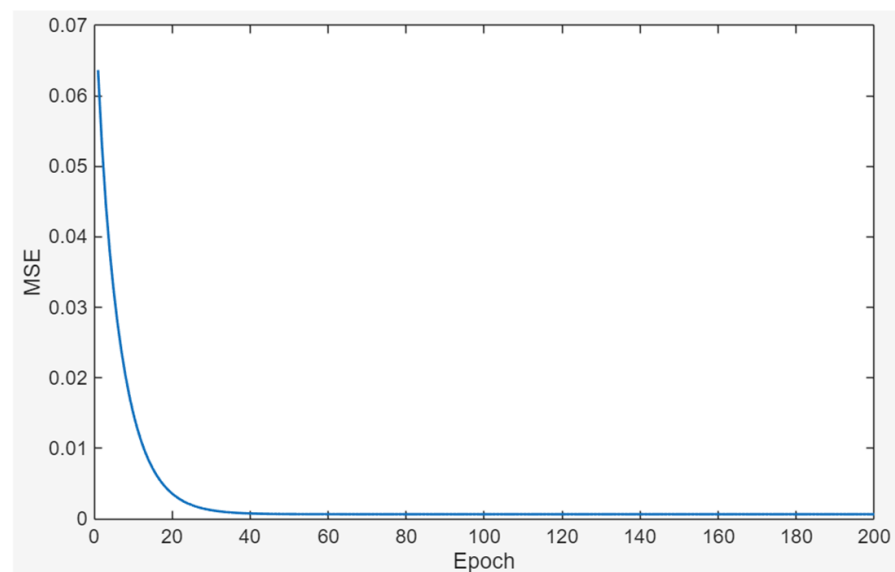


Figure 9. IEEE 37-bus system MSE convergence curve.

Table 10. Architectural comparison for CNN model selection based on MSE and CTI violations.

Convolution Layer Number	Kernel Sizes	MSE	Violation of CTI Restrictions	COT (Sec.)
1	3×3	0.000702	0	127.889428
1	5×5	0.000713	32	97.921631
2	3×3	0.000665	0	125.493302
2	5×5	0.000672	6	114.940257
3	3×3	0.000664	0	125.490958
3	5×5	0.000876	48	113.939937

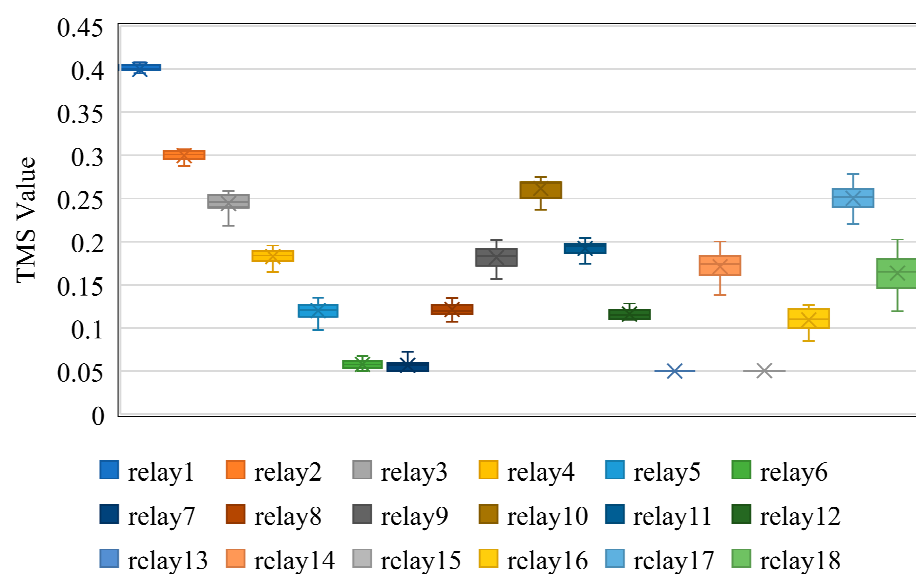


Figure 10. Box-and-whisker plot of CNN-predicted TMS values for the IEEE 37-bus system under $\pm 3\%$ fault-current input perturbations.

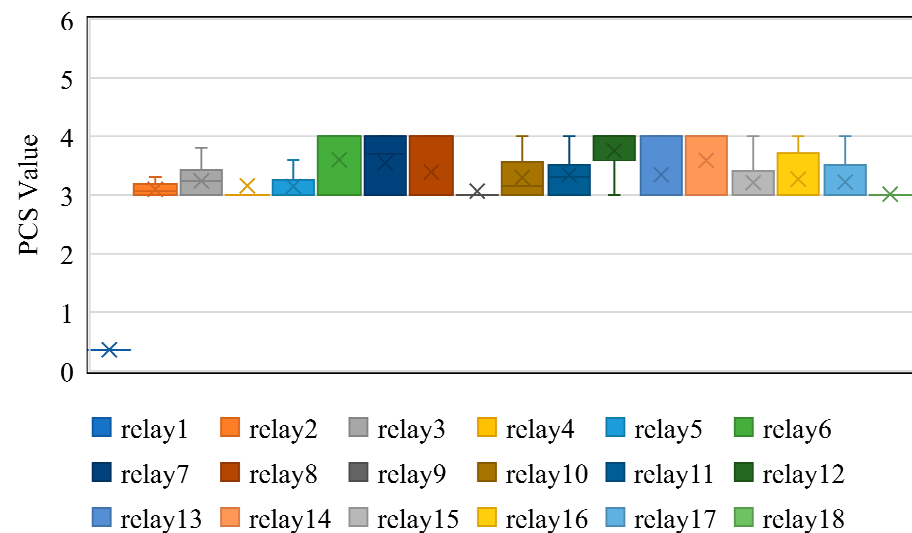


Figure 11. Box-and-whisker plot of CNN-predicted PCS values for the IEEE 37-bus system under $\pm 3\%$ fault-current input perturbations.

4.2.3. Baseline Comparison Between CNN and MLP

To further evaluate the suitability of the CNN architecture, a multilayer perceptron (MLP) was implemented as a baseline surrogate model. Unlike the CNN, which uses the structured $R \times M$ fault-current matrix to extract relay–fault-location correlation patterns, the MLP receives a one-dimensional relay–current vector as input. Specifically, for each fault-location sample, the input vector consists of the fault currents observed by all coordinated relays. Therefore, the input dimension of the MLP is $R \times 1$. The output layer was designed to predict the complete relay-setting vector, including both TMS and PCS values for all coordinated relays, resulting in an output dimension of $2R \times 1$. For the 16-bus system, the MLP input and output dimensions were 16×1 and 32×1 , respectively. For the IEEE 37-bus system, the corresponding input and output dimensions were 18×1 and 36×1 , respectively.

The MLP baseline consisted of three fully connected hidden layers with 128, 64, and 64 neurons. ReLU activation functions were used after each hidden layer. The model was trained using the Adam optimizer with the mean squared error (MSE) loss function. The maximum number of training epochs was set to 300, and no early stopping mechanism was applied. The same training, validation, and testing data partitions were used for the MLP and CNN models to maintain consistency in the baseline comparison.

The purpose of this comparison is to examine whether preserving the relay–fault-location matrix structure through convolutional feature extraction provides an advantage over a fully connected regression model that uses relay–current vectors as inputs. The prediction accuracy of the two models was evaluated using mean absolute error (MAE) and mean absolute percentage error (MAPE). MAE quantifies the average absolute deviation between the predicted relay settings and the optimized reference values, whereas MAPE expresses the prediction error as a percentage.

As shown in Table 11, the CNN surrogate model achieved lower prediction errors than the MLP baseline for both TMS and PCS in the two test systems. For the 16-bus system, the CNN reduced the TMS MAE from 0.023 to 0.015 and the TMS MAPE from 2.68% to 1.72%. For PCS prediction, the MAE decreased from 0.0273 to 0.0143, and the MAPE decreased from 2.94% to 1.61%. Similar results were obtained for the IEEE 37-bus system. The CNN reduced the TMS MAE from 0.037 to 0.021 and the TMS MAPE from 3.01% to 2.30%. For PCS prediction, the MAE decreased from 0.0281 to 0.0192, and the MAPE decreased from 3.42% to 2.11%.

Table 11. Prediction accuracy comparison between the MLP and CNN surrogate models.

Test System	Model	TMSs		PCSs	
		MAE	MAPE (%)	MAE	MAPE (%)
16-bus system	MLP	0.023	2.68	0.0273	2.94
	CNN	0.015	1.72	0.0143	1.61
IEEE 37-bus system	MLP	0.037	3.01	0.0281	3.42
	CNN	0.021	2.30	0.0192	2.11

4.3. Case Study 1: Performance Analysis on the 16-Bus System

This section validates the practical effectiveness of the trained CNN model. The goal is to demonstrate that the relay settings (TMSs and PCSs) predicted by the CNN can ensure proper protection coordination when implemented in the ETAP simulation environment.

4.3.1. Baseline Performance with Fixed Settings of the 16-Bus System

To test the model's generalization, three distinct test cases (Case 1, 2, 3) representing different DER penetration rates (PRs) were used. These cases were not part of the original training or validation datasets. First, to establish a baseline, the system's performance was evaluated without adaptive coordination (i.e., using traditional fixed settings). Table 12 details the operational status of primary and backup relays for various fault locations under these fixed settings. The table reveals 12 distinct miscoordination (MC) conditions across the different PR cases in which CTI violations occur. This result confirms that traditional fixed settings are inadequate and unreliable in grids with high DER penetration. In the table, N/A indicates relays that are not involved in the corresponding protection coordination path or that do not detect a fault current requiring operating-time evaluation at the specified fault location. The same notation is used in the following relay coordination and miscoordination tables.

Table 12. Relay miscoordination in the 16-bus system.

Cond.	Fault Loc. (Bus)	Case	Fault Current (kA) at Relay/Relay OT (Sec.)					Miscoordination Statement
			Relay 1	Relay 2	Relay 3	Relay 4	Relay 5	
1	3	1	5.956/1.54	5.954/1.16	5.956/0.991	N/A	N/A	CTI out of range
2	4	1	5.745/1.59	5.762/1.18	5.753/1.01	6.284/0.875	N/A	CTI out of range
3	5	1	5.494/1.65	5.494/1.21	5.494/1.04	5.991/0.902	6.941/0.784	CTI out of range
4	3	2	5.934/1.55	5.933/1.16	5.927/0.994	3.479/1.37	N/A	Relay 4 mis-action
5	4	2	5.773/1.58	5.747/1.18	5.765/1.01	6.287/0.875	N/A	CTI out of range
6	5	2	5.486/1.65	5.459/1.22	5.474/1.04	5.998/0.901	7.235/0.767	CTI out of range
7	3	3	5.947/1.54	5.943/1.16	5.943/0.992	4.147/1.17	N/A	R4 mis-action
8	4	3	5.779/1.58	5.754/1.18	5.798/1.01	6.291/0.874	N/A	CTI out of range
9	5	3	5.481/1.65	5.487/1.22	5.445/1.05	5.985/0.902	7.496/0.752	CTI out of range
10	3	4	5.946/1.54	5.943/1.16	5.943/0.992	4.845/1.04	N/A	R4 mis-action
11	4	4	5.762/1.58	5.763/1.18	5.763/1.01	6.285/0.875	3.025/1.5	R5 mis-action
12	5	4	5.466/1.66	5.464/1.22	5.464/1.05	5.959/0.905	7.776/0.736	CTI out of range

Note: Each entry is expressed as fault current/operating time. OT denotes operating time, and CTI denotes the coordination time interval between the primary and backup relays. "CTI out of range" indicates that the CTI violates the required coordination margin, while "mis-action" indicates an incorrect relay operating sequence.

4.3.2. CNN-Predicted Relay Settings for the 16-Bus System

Next, the proposed CNN framework was applied. The system conditions for the same three test cases were fed into the trained CNN model. The model instantaneously generated new, "preventively calibrated" TMSs and PCSs. Table 13 presents the CNN-predicted relay

parameters, including TMSs and PCSs, for each relay across the unseen test cases. These settings are subsequently validated in ETAP.

Table 13. The 16-bus system CNN-predicted relay settings.

Relay No	Ip					TMSs				
	Init.	Case 1	Case 2	Case 3	Case 4	Init.	Case 1	Case 2	Case 3	Case 4
1	240	200	200	240	220	0.28	0.315	0.322	0.335	0.33
2	1000	1265	1372	1600	1383	0.29	0.293	0.284	0.243	0.271
3	1200	1600	1200	1490	1222	0.26	0.209	0.264	0.216	0.248
4	1200	1200	1600	1600	1395	0.21	0.223	0.19	0.179	0.197
5	1200	1600	1286	1600	1600	0.20	0.161	0.199	0.166	0.169
6	1200	1200	1200	1200	1200	0.153	0.157	0.158	0.163	0.166
7	1200	1200	1200	1200	1200	0.09	0.109	0.107	0.112	0.114
8	1200	1200	1200	1197	1200	0.05	0.05	0.052	0.056	0.06
9	1200	1200	1200	1189	1200	0.06	0.05	0.05	0.05	0.05
10	1300	1200	1400	1420	1250	0.127	0.145	0.203	0.192	0.19
11	1200	1200	1200	1200	1430	0.05	0.05	0.116	0.13	0.118
12	1200	1200	1400	1270	1220	0.05	0.05	0.092	0.136	0.052
13	1200	1400	1400	1429	1400	0.15	0.139	0.155	0.159	0.15
14	1200	1200	1200	1400	1400	0.09	0.1	0.115	0.094	0.096
15	1200	1200	1205	1200	1260	0.05	0.05	0.065	0.05	0.05
16	1200	1200	1200	1195	1220	0.05	0.05	0.05	0.05	0.05

4.3.3. Coordination Validation in ETAP for the 16-Bus System

The CNN-predicted settings listed in Table 13 were implemented in the ETAP model, and the same fault simulations were repeated. Table 14 summarizes the coordination verification results for the 16-bus system. Compared with the fixed-setting baseline in Table 12, the 12 miscoordination events were eliminated after applying the CNN-predicted relay settings. The primary and backup relays operated in the expected sequence, and the corresponding CTI values remained within the required coordination range of 0.2–0.4 s across the studied scenarios. These results indicate that the proposed CNN-based preventive calibration framework can provide coordinated relay-setting recommendations for the 16-bus system.

Table 14. Relay well-coordination results of the CNN-predicted settings in the 16-bus system.

Cond.	Fault Loc. (Bus)	Case	Fault Current (kA) at Relay/Relay OT (Sec.)				
			Relay 1	Relay 2	Relay 3	Relay 4	Relay 5
1	3	1	5.956/1.57	5.954/1.29	5.956/1.09	N/A	N/A
2	4	1	5.745/1.61	5.762/1.32	5.753/1.12	6.284/0.919	N/A
3	5	1	5.509/1.66	5.494/1.37	5.501/1.16	6.007/0.952	6.957/0.752
4	3	2	5.934/1.55	5.933/1.34	5.927/1.14	N/A	N/A
5	4	2	5.773/1.58	5.747/1.37	5.765/1.16	6.287/0.959	N/A
6	5	2	5.486/1.64	5.459/1.42	5.474/1.2	5.998/0.996	7.235/0.793
7	3	3	5.947/1.49	5.943/1.29	5.943/1.08	N/A	N/A
8	4	3	5.779/1.52	5.754/1.32	5.798/1.1	6.291/0.899	N/A
9	5	3	5.481/1.58	5.487/1.37	5.445/1.15	5.985/0.938	7.496/0.715
10	3	4	5.946/1.54	5.943/1.16	5.943/0.992	4.845/1.04	N/A
11	4	4	5.762/1.62	5.763/1.31	5.763/1.1	6.285/0.9	N/A
12	5	4	5.466/1.69	5.464/1.37	5.464/1.14	5.959/0.938	7.776/0.738

4.3.4. TCC Curve Validation for the 16-Bus System

To visually verify the coordination results summarized in Table 14, three representative TCC curve plots are presented. These curves were generated in ETAP using the TMS and PCS values predicted by the CNN, as listed in Table 13. Figure 12 illustrates the TCC plot

for a fault at Bus 5 in Case 2. The relay operating sequence follows Relay 5 → Relay 4 → Relay 3 → Relay 2 → Relay 1, and the CTI between adjacent primary–backup relay pairs remains within the required coordination range. Figure 13 corresponds to a fault at Bus 3 in Case 3, where the relay sequence follows Relay 3 → Relay 2 → Relay 1. Figure 14 shows the TCC plot for a fault at Bus 4 in Case 4, where the relay operating sequence follows Relay 4 → Relay 3 → Relay 2 → Relay 1.

These representative TCC plots provide visual confirmation that the CNN-predicted relay settings can restore the expected primary–backup operating sequence and maintain the required CTI margins for the selected 16-bus test cases. Together with the numerical results in Table 14, these plots support the effectiveness of the proposed preventive calibration framework in mitigating DER-induced protection miscoordination.

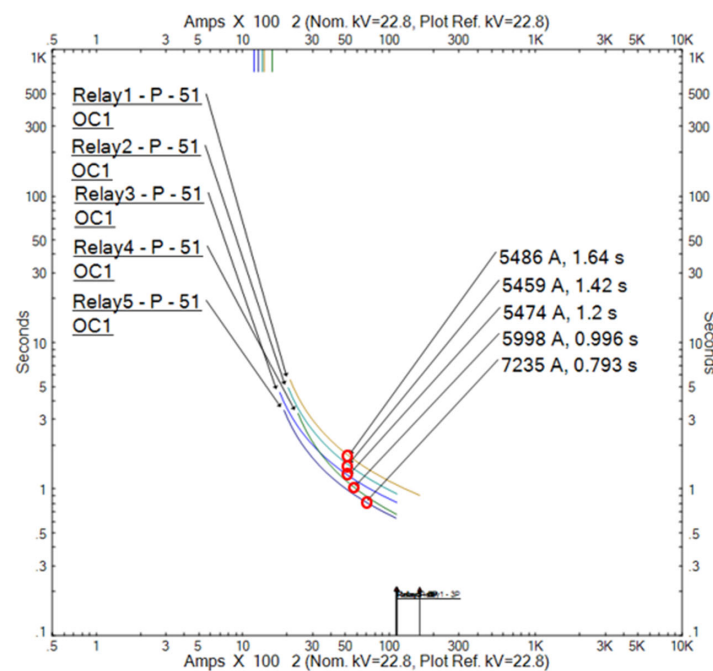


Figure 12. TCC curves of Relays 1–5 for condition 6 in the 16-bus system.

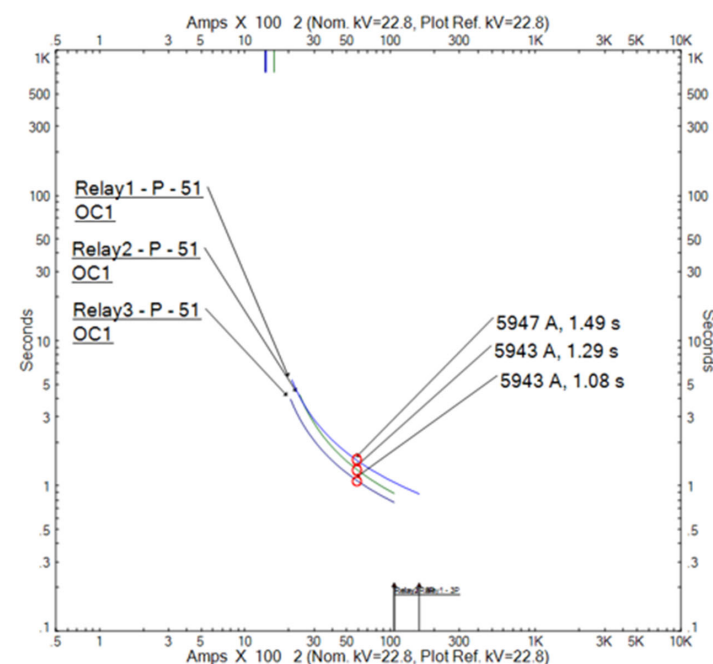


Figure 13. TCC curves of Relays 1–3 for condition 7 in the 16-bus system.

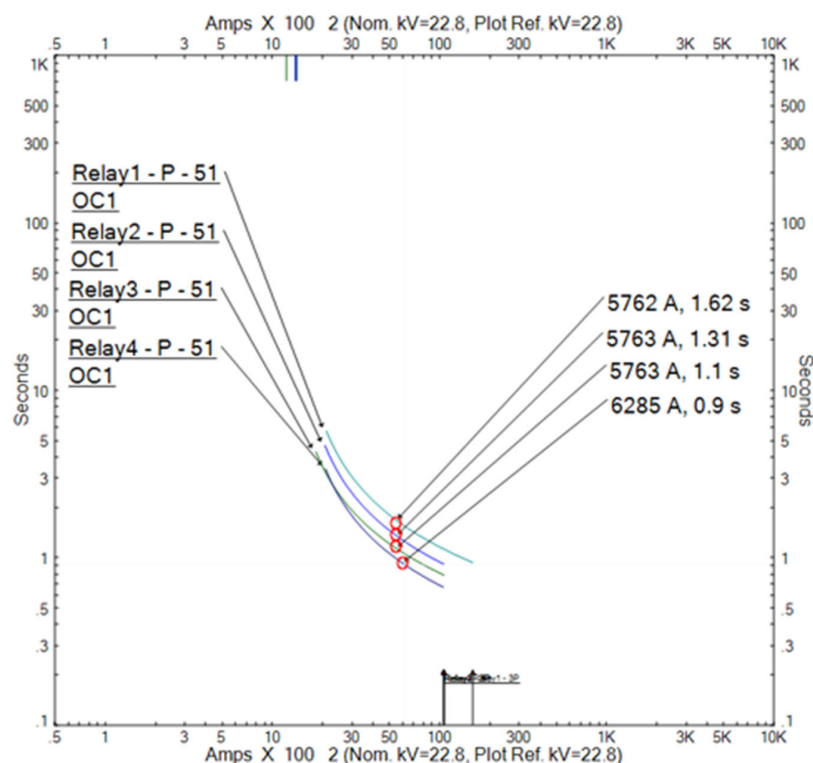


Figure 14. TCC curves of Relays 1–4 for condition 11 in the 16-bus system.

4.4. Case Study 2: Performance Analysis on the IEEE 37 Bus System

The trained CNN model demonstrates the scalability and generalizability of the proposed framework, which was validated on the larger, more complex IEEE 37-bus test system. This section follows the same rigorous validation procedure as the 16-bus case.

4.4.1. Baseline Performance with Fixed Settings of the IEEE 37 Bus System

First, the system's baseline performance using traditional, fixed relay settings was established. Three challenging test cases (Cases 1, 2, and 3) representing different DER penetration levels (50%, 65%, and 82%) were selected. These cases were not part of the model training data. Table 15 details the simulation results for these cases using fixed settings. Similarly to the 16-bus system, the IEEE 37-bus system suffers from significant protection failures. The results show six critical MC events, such as 'R4 mis-action', in which the backup relay trips faster than the primary relay, compromising system selectivity. The initial TMSs and PCSs were simulated, confirming that static settings are insufficient for complex, dynamic grids.

Table 15. Relay miscoordination in the IEEE 37-bus system.

Cond.	Fault Loc. (Bus)	Case	Fault Current (kA) at Relay/Relay OT (Sec.)						MC Statement
			R1	R2	R3	R4	R10	R11	
1	3	1	5.663/1.54	5.665/1.15	N/A	0.978/1.36	N/A	N/A	R4 mis-action
2	27	1	5.473/1.58	5.472/1.18	N/A	0.945/1.45	6.988/0.77	7.145/0.597	R4 mis-action
3	3	2	5.663/1.54	5.665/1.15	N/A	1.135/1.05	N/A	N/A	R4 mis-action
4	27	2	5.466/1.58	5.463/1.18	N/A	1.095/1.11	7.189/0.76	7.384/0.587	R4 mis-action
5	3	3	5.663/1.54	5.665/1.15	1.634/1.43	1.310/0.861	N/A	N/A	R3 and R4 mis-action
6	27	3	5.458/1.58	5.457/1.18	1.574/1.5	1.262/0.904	7.385/0.75	7.611/0.579	R3 and R4 mis-action

Note: N/A follows the same definition given in Table 12. Each relay entry is expressed as fault current/operating time. MC denotes miscoordination.

4.4.2. CNN-Predicted Relay Settings for the IEEE 37 Bus System

Next, the operating conditions for the same three test cases were fed into the CNN model trained for the IEEE 37-bus system, as described in Section 4.2.2. The model generated a new set of recommended TMS and PCS parameters for the tested operating conditions. Table 16 presents the CNN-predicted relay settings for all relevant relays across the three test cases.

Table 16. IEEE 37-bus CNN-predicted relay settings.

No	Ip				TMSs			
	Init.	Case 1	Case 2	Case 3	Init.	Case 1	Case 2	Case 3
1	129	129	105	105	0.4	0.37	0.402	0.35
2	930	957	987	900	0.29	0.3	0.29	0.307
3	750	1008	996	930	0.16	0.234	0.235	0.252
4	810	900	972	900	0.11	0.183	0.176	0.187
5	900	1200	1044	900	0.09	0.108	0.115	0.121
6	900	903	1170	900	0.06	0.06	0.055	0.065
7	960	900	900	900	0.068	0.07	0.065	0.056
8	1050	900	1200	1077	0.111	0.112	0.111	0.103
9	900	1200	900	900	0.17	0.164	0.17	0.2
10	900	1059	900	1056	0.18	0.248	0.26	0.255
11	900	1176	966	900	0.13	0.174	0.186	0.2
12	900	1200	1191	1200	0.107	0.11	0.107	0.114
13	900	1137	906	900	0.05	0.05	0.05	0.05
14	960	1110	900	900	0.192	0.182	0.192	0.193
15	840	1110	900	1200	0.05	0.05	0.05	0.05
16	1050	900	1200	900	0.097	0.09	0.104	0.11
17	930	930	1200	1200	0.227	0.244	0.05	0.21
18	900	900	900	900	0.163	0.166	0.097	0.16

4.4.3. Coordination Validation in ETAP for the IEEE 37 Bus System

The CNN-predicted settings listed in Table 16 were implemented in the ETAP simulation model, and the fault analyses were repeated under the same test conditions. Table 17 summarizes the coordination verification results for the IEEE 37-bus system. Compared with the fixed-setting baseline in Table 15, the six miscoordination events were eliminated after applying the CNN-predicted relay settings. For each studied fault location, the primary and backup relays operated in the expected sequence, and the corresponding CTI values were restored to the required coordination margin. These results indicate that the proposed CNN-based preventive calibration framework can provide coordinated relay-setting recommendations for a larger and more complex distribution test system.

Table 17. Coordination verification results of the CNN-predicted settings in the IEEE 37-bus system.

Cond.	Fault Loc. (Bus)	Case	Fault Current (kA) at Relay/Relay OT (Sec.)					
			R1	R2	R3	R4	R10	R11
1	3	1	5.663/1.34	5.665/1.14	N/A	N/A	N/A	N/A
2	27	1	5.473/1.43	5.472/1.19	N/A	N/A	6.988/0.904	7.145/0.672
3	3	2	5.663/1.37	5.665/1.12	N/A	N/A	N/A	N/A
4	27	2	5.466/1.42	5.463/1.19	N/A	N/A	7.189/0.925	7.384/0.712
5	3	3	5.663/1.35	5.665/1.12	N/A	N/A	N/A	N/A
6	27	3	5.458/1.38	5.457/1.16	N/A	N/A	7.385/0.886	7.611/0.64

4.4.4. TCC Curve Validation for the IEEE 37 Bus System

To visually verify the coordination results summarized in Table 17, three representative TCC curve plots are presented. These curves were generated in ETAP using the TMS and PCS values predicted by the CNN, as listed in Table 16. Figures 15–17 show representative TCC curves for faults at Bus 3 in Case 1, Bus 27 in Case 2, and Bus 27 in Case 3, respectively. In these representative cases, the relay operating sequences follow the expected primary–backup order, and the CTI values between adjacent relay curves remain within the required coordination margin. These TCC plots provide visual confirmation that the CNN-predicted relay settings can restore the expected relay operating sequence and maintain the required CTI margins for the selected IEEE 37-bus test cases. Together with the numerical results in Table 17, these plots support the applicability of the proposed preventive calibration framework to a larger and more complex distribution test system.

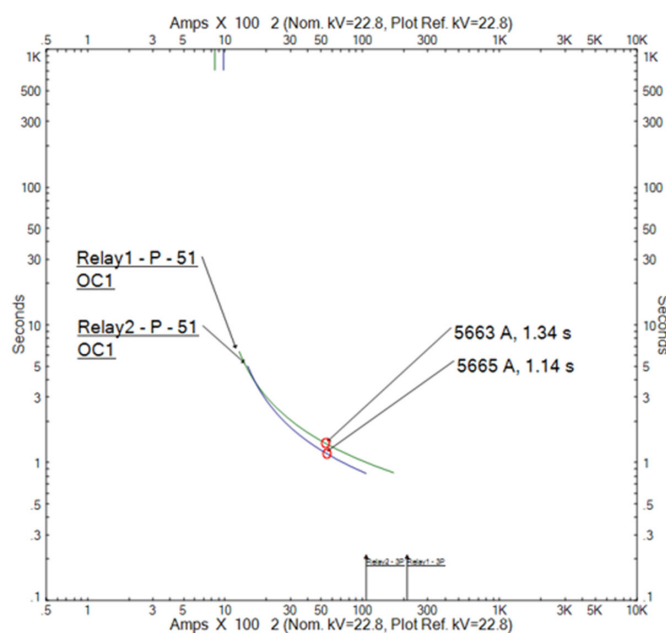


Figure 15. TCC curves of Relays 1–2 for condition 1 in the IEEE 37-bus system.

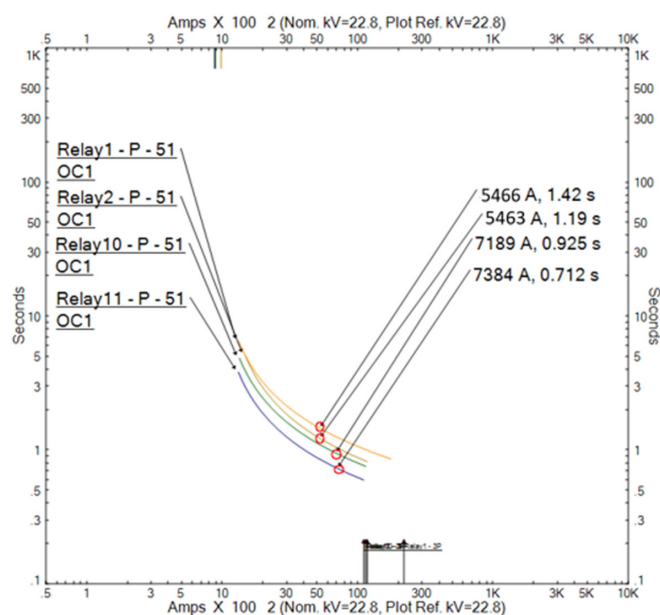


Figure 16. TCC curves of Relays 1, 2, 10, and 11 for condition 4 in the IEEE 37-bus system.

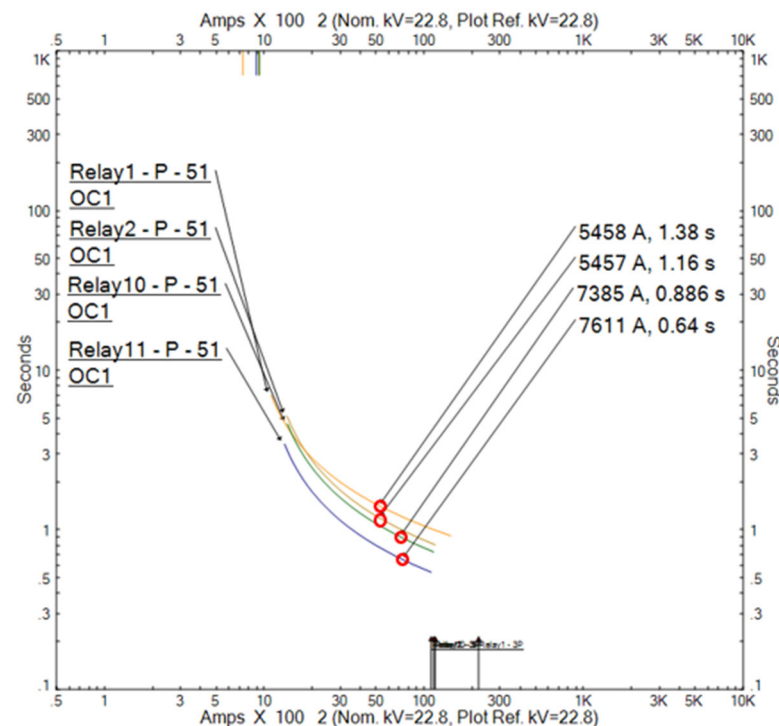


Figure 17. TCC curves of Relays 1, 2, 10, 11 for condition 6 in the IEEE 37-bus system.

4.5. Computational Efficiency Comparison

In addition to prediction accuracy and protection coordination performance, computational efficiency is an important factor for the practical application of preventive relay setting calibration. In the proposed offline–online framework, the heuristic optimization algorithms and the CNN model serve different computational roles. RIA and MPSO are used in the offline stage to generate optimized relay-setting labels across different DER-generation and load-demand operating scenarios, whereas the trained CNN is used in the online stage to provide rapid relay-setting recommendations after offline training is complete.

It should be emphasized that the computation times of RIA/MPSO and CNN inference are not intended to represent competing algorithms executed in the same operational stage. For a given operating scenario, defined by a specific DER-generation condition and load-demand condition, RIA or MPSO performs iterative protection-coordination optimization using the fault-current information observed by the coordinated relays. In this study, the average computation time required to obtain the optimized TMSs and PCSs for one operating scenario was approximately 300 s for RIA and 250 s for MPSO. When the optimization procedure is repeated across many operating scenarios during dataset construction, the accumulated offline computation time can easily reach several thousand seconds or more.

By contrast, the CNN is trained offline using the optimized TMS and PCS labels generated from various operating scenarios. After training, the CNN directly maps the fault-current input features of a new operating condition to the recommended TMSs and PCSs without re-executing the iterative optimization process. Therefore, the CNN computation time reported in Table 18 represents the online inference time of the trained model, rather than the time required for offline optimization or model training. Table 18 summarizes the computation times for the IEEE 37-bus test system and distinguishes between the offline optimization-based label-generation stage and the online CNN inference stage.

Table 18. Computation time comparison between offline optimization-based label generation and online CNN inference.

Method	Execution Stage	Computational Role	Measurement Basis	Computation Time
RIA	Offline optimization	Optimized TMS/PCS label generation	per operating scenario	Approximately 300 s for one scenario
MPSO	Offline optimization	Optimized TMS/PCS label generation	per operating scenario	Approximately 250 s for one scenario
CNN	Online inference	Relay-setting recommendation after offline training	per new input sample	Fast surrogate inference for relay-setting recommendation

As shown in Table 18, the heuristic optimization algorithms require considerably more computation time because they solve the relay coordination problem through iterative search and repeated evaluation of coordination constraints. This computational requirement is acceptable for offline planning and training-data generation, but it limits their direct use for frequent online relay-setting adjustment. By contrast, once the CNN has been trained using the optimized relay-setting labels generated in the offline stage, it can rapidly infer the corresponding TMS and PCS recommendations for a new operating scenario.

Therefore, the purpose of the computational-time analysis is not to claim that the CNN replaces RIA or MPSO in the offline design stage. Instead, it demonstrates that the proposed offline–online architecture integrates the strengths of both approaches. RIA and MPSO provide optimization-based reference relay settings for supervised learning, while the CNN provides fast relay-setting recommendations for online decision support. Before practical implementation, the CNN-recommended settings should still be verified through setting-boundary checks, primary–backup relay sequence confirmation, and CTI validation.

5. Discussion

5.1. Mechanism of Protection Miscoordination Under DER Penetration

The simulation results indicate that protection miscoordination becomes more likely as DER penetration and load conditions vary. This phenomenon is mainly caused by the variation in fault-current magnitude and direction introduced by DERs. In conventional radial distribution systems, fault currents generally flow from the substation toward the fault point, and relay coordination can be achieved using fixed-time-current characteristics. However, when DERs are connected at different feeder locations, they may contribute additional fault current from downstream or lateral positions. As a result, the fault current observed by each relay may increase, decrease, or change its relative distribution depending on the DER output, load demand, and fault location.

These changes directly affect the operating time of inverse-time overcurrent relays. A relay that originally served as a backup device may experience a fault current close to, or even exceeding, that observed by the primary relay, leading to a reduced coordination time interval or an incorrect operating sequence. Therefore, the miscoordination cases observed in the test systems are the result not only of increased fault-current magnitude, but also of the redistribution of fault-current contributions among primary and backup relay pairs. This explains why fixed relay settings may fail under DER-rich operating conditions, even though they were effective in the original base case.

The proposed CNN-based framework addresses this issue by learning the relationship between the system-wide fault-current signature and the corresponding optimized relay-setting vector. Instead of using a single fault current or operating condition as input, the CNN uses a structured fault-current matrix that captures the responses of multiple relays

across predefined fault locations. This enables the model to extract correlation patterns among fault locations, relay measurements, and operating scenarios. Consequently, the trained CNN can recommend TMS and PCS values that restore the primary–backup operating sequence and maintain the CTI constraints under the studied DER and load scenarios.

5.2. Practical Implementation and Safety Boundaries

Although the proposed CNN-based framework can rapidly recommend relay settings, practical implementation requires additional safety mechanisms before any setting is applied to field devices. The predicted TMS and PCS values should not be written directly to intelligent electronic devices without verification. Before implementation, the recommended settings must pass relay-setting boundary checks, primary–backup relay-sequence checks, and CTI verification. These verification procedures are necessary to ensure that the predicted settings satisfy both numerical limits and protection coordination requirements.

In practical deployment, the CNN output should be regarded as a decision-support recommendation for operators or protection engineers. The verified settings may then be submitted for operator confirmation or transferred to IEDs through a secured engineering procedure. Online modification of relay settings should also consider communication security, authentication, event logging, protection blocking, and rollback procedures. If abnormal input data, incomplete measurements, or out-of-range operating conditions are detected, the system should reject the CNN recommendation and retain the existing verified relay settings.

Therefore, the proposed method is positioned as a preventive calibration and decision-support framework rather than an unrestricted automatic setting-update mechanism. Full closed-loop field implementation would require further validation of cybersecurity, communication reliability, protection blocking logic, and coordination with utility operating procedures.

5.3. Retraining Requirements Under Permanent Topology Changes

The trained CNN model is valid within the operating and topological conditions represented in the offline-generated dataset. For variations in DER output and load demand that remain within the training range, the model can provide relay-setting recommendations without retraining. However, if a permanent topology change occurs, such as feeder reconfiguration, relay relocation, addition of new DER units, removal of existing DER units, or modification of primary–backup relay pairs, the fault-current distribution and relay coordination relationships may change substantially.

Under such conditions, the original CNN model should not be directly applied without additional verification. A modified network model should first be established, and new fault-current data should be generated through short-circuit analysis. The relay-setting optimization procedure should then be performed again to update the supervised learning targets. Finally, the CNN model should be retrained or fine-tuned using the updated dataset before deployment.

This topology-dependent retraining requirement is a limitation of the present framework. Nevertheless, the offline–online architecture remains applicable because the retraining process can be performed offline after major network modifications. Future work will investigate transfer learning and incremental learning techniques to reduce the retraining burden when the distribution network undergoes partial topology changes.

5.4. Limitations and Future Work

Several limitations of this study should be noted. First, the proposed CNN model is trained using simulation data generated from predefined DER-generation and load-demand scenarios. Therefore, the model performance depends on the representativeness of the offline dataset. If the online operating condition falls outside the training range,

the recommendation should be treated with caution and verified before use. In addition, although the dataset-generation procedure considered $\pm 50\%$ variations in DG generation and load demand and retained only load-flow-feasible operating scenarios, the resulting dataset still represents a predefined scenario space. No fixed random seed was explicitly specified in the original random sampling process; therefore, the reported variation range, interval-based sampling procedure, feasibility screening, and scenario-level data partitioning are important for supporting reproducibility. The scenario-level partitioning strategy reduces data leakage by preventing closely related samples generated from the same operating condition from appearing simultaneously in both the training and test sets.

Second, the present study focuses on preventive relay setting calibration based on fault-current signatures and optimized TMS/PCS targets. Although the $\pm 3\%$ fault-current perturbation tests indicate stable CNN prediction behavior under small measurement variations, broader robustness evaluations that account for missing measurements, communication delays, topology errors, and cyber-related data anomalies are still necessary before field deployment.

Third, although the proposed method improves online decision speed by replacing iterative optimization with CNN inference, the quality of the predicted settings still depends on the quality of the offline-optimized labels. Thus, the heuristic optimization results should be interpreted as high-quality optimized solutions rather than mathematically proven global optima. Although the CNN outperformed the MLP baseline in this study, only one representative fully connected neural-network baseline was considered. Future research will include additional machine-learning baselines, broader sensitivity analysis under noisy measurements, topology-change scenarios, and hardware-in-the-loop or field-oriented validation to further evaluate the practical applicability of the proposed framework.

6. Conclusions

The increasing penetration of DERs introduces significant variations in fault-current magnitude, direction, and distribution, making conventional fixed-setting overcurrent protection schemes increasingly vulnerable to miscoordination. To address this issue, this paper proposed and evaluated a deep learning-based preventive relay setting calibration framework for enhancing protection coordination robustness in DER-rich distribution networks. The proposed framework adopts an offline–online architecture. In the offline stage, heuristic optimization algorithms, including RIA and MPSO, were used in conjunction with ETAP-based fault analysis to generate a comprehensive dataset of high-quality optimized TMS and PCS values across various DER-generation and load-demand scenarios. The training datasets were constructed from randomly sampled DG/load operating scenarios within the predefined variation ranges, screened by load-flow feasibility analysis, and partitioned at the operating-scenario level to reduce data leakage. In the online stage, a lightweight CNN was trained as a fast surrogate model to learn the nonlinear mapping between system fault-current signatures and the corresponding optimized relay-setting vectors.

The proposed framework was evaluated using a 16-bus distribution system and the IEEE 37-bus test feeder. The results show that the trained CNN models converged smoothly during training and exhibited stable prediction behavior under $\pm 3\%$ fault-current input perturbations. The CNN also achieved lower prediction errors than the MLP baseline, indicating that preserving the relay–fault-location fault-current matrix structure through convolutional feature extraction is beneficial for relay-setting prediction. The ETAP verification results further demonstrated that the CNN-predicted settings could restore the expected primary–backup relay operating sequence and maintain CTI constraints under the studied DER-rich scenarios. In the 16-bus system, the proposed method eliminated the 12 miscoordination events observed under fixed relay settings. In the IEEE 37-bus

test feeder, the method also eliminated the 6 critical miscoordination events identified in the baseline case, indicating its applicability to larger and more complex distribution test systems.

The computational results based on the IEEE 37-bus test system further show that RIA and MPSO require approximately 300 s and 250 s, respectively, to generate optimized relay-setting labels for one operating scenario in the offline optimization stage, whereas the trained CNN-based surrogate model can provide relay-setting recommendations in approximately 0.05 s during online inference. These results demonstrate that the proposed offline–online architecture transfers the main computational burden from online operation to offline training and supports near-real-time relay-setting decision support.

In summary, this study bridges the computational gap between heuristic optimization and real-time protection decision support. By rapidly recommending coordinated TMS and PCS values, the proposed CNN-based preventive calibration framework provides a practical tool for improving protection coordination robustness in DER-rich distribution networks. For practical deployment, the CNN output should be used as a decision-support recommendation and verified through setting-boundary checks, CTI verification, relay-sequence confirmation, and secure engineering procedures before implementation. In addition, major permanent topology changes require updating the dataset and retraining or fine-tuning the model before the trained CNN is applied to the modified network.

Author Contributions: T.-S.Z. provided project ideas and performed system models, data analysis, model simulations, article writing, and editing. J.-L.J. provided project ideas, research funding support, and related experience. J.-J.C. performed the MATLAB coding and data curation, and also conducted simulations. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Atomic Research Institute, Taiwan, R.O.C., under contract No. NL1150167, project No. 115A013.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data and program codes generated and used in this study are associated with an ongoing project funded by the National Atomic Research Institute, Taiwan, R.O.C. Due to the project's confidentiality agreement, the original program code and related project materials are not publicly available during the project period. Upon completion of the project, the relevant program codes and materials will be transferred to the National Atomic Research Institute in accordance with the project agreement.

Acknowledgments: The authors would like to thank the National Atomic Research Institute, Taiwan, R.O.C., for the financial support provided to this work under contract No. NL1150167, project No. 115A013. The authors declare that no artificial intelligence tools were used to generate the scientific results, data, analyses, or conclusions of this study. AI-assisted language editing tools may have been used only to improve readability and grammar. All content was reviewed and approved by the authors.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Basak, P.; Chowdhury, S.; Halder nee Dey, S.; Chowdhury, S.P. A literature review on integration of distributed energy resources in the perspective of control, protection and stability of microgrid. *Renew. Sustain. Energy Rev.* **2012**, *16*, 5545–5566. [\[CrossRef\]](#)
2. Holguin, J.P.; Rodriguez, D.C.; Ramos, G. Reverse power flow (RPF) detection and impact on protection coordination of distribution systems. *IEEE Trans. Ind. Appl.* **2020**, *56*, 2393–2401. [\[CrossRef\]](#)
3. Alasali, F.; Al-Hayajneh, A.; Zarour, E.; Alsmadi, O. Optimal protection coordination scheme of overcurrent relays for microgrid system. In Proceedings of the 10th International Conference on Renewable Energy Research and Application (ICRERA), Istanbul, Turkey, 26–29 September 2021; pp. 88–93. [\[CrossRef\]](#)

4. Mahindara, V.R.; Rodriguez, D.F.C.; Pujiantara, M.; Priyadi, A.; Purnomo, M.H.; Muljadi, E. Practical challenges of inverse and definite-time overcurrent protection coordination in modern industrial and commercial power distribution system. *IEEE Trans. Ind. Appl.* **2021**, *57*, 187–197. [\[CrossRef\]](#)
5. Lestari, D.S.; Pujiantara, M.; Purnomo, M.H.; Rahmatullah, D. Adaptive DOCR coordination in loop distribution system with distributed generation using firefly algorithm-artificial neural network. In Proceedings of the International Conference on Information and Communications Technology (ICOIACT), Yogyakarta, Indonesia, 6–7 March 2018; pp. 579–584. [\[CrossRef\]](#)
6. Zeineldin, H.H.; Mohamed, Y.A.R.I.; Khadkikar, V.; Pandi, V.R. A protection coordination index for evaluating distributed generation impacts on protection for meshed distribution systems. *IEEE Trans. Smart Grid* **2013**, *4*, 1523–1532. [\[CrossRef\]](#)
7. Atsever, M.B.; Karacasu, O.; Hocaoglu, M.H. Performance analysis of different optimization algorithms on overcurrent relay coordination problem in distribution systems. Proceeding of the 11th Smart Grid Conference (SGC), Tabriz, Iran, 7–9 December 2021; pp. 1–6. [\[CrossRef\]](#)
8. Zhan, T.S.; Su, C.L.; Lee, Y.D.; Jiang, J.L.; Yu, J.T. Adaptive OCRs coordination in distribution system with distributed energy resources contribution. *AIMS Energy* **2023**, *11*, 1278–1305. [\[CrossRef\]](#)
9. Zhan, T.S.; Lee, Y.D.; Jiang, J.L. Optimal OCR coordination in a high penetration distribution power system using a refined immune algorithm with an auto-tuning reproductive mechanism. *AIMS Energy* **2024**, *12*, 1225–1263. [\[CrossRef\]](#)
10. Mohamad Idin, M.A.; Osman, M.K.; Mohd Napiah, N.A.; Saad, Z.; Ahmad, K.A.; Omar, S. Time-current characteristic measurement of overcurrent relay in power system using multilayer perceptron network. In Proceedings of the International Conference on Intelligent and Advanced Systems, Kuala Lumpur, Malaysia, 15–17 June 2010; pp. 1–5. [\[CrossRef\]](#)
11. Musirikare, A.; Pujiantara, M.; Tjahjono, A.; Purnomo, M.H. ANN-based modeling of directional overcurrent relay characteristics applied in radial distribution system with distributed generations. Proceeding of the 10th International Conference on Information Technology and Electrical Engineering (ICITEE), Bali, Indonesia, 24–26 July 2018; pp. 52–57. [\[CrossRef\]](#)
12. Rahmatullah, D.; Dewantara, B.Y.; Iradiratu, D.P.K. Adaptive DOCR coordination in loop electrical distribution system with DG using artificial neural network LMBP. In Proceedings of the International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), Yogyakarta, Indonesia, 21–22 November 2018; pp. 560–565. [\[CrossRef\]](#)
13. Kabeel, M.A.; Eladany, M.M.; ElDesouky, A.A. A centralized protection scheme for microgrids with artificial neural network-based on fault detection and location. In Proceedings of the 23rd International Middle East Power Systems Conference (MEPCON), Cairo, Egypt, 13–15 December 2022; pp. 1–5. [\[CrossRef\]](#)
14. Hartmann, N.B.; dos Santos, R.C.; Grilo, A.P.; Vieira, J.C.M. Hardware implementation and real-time evaluation of an ANN-based algorithm for anti-islanding protection of distributed generators. *IEEE Trans. Ind. Electron.* **2018**, *65*, 5051–5059. [\[CrossRef\]](#)
15. Kadhum, S.J.K.; Ibrahim, A.A. An analysis system of fault diagnosis and classification in electrical energy system distribution networks based on convolutional neural network. In Proceedings of the 7th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), Ankara, Turkiye, 26–28 October 2023; pp. 1–6. [\[CrossRef\]](#)
16. Tant, C.; Gokaraju, R. Optimized CNN for fault recognition in power systems with high penetrations of inverter-based resources. In Proceedings of the IEEE/IAS 61st Industrial and Commercial Power Systems Technical Conference (I&CPS), Montreal, QC, Canada, 11–14 May 2025; pp. 1–8. [\[CrossRef\]](#)
17. Kordowich, G.; Jaworski, M.; Lorz, T.; Scheibe, C.; Jaeger, J. A hybrid protection scheme based on deep reinforcement learning. Proceeding of 2022 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe), Novi Sad, Serbia, 10–12 October 2022; pp. 1–6. [\[CrossRef\]](#)
18. Singh, P.; Safavizadeh, A.; Jatskevich, J.; Choudhary, N.K.; Singh, N. Hybrid PSO–DNN framework for improved relay coordination in AC microgrids under high-impedance faults. In Proceedings of the 7th Global Power, Energy and Communication Conference (GPECOM), Bochum, Germany, 11–13 June 2025; pp. 908–913. [\[CrossRef\]](#)
19. Kim, W.H.; Kim, J.Y.; Chae, W.K.; Kim, G.; Lee, C.K. LSTM-based fault direction estimation and protection coordination for networked distribution system. *IEEE Access* **2022**, *10*, 40348–40357. [\[CrossRef\]](#)
20. Jiang, J.L.; Chen, Y.L.; Zhan, T.S.; Chi, J.J. Convolutional neural network-enhanced protection coordination under high DER penetration. In Proceedings of the 3rd International Conference on Artificial Intelligence and Power Engineering (AIPE), Nara, Japan, 28–31 October 2025; pp. 26–31. [\[CrossRef\]](#)
21. *IEEE Std 242-2001™*; IEEE Recommended Practice for Protection and Coordination of Industrial and Commercial Power Systems. The Institute of Electrical and Electronics Engineers, Inc.: New York, NY, USA, 2001.

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.