JOURNAL OF NANO- AND ELECTRONIC PHYSICS
Vol. 18 No 4, 04010(5pp) (2026)

ЖУРНАЛ НАНО- ТА ЕЛЕКТРОННОЇ ФІЗИКИ
Том 18 № 4, 04010(5cc) (2026)

REGULAR ARTICLE

Design of Triband Monopole Antenna Using Machine Learning Assisted Optimization

B. Sivashanmugavalli*, R. Mahalakshmi, S. Anusiya

B.S. Abdur Rahman Crescent Institute of Science and Technology, Vandalore, Chennai, India

(Received 20 March 2026; revised manuscript received 19 August 2026; published online 21 August 2026)

This paper presents a machine learning-driven optimization strategy for designing a compact triband monopole antenna operating at 2.4 GHz, 3.5 GHz, and 5.8 GHz, suitable for WLAN, WiMAX, and sub-6 GHz 5G communication. A wide range of antenna models with varied slot patterns were simulated in Ansys HFSS to create a dataset of return-loss (S_{11}) characteristics. Three deep-learning structures a 3-layer Feedforward Neural Network (FNN), a 5-layer FNN, and a Convolutional Neural Network (CNN) were employed to predict optimal slot dimensions, trained with Adam, Nadam, and AdamW optimizers. The 5-layer FNN (Nadam) achieved a prediction accuracy of 66.66 %, while the 3-layer FNN (Adam) produced the best overall match for triband response. To further refine predictions, Differential Evolution (DE) optimization was applied to tune geometric parameters. HFSS re-validation confirmed improved antenna behavior with higher gain, lower S_{11}, and reduced manual design effort. The proposed ML-assisted hybrid framework demonstrates a scalable, time-efficient, and accurate approach for multiband antenna optimization.

Keywords: Triband monopole antenna, Machine learning, FNN, CNN, HFSS, Optimizer, Antenna design.

DOI: 10.21272/jnep.18(4).04010 PACS number: 84.40.Ba

1. INTRODUCTION

The rapid advancement of wireless technologies such as 5G, IoT, and WLAN has created a growing requirement for compact and multiband antennas that can efficiently operate across multiple frequency ranges. Conventional antenna design methods depend on manual parameter variation and electromagnetic simulations, which are often time-consuming and computationally demanding. These traditional techniques become inefficient when dealing with complex geometries or multi-objective performance optimization involving gain, bandwidth, and impedance matching. Although several recent studies have applied machine learning (ML) to improve antenna prediction and design accuracy, important limitations remain. Many works achieve good predictive results but lack integration with EM solvers or experimental verification, while others do not compare different optimizers or address generalization to real antenna structures.

Surrogate model assisted differential evolution for antenna synthesis (SADEA), is presented in [1]. Joung et al., [2] presented the KNN and SVM based antenna selection. In existing research, the performance metrics like bandwidth, efficiency and isolation are predicted using machine learning algorithms. In [3] the MIMO antenna is designed and the isolation between the antenna elements are predicted using Gaussian regression method. Rahman et al., designed and simulated the 5G antenna using electromagnetic simulator. The simulation model parameters are fed to the ML. The efficiency and bandwidth of the simulated antenna are predicted using ML model. Wu et al. [5] introduced a hybrid framework combining Gaussian Process Regression with CNNs for estimating resonant frequencies and return loss.

In existing literature, machine learning techniques are predominantly employed either to predict the performance metrics of antennas or to apply specific optimization algorithms for fine-tuning antenna geometry selection. The antenna design parameters can also be predicted using machine learning techniques; however, the existing literature lacks a comprehensive comparison of different ML algorithms for this purpose. To address these gaps, this paper proposes a hybrid optimization framework that integrates HFSS-generated simulation data, deep learning models, and Differential Evolution to enable automated design of a triband monopole antenna.

Three deep architectures 3-layer FNN, 5-layer FNN, and CNN are trained using Adam, Nadam, and AdamW optimizers. Differential Evolution is then applied to refine the predicted slot dimensions and locate the most suitable design combinations. This integrated method effectively reduces the number of required simulations, enhances prediction reliability, and provides a structured workflow for fast and precise triband antenna optimization.

2. ANTENNA DESIGN

A rectangular micro strip patch antenna serves as the antenna's fundamental structure; it was selected for its small profile, simplicity of integration, and adaptability for contemporary wireless applications. Prior to implementing multiband operation, it constitutes the base configuration and is intended to resonate initially at 3.5 GHz.

* Correspondence e-mail: sivashanmugavalli@crescent.education

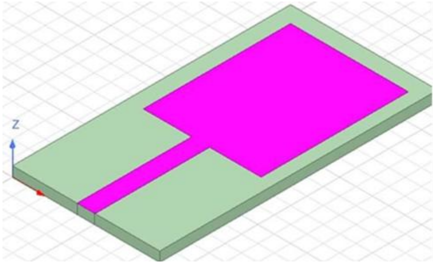

Fig. 1 – Rectangular patch antenna

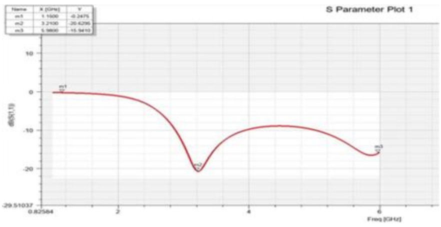

Fig. 2 – Return loss of rectangular patch antenna

Fig. 1 shows the edge fed rectangular patch antenna and Fig. 2 shows a return loss curve to ensure the resonance.

Resonating at 3.5 GHz, the base antenna is a rectangular microstrip patch built on a FR4 substrate ($\varepsilon_r = 4.4$, $h = 1.6$ mm). Three slots are cut out of the patch to enable triband capability (2.4 GHz, 3.5 GHz, and 5.8 GHz). Accurate slot location and dimensioning are essential because the mutual coupling and capacitance between slots define the resonant behavior.

3. OPTIMIZATION USING MACHINE LEARNING

In order to attain the intended multi-band resonance with the least amount of return loss, optimization in this study focuses on optimizing the antenna slot dimensions. S_{11} performance is predicted using machine learning models, which inform design parameter revisions. Design time and manual iterations are greatly decreased as a result. The flow chart in Fig. 3 explains the proposed work's technique.

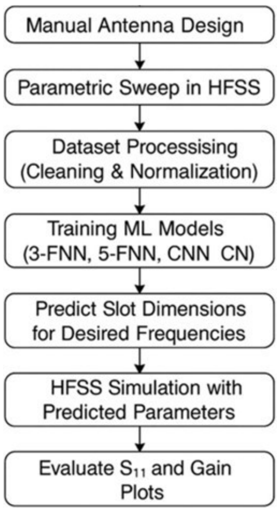

Fig. 3 – ML-assisted antenna design workflow

A. Data Collection

In Ansys HFSS, six slot parameters are varied to generate a large dataset covering 73,000 antenna designs. For each design, the S_{11} return loss curve is extracted across the frequency range. The dataset uses the slot dimensions as inputs and the S_{11} values as outputs, enabling machine learning models to predict antenna performance from the slot design.

B. Dataset Processing and Splitting

In order to ensure successful convergence during training, the raw dataset is preprocessed by normalizing the input characteristics to bring all slot dimensions to a common scale. For consistency across all samples, the S_{11} output is arranged into fixed frequency steps. Following preprocessing, the dataset is divided into training and testing sets, with 20 % put aside for testing and validation and the remaining 80 % used to train the machine learning models.

C. Machine Learning Model Training

To map slot parameters to the corresponding S_{11} response, three neural network models were trained: a 3-layer FNN, a 5-layer FNN, and a CNN. The 3-layer FNN uses six input nodes for the slot dimensions, one hidden layer with ReLU activation, and an output layer for predicting S_{11} values. With the Adam optimizer, it converged quickly and reached an accuracy of 66.66 %. The 5-layer FNN adds extra hidden layers and dropout, allowing it to learn more complex nonlinear patterns while reducing overfitting; this model, trained with AdamW, obtained 33.33 % accuracy. A 1D CNN was also implemented to capture frequency-dependent variations in S_{11} through convolution and pooling layers followed by fully connected layers. Although the CNN trained with Nadam did not accurately reach the target frequency points, it was still effective in identifying subtle input-output relationships, but performed slightly below the FNN models in overall accuracy.

D. Optimizers Used

By adjusting model weights to reduce the error between expected and actual outputs, optimizers are essential to neural network training. To effectively modify the weights during backpropagation, they employ techniques based on gradient descent. A competent optimizer is crucial for accurate and efficient learning since it speeds up convergence, prevents local minima, and enhances the model's capacity to generalize to new input. To improve model generalization and convergence, a number of optimizers were investigated. Below is a brief summary of their roles.

1. Adam (Adaptive Moment Estimation)

The advantages of momentum and RMSProp are combined in Adam, an adaptive learning rate optimizer. It keeps track of two exponential moving averages: the uncentered variance of gradients (v_t) and the mean of gradients (m_t). To guarantee stability in early iterations, these are bias- corrected. Adam's resilience makes it popular and efficiently speeds up convergence.

The update rule is expressed in the following Below Eq.:

$$\theta_{t+1} = \theta_t - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}$$

Where θ_t is the parameter at iteration t, η is the learning rate, $\hat{m}_t$ and $\hat{v}_t$ are the bias-corrected first and second moments, and ϵ is a small constant to prevent division by zero.

2. Nadam (Nesterov-accelerated Adaptive Moment Estimation)

By adding Nesterov momentum to its update rule, Nadam improves on Adam and provides improved convergence for certain datasets. Nesterov's technique is more accurate and steadier than ordinary momentum since it predicts the direction of the subsequent update. When faced with challenging, non-convex situations, Nadam typically performs better.

The Nadam update rule is given in Below Eq.:

$$\theta_{t+1} = \theta_t - \eta\left(\frac{\beta_1 \hat{m}_t + \frac{(1-\beta_1)}{1-\beta_1^t} g_t}{\sqrt{\hat{v}_t}+\epsilon}\right)$$

Here, g_t is the current gradient, and other variables remain as defined in Eq. (5), with β_1 as the decay rate for the first moment.

3. AdamW (Adam with Decoupled Weight Decay)

Better generalization results from AdamW's modification of Adam, which separates weight decay from the gradient update step. AdamW uses weight decay directly on the weights, in contrast to conventional L2 regularization that modifies gradients. This stabilizes training and helps prevent overfitting, particularly in deep networks.

Its weight update rule is shown in Below Eq.:

$$\theta_{t+1} = \theta_t - \eta\left(\frac{\hat{m}_t}{\sqrt{\hat{v}_t}+\epsilon} + \lambda \cdot \theta_t\right)$$

Where λ is the weight decay factor, and all other variables follow from Adam's notation.

4. RMSProp (Root Mean Square Propagation)

An adaptive learning technique called RMSProp modifies the learning rate according to every parameter. In order to smooth the updates, it keeps a moving average of squared gradients and divides the current gradient by the square root of this average. This makes training stable even in the presence of noisy gradients.

The update expression is defined in Below Eq.:

$$\theta_{t+1} = \theta_t - \eta \frac{g_t}{\sqrt{E[g^2]_t+\epsilon}}$$

Where $E[g^2]_t$ is the exponential average of past squared gradients.

5. Adadelta

By lowering aggressive, monotonically declining learning rates, Adadelta outperforms Adagrad. Without the need for manual η adjustment, it automatically adjusts the learning rate based on a moving window of gradient changes. This makes it appropriate for large-scale and online training assignments.

The weight update for Adadelta is given in Below Eq.:

$$\theta_{t+1} = \theta_t - \frac{RMS[\Delta\theta]_t}{RMS[g]_t} \cdot g_t$$

Where $RMS[\Delta\theta]_t$ is the root mean square of previous weight updates and $RMS[g]_t$ is the root mean square of gradients.

E. Differential Evolution (DE) for Optimization

A global optimization technique called Differential Evolution is employed to adjust the slot characteristics that the machine learning models anticipate. It uses crossover, mutation, and selection processes to evolve a population of potential solutions. Finding the best antenna designs involves minimizing the fitness function, which is determined by the projected return loss (S_{11}). DE works particularly well for nonlinear, high-dimensional situations.

The DE objective is expressed in Below Eq.:

$$Fitness = \min_{\vec{x}} \sum_f \left|S_{11}^{pred}(f, \vec{x})\right|$$

Where S_{11}^{pred} is the expected return loss across frequency fand $\vec{x}$ is the slot parameter vector. For validation, configurations with the lowest fitness are re- simulated in HFSS.

4. RESULTS AND DISCUSSION

Using the trained machine learning models, a specialized Python application was created to automate the antenna slot optimization procedure. In order to estimate S_{11} values depending on input slot dimensions, this application integrated the models' predictive capacity. This method was used to aim resonance at about – 60 dB by methodically updating the slot parameters to obtain substantially negative return loss. The Python application made it possible to explore the design space effectively, which decreased manual labor and sped up the optimization process. The slot dimensions were improved using the ML predictions; the resulting outcomes are shown in the table below and confirmed by HFSS simulations.

A. Evaluation of 3 Layer Feed Forward Neural Network (3 FNN) Models

Table 1 – Comparison of 3-Layer FNN Models

Optimizer	**Resonance Frequencies (GHz)**	**REMARKS**
Adam	2.65 / 3.9 / 5.65	All bands matched closely
Nadam	2.9	Only one band achieved
AdamW	2.8	Slight shift, limited coverage

The 3-layer FNN with layers Input, 128, 64, and Output was trained using three optimizers. Adam (Model 1) produced the best triband response, with only minor frequency shifts below 0.3 GHz. Models trained with Nadam and AdamW mainly converged to a single band around 2.8 to 2.9 GHz, indicating limited learning of the higher-frequency features. Nadam (Model 5) was the next best model, successfully identifying all three bands. Overall, the results show that predicting multiband antenna behavior is highly dependent on the choice of optimizer, as both Adam and AdamW tended to fall back to a single-band output even in deeper networks.

Table 2 – Comparison of 5-Layer FNN Models

Optimizer	HFSS Frequencies (GHz)	Remarks
Adam	2.8	Single band near match
Nadam	2.8 / 4.45 / 5.65	Accurate prediction for all bands
AdamW	3	Shifted response

Near 3 GHz, all CNN variations generated a single resonance, suggesting that they were unable to adequately capture the multi-band behavior.

Table 3 – Comparison of CNN Models

Optimizer	HFSS Frequencies (GHz)	Remarks
Adam	2.95	Single band observed
RMSProp	3	Missed all target bands
AdaDelta	3	Missed all target bands

This implies that Feedforward Neural Networks (FNNs) perform better than Convolutional Neural Networks (CNNs) in numerical slot- parameter regression when it comes to precisely forecasting multi-band antenna responses.

Table 4 – Evaluation Results

Optimizer	RMSE (GHz)	R^2 Score
Adam	0.86	0.959
Nadam	0.601	0.820

The improved antenna showed stable peak gains of 2.8 to 4.5 dBi and an omnidirectional pattern, making it suitable for wireless use. Slot dimensions predicted by the Adam-trained 3-layer FNN were highly accurate, and HFSS simulations confirmed resonances at 2.65, 3.9, and 5.65 GHz. This model gave the most consistent results across all three bands. Figs. 5, 6 show the modified antenna and its return loss.

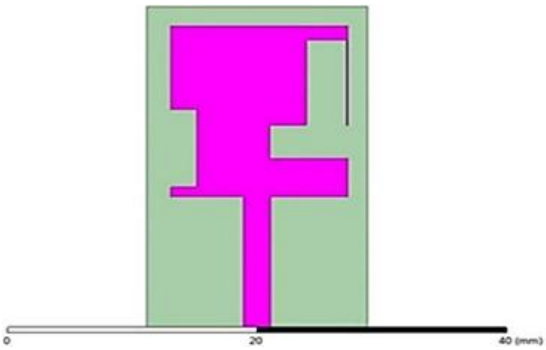

Fig. 5 – Antenna Design with 3 – FNN Adam Optimizer Slot Dimensions

The Nadam-trained 5-layer FNN also gave strong results, generating slot dimensions that produced resonances below – 10 dB in the target bands. It showed good gain and generalization. Figs. 7, 8 show the modified antenna and its return loss.

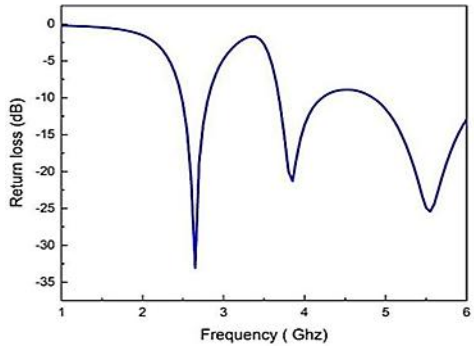

Fig. 6 – S_{11} Plot with 3 – FNN Adam Optimizer Slot Dimensions

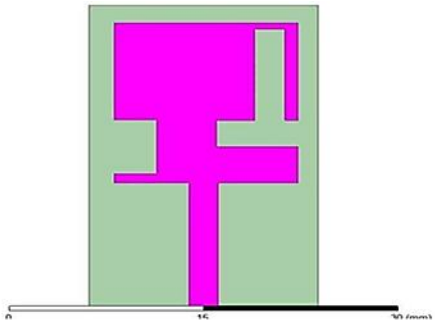

Fig. 7 – Antenna Design 5 – FNN Nadam Optimizer Slot Dimensions

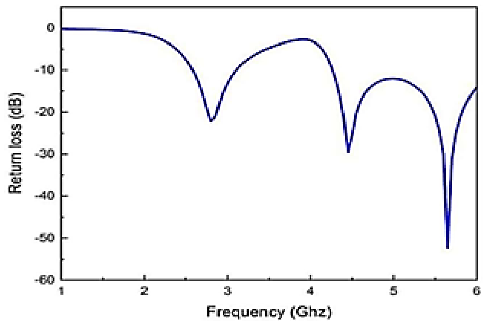

Fig. 8 – S_{11} Plot 5 – FNN Nadam Optimizer Slot Dimensions

5. CONCLUSION

This work presents a machine learning–assisted design framework for improving the performance of a triband monopole antenna. By integrating electromagnetic simulation data, deep learning prediction models, and Differential Evolution optimization, the approach achieves faster and more accurate results than conventional manual design methods. The findings show that combining ML with HFSS simulations effectively identifies slot dimensions that achieve the desired resonant frequencies with improved gain and reduced return loss. Future work will expand the dataset with additional antenna parameters and material properties to enhance model generalization.

REFERENCES

1. B. Liu, H. Aliakbarian, Z. Ma, G.A.E. Vandenbosch, G. Gielen, P. Excell, *IEEE Trans. Antennas Propag.* **62** No 1, 7 (2014) https://doi.org/10.1109/TAP.2013.2283605.
2. Joung, *IEEE Commun. Lett.* **20** No 11, 2241 (2016) https://doi.org/10.1109/LCOMM.2016.2594776.
3. K. Sharma, G.Pr. Pandey, *PIER C* **108**, 227 (2021) https://doi.org/10.2528/PIERC20120301.
4. Rahman, Md Afzalur, Samir Salem Al-Bawri, Samia Larguech, Sultan S. Alharbi, Saeed Alsowail, Noorlindawaty Md Jizat, Mohammad Tariqul Islam, *Sci. Rep.* **15** No 1, 22866 (2025) https://doi.org/10.1038/s41598-025-06391-1.
5. Wu et al., *Journal of Antenna Design and Optimization* **36** No 3, 205 (2022).

Розробка тридіапазонної монопольної антени з використанням оптимізації за допомогою машинного навчання

B. Sivashanmugavalli, R. Mahalakshmi, S. Anusiya

B.S. Abdur Rahman Crescent Institute of Science and Technology, Vandalore, Chennai, India

Представлено стратегію оптимізації на основі машинного навчання для проєктування компактної тридіапазонної монопольної антени, що працює на частотах 2,4 ГГц, 3,5 ГГц та 5,8 ГГц, придатної для WLAN, WiMAX та зв'язку 5G нижче 6 ГГц. Широкий спектр моделей антен з різними схемами розміщення слотів було змодельовано в Ansys HFSS для створення набору даних характеристик відбитих втрат (*S*11). Для прогнозування оптимальних розмірів слотів було використано три структури глибокого навчання: 3-шарову нейронну мережу прямого зв'язку (FNN), 5-шарову FNN та згорткову нейронну мережу (CNN), які були навчені за допомогою оптимізаторів Adam, Nadam та AdamW. 5-шарова FNN (Nadam) досягла точності прогнозування 66,66 %, тоді як 3-шарова FNN (Adam) забезпечила найкращу загальну відповідність для тридіапазонної характеристики. Для подальшого уточнення прогнозів для налаштування геометричних параметрів було застосовано диференціально-еволюційну (DE) оптимізацію. Повторна перевірка HFSS підтвердила покращену поведінку антени з вищим коефіцієнтом підсилення, нижчим *S*11 та зменшенням зусиль ручного проектування. Запропонована гібридна структура на основі машинного навчання демонструє масштабований, ефективний за часом та точний підхід до оптимізації багатодіапазонної антени.

Ключові слова: Тридіапазонна монопольна антена, Машинне навчання, FNN, CNN, HFSS, Оптимізатор, Проєктування антени.