Department of Electronics and Telecommunications

International Institute of Information Technology Bhubaneswar

Bhubaneswar, Odisha, 751003, India

ML optimised RFID antenna using openEMS and octave

This report is submitted in partial fulfilment of the requirements for the award of the degree of

Bachelor of Technology

in

Electronics and Telecommunications Engineering

By

Biswajit Nahak (ID: B222016)
Daksh Katiyar (ID: B222018)
Komlaksh Sharma (ID: B222025)

Under the guidance of

Prof. Dr. Sanjeev Kumar Mishra

(Assistant Professor)

(IIIT Bhubaneswar)

TABLE OF CONTENTS :

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to all those who supported us throughout the course of our antenna design project.

First and foremost, we are deeply thankful to our guide, Dr. Sanjeev Kumar Mishra, Assistant Professor, Department of ETC, IIIT Bhubaneswar, for their invaluable guidance, constructive feedback, and constant encouragement during every phase of the project. Their expertise and support played a crucial role in the successful completion of our work.

We are also grateful to the faculty and staf of the ETC for teaching laboratory facilities, simulation tools (such as CST/Ansys HFSS), and technical skills essential for our research. A special thanks to our fellow students and friends who supported us directly or indirectly through their helpful discussions and suggestions. This project has been a great learning experience for both of us, and we are truly thankful for the opportunity to explore and apply advanced concepts in antenna design.

OBJECTIVE:

The primary objective of this project is to design, simulate, and optimize a microstrip RFID patch antenna using computational electromagnetic tools and machine learning techniques. The project aims to improve critical antenna performance metrics such as return loss (S11), voltage standing wave ratio (VSWR), directivity, input impedance, and efficiency by tuning the geometric parameters of the antenna.

This is achieved by first generating a large dataset of antenna simulations using openEMS and Octave, sweeping over a defined range of patch length, width, and substrate thickness. The resulting dataset is used to train a machine learning model that

🕮 INTRODUCTION

With the growing demand for smart identification and tracking systems, **Radio Frequency Identification (RFID)** has become a key technology in sectors such as logistics, healthcare, transportation, and retail. At the heart of any RFID system lies the antenna — a critical component responsible for wireless communication between the tag and the reader.

Among various antenna types, the **microstrip patch antenna** stands out due to its low profile, ease of fabrication, low cost, and compatibility with planar and embedded structures. However, designing an efficient patch antenna involves a delicate balance of parameters including the **patch length, width, and substrate thickness**, which collectively determine key performance metrics such as **resonant frequency, impedance matching, directivity, return loss, and efficiency**.

Traditionally, antenna design has relied on trial-and-error approaches or limited parametric sweeps. In this project, we adopt a **machine learning–driven optimization framework** to automate and enhance the design process. By generating a large simulation dataset and training predictive models, we are able to identify **near-optimal design parameters** quickly — significantly reducing development time.

This report details the design and simulation of a **2.45 GHz RFID patch antenna**, the generation of a large performance dataset using openEMS, and the application of ML algorithms to predict and optimize antenna behavior. The best design, as determined by the model, is then validated through full-wave simulation in openEMS.

Antenna Design

predicts antenna performance from geometry and then identifies the **best-performing configuration**.

Ultimately, the optimized design is validated using openEMS by simulating the electromagnetic behaviour of the antenna and analysing its real-world performance. The project also explores practical aspects such as **simulation time constraints.**

Through this project, we demonstrate how combining electromagnetic simulation with data-driven ML models can significantly reduce design cycles and lead to more efficient antenna systems for RFID and wireless communication applications.

⚙ ANTENNA DESIGN WORKFLOW IN OCTAVE (USING openEMS):

To validate the performance of the ML-optimized RFID antenna, we implemented a full electromagnetic simulation using **Octave and openEMS**. This simulation models the physical structure of the patch antenna, including the substrate, ground plane, and feed. The design process follows a structured workflow — from geometry setup and meshing to simulation execution.

◇ 1. Simulation Initialization

close all; clear; clc;

physical_constants;

unit = 1e-3; % define all dimensions in millimeters

Closes figures, clears variables and console.

Loads physical constants like EPS0, MU0, and sets the unit scale to millimeters.

◇ 2. Define Antenna and Substrate Dimensions

patch.width = 42;

patch.length = 35;

substrate.epsR = 4.3;

substrate.thickness = 2;

Sets the patch size (42 mm × 35 mm).

Chooses FR4 substrate ($\varepsilon r = 4.3$) with thickness = 2 mm.

◇ 3. Feeding and Simulation Box Setup

feed.pos = -6; % feed location along x-axis

feed.R = 50; % feed resistance

SimBox = [200 200 150]; % size of the simulation domain

Feed is added 6 mm from the patch center.

Simulation domain is defined in all three directions.

◇ 4. FDTD Setup

f0 = 2e9; fc = 1e9;

FDTD = InitFDTD('NrTs', 30000);

FDTD = SetGaussExcite(FDTD, f0, fc);

FDTD = SetBoundaryCond(FDTD, {'MUR', ..., 'MUR'});

Initializes the time-domain solver.

Excitation is a Gaussian pulse centered at 2 GHz.

Sets MUR absorbing boundary conditions.

◇ 5. Geometry & Mesh Initialization

CSX = InitCSX();

mesh.x/y/z = [...]; % initial simulation space

Initializes the geometry and mesh for the simulation using CSXCAD.

◇ 6. Antenna Structure Definition

Patch:

CSX = AddMetal(CSX, 'patch');

CSX = AddBox(...); % creates metal patch

Substrate:

CSX = AddMaterial(..., 'substrate');

CSX = SetMaterialProperty(...);

CSX = AddBox(...); % defines dielectric substrate

Ground Plane:

CSX = AddMetal(CSX, 'gnd');

CSX = AddBox(...); % adds ground below substrate

◇ 7. Feeding Port Setup

[CSX, port] = AddLumpedPort(..., start, stop, [0 0 1], true);

Adds a lumped port (feed) between two Z-points.

Oriented in Z-direction, with excitation.

◇ 8. Mesh Refinement

mesh = DetectEdges(...);

mesh = SmoothMesh(...);

CSX = DefineRectGrid(CSX, unit, mesh);

Mesh is smoothed near edges for better accuracy.

Defined on a rectilinear grid.

◇ 9. Add Field and Far-Field Monitors

CSX = AddDump(...);

[CSX, nf2ff] = CreateNF2FFBox(...);

Adds a monitor for near-field and far-field analysis.

Essential for calculating directivity and radiation pattern.

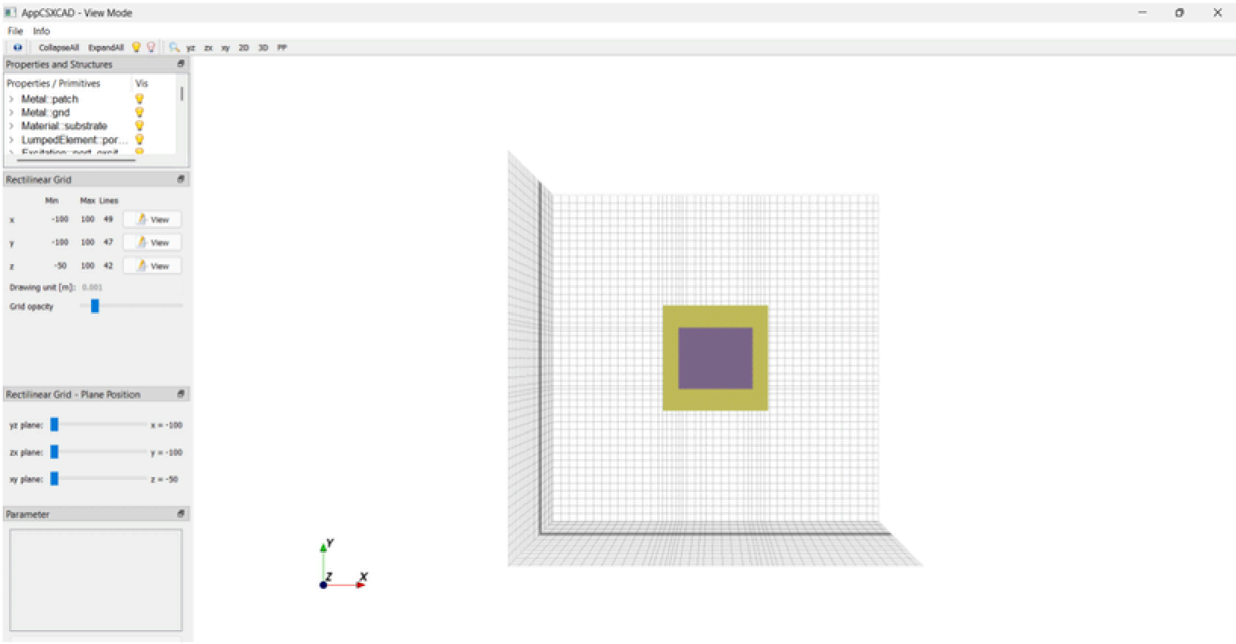
◇ 10. Prepare and Run Simulation

WriteOpenEMS(...);

CSXGeomPlot(...); % shows antenna geometry

RunOpenEMS(...); % runs the solver

Writes geometry to XML file and executes the solver.



Simulation and Saving into CSV

To create a large dataset for machine learning-based optimization, we conducted a parameter sweep of the patch antenna design by varying the following parameters:

Parameter	Range	Step
Patch Length (mm)	25 to 35	1 mm
Patch Width (mm)	35 to 45	1 mm
Substrate Thickness (mm)	1.0 to 2.0	0.1 mm

This resulted in a total of 11 × 11 × 11 = 1331 unique antenna configurations. Simulating them all in a single run would have been highly time-consuming and memory-intensive. To manage this efficiently, we implemented a batch-wise simulation approach in Octave, running 100 simulations per batch, and saving the results into separate CSV files.

Maximize Directivity and Efficiency

Minimize VSWR and S11 (more negative)

```
 ML-Optimized Parameters:
Length        35.0
Width         42.0
Thickness      2.0
Name: 4784, dtype: float64

 Predicted Performance:
Directivity_dBi      7.5720
Efficiency           0.8691
S11_dB             -11.5887
VSWR                 1.7012
Name: 4784, dtype: float64

 Composite Score: 0.7818
```

This modeling approach enabled us to efficiently explore the design space and identify the most promising geometry without running additional full-wave simulations. The predicted configuration was later validated using openEMS, confirming the reliability of the machine learning model.

Each simulation computes key performance parameters such as:

S11 (Return Loss) in dB

VSWR (Voltage Standing Wave Ratio)

Input Impedance (real and imaginary)

Directivity in dBi

Efficiency in percentage

Once all batches were completed, we merged the individual CSV files into a single dataset using Python for further analysis.Here are the first five datset of our full merged csv-

```
import pandas as pd
import numpy as np
from sklearn.ensemble import RandomForestRegressor
from sklearn.preprocessing import MinMaxScaler

# Load dataset

df = pd.read_csv("C:\\openEMS_v0.0.36\\openEMS\\matlab\\rfid_project\\antenna_dataset_full.csv")
```

```
df.head()
```

	Length	Width	Thickness	S11_dB	VSWR	Impedance	Directivity_dBi	Efficiency
0	25.0	35.0	1.0	-9.46	2.29	41.10	6.18	0.95
1	26.0	35.0	1.0	-7.60	2.03	56.59	5.08	0.73
2	27.0	35.0	1.0	-9.58	1.98	52.74	6.29	0.82
3	28.0	35.0	1.0	-10.26	1.73	49.69	6.65	0.79
4	29.0	35.0	1.0	-11.14	1.91	47.15	6.38	0.90

🧠 ML-Based Modeling and Parameter Prediction:

To streamline antenna design, we developed a machine learning–based model that predicts antenna performance (like directivity, S11, VSWR, and efficiency) from geometric parameters — specifically patch length, width, and substrate thickness.

Modeling Approach

The dataset of 1331 simulations was loaded using Pandas.

Features used for training: Length, Width, Thickness

Targets: Directivity_dBi, Efficiency, S11_dB, and VSWR

Separate Random Forest Regressors were trained for each target using scikit-learn.

To select the best antenna configuration, a composite scoring function was created to balance multiple goals:

Final Simulation & Performance Evaluation:

After identifying the best-performing antenna geometry using machine learning, we conducted a final full-wave electromagnetic simulation using openEMS and Octave. The aim of this step was to **verify the accuracy of ML predictions** by extracting all key antenna performance metrics through post-processing.

◇ 1. Frequency Setup and Port Calculation

freq = linspace(max([1e9, f0 - fc]), f0 + fc, 501);

port = calcPort(port, Sim_Path, freq);

Defines the frequency sweep range for analysis.

Loads voltage and current signals from the simulation result files.

◇ 2. Return Loss (S11) and Resonant Frequency

s11 = port.uf.ref ./ port.uf.inc;

*[min_s11_db, f_res_ind] = min(20*log10(abs(s11)));*

f_res = freq(f_res_ind);

Calculates the reflection coefficient (S11) in decibels.

Identifies the frequency where return loss is minimum → resonant frequency.

◇ 3. Input Impedance

Zin = port.uf.tot ./ port.if.tot;

Zin_res = Zin(f_res_ind);

Zin_real = real(Zin_res);

Zin_imag = imag(Zin_res);

Computes the antenna's input impedance at resonance.

Separates real and imaginary components.

◇ 4. VSWR Calculation

VSWR = (1 + S11_abs) / (1 - S11_abs);

Calculates the Voltage Standing Wave Ratio, a key measure of impedance matching.

◇ 5. Far-Field Radiation Analysis

nf2ff = CalcNF2FF(...);

Converts near-field results to far-field.

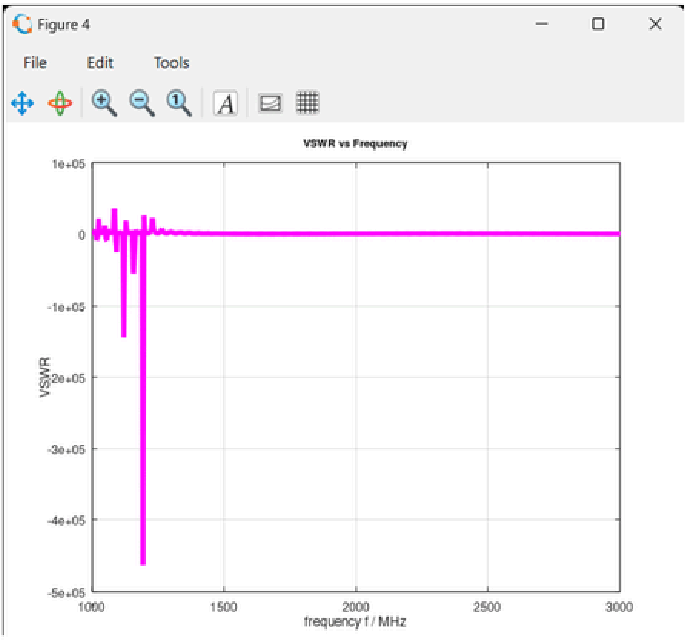
◇ 6. Results Display

fprintf(...);

Displays final numerical outputs in console:

S11 (dB)

VSWR



🎯 Application

The ML-optimized microstrip patch antenna designed in this project is primarily intended for **RFID systems operating at 2.45 GHz**, a standard frequency in the ISM band. Due to its compact size, planar structure, and enhanced performance, this antenna can also be adapted for a variety of wireless communication applications, including:

- **Inventory management** and **logistics tracking**
- **Contactless payment systems**
- **Healthcare asset monitoring**
- **Smart agriculture** and **IoT devices**
- **Wireless sensor networks (WSNs)**

Its directional radiation and good impedance matching make it a suitable choice for both **active** and **passive RFID applications**, especially where space and power are limited.

Input Impedance

Directivity

Efficiency

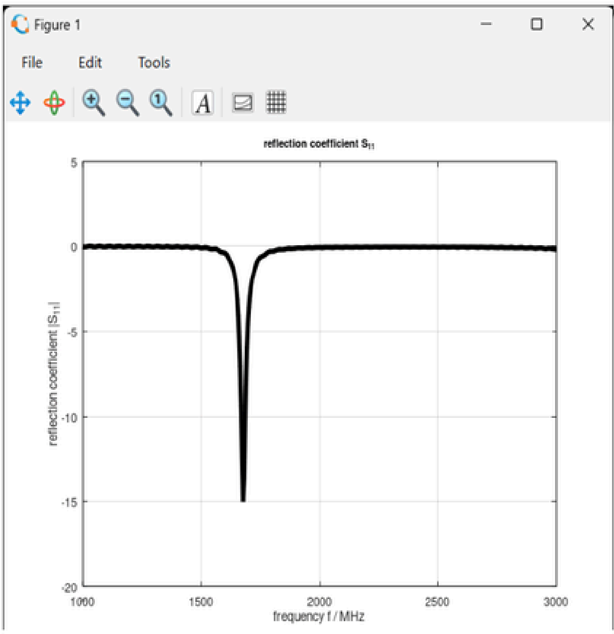
```
----------------------------------------------------------------
--- Simulation Results at Resonance (1676.00 MHz) ---
S11 (dB): -15.02 dB
VSWR: 1.43
Input Impedance Zin: 71.14 + j-3.98 Ohm
Maximum Directivity (Dmax in dBi): 5.95 dBi
Efficiency: 83.86 %
>> |
```

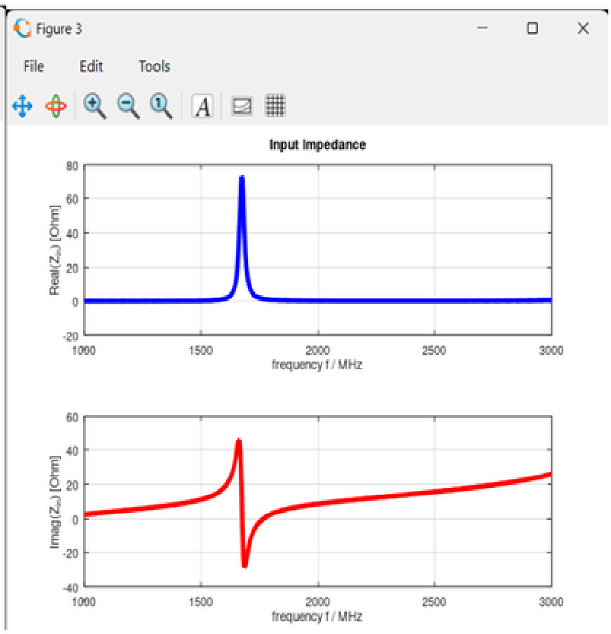
◇ 7. Performance Visualization

S11 Plot vs Frequency

Input Impedance Plot: Real and Imaginary parts

VSWR vs Frequency





🔮 Future Scope

This project successfully demonstrates how machine learning can accelerate antenna design and performance prediction. However, several future improvements and expansions are possible:

- 🔧 **Feed Optimization**: Investigate different feed positions or techniques (e.g., inset feed, probe feed) for even better impedance matching.
- 📈 **Multi-objective Optimization**: Incorporate more advanced ML methods (like Bayesian optimization or genetic algorithms) for simultaneously optimizing multiple goals.
- 🌐 **MEEP Integration**: Use MEEP (FDTD in Python) for faster and more scalable simulations in future iterations.
- 🛰️ **Antenna Arrays**: Extend the methodology to design and optimize antenna arrays for beam steering or MIMO systems.
- 📡 **Real-World Fabrication**: Manufacture and measure the antenna prototype to validate simulation predictions.
- 🤖 **AutoML Pipeline**: Build a full design loop where geometry is predicted, simulated, and scored automatically — reducing human input even further.